



OnCommand Workflow Automation Documentation

OnCommand Workflow Automation

NetApp
October 09, 2025

This PDF was generated from <https://docs.netapp.com/us-en/workflow-automation/index.html> on October 09, 2025. Always check docs.netapp.com for the latest.

Table of Contents

OnCommand Workflow Automation Documentation	1
OnCommand Workflow Automation Release Notes	2
Installation and setup for Linux	3
Overview of OnCommand Workflow Automation	3
WFA features	3
WFA license information	3
OnCommand Workflow Automation deployment architecture	4
OnCommand Workflow Automation installation and setup overview	4
Known limitations for OnCommand Workflow Automation	5
System requirements for installing OnCommand Workflow Automation	6
Hardware requirements for installing WFA	6
Software requirements for installing WFA	6
Ports required for Workflow Automation	7
Prerequisites for installing Workflow Automation	9
Required configuration information	9
Install Perl modules on CentOS and RHEL	10
Managing high availability	12
Set up Workflow Automation in VCS for high availability	12
Configure an earlier version of OnCommand Workflow Automation for high availability	17
Uninstall Workflow Automation in a VCS environment	17
Back up and restore the OnCommand Workflow Automation database and configurations on Linux	18
Setting up OnCommand Workflow Automation	18
Access OnCommand Workflow Automation	18
OnCommand Workflow Automation data sources	19
Create local users	23
Configure the credentials of a target system	24
Configuring OnCommand Workflow Automation	26
Disable the default password policy	30
Modify the default password policy	30
Enable or disable remote access to the OnCommand Workflow Automation database	30
Modify the transaction timeout setting of OnCommand Workflow Automation	31
Configure the timeout value for Workflow Automation	31
Enabling ciphers and adding new ciphers	32
Upgrade from OnCommand Workflow Automation 3.1 or later	32
Pack identification during upgrade	33
Upgrading third-party products	34
Upgrade OpenJDK	34
Upgrade MySQL on Linux	34
Backing up the OnCommand Workflow Automation database	35
Backup and restoration of user credentials	36
Back up the WFA database from the web portal	36
Backing up the WFA database using the CLI	36
Backing up the WFA database using REST APIs	38

Restoring the OnCommand Workflow Automation database	38
Restore the WFA database	39
Restoring the WFA database using the CLI	39
Restoring the WFA database using REST APIs	41
Reset the admin password created during installation	42
Import OnCommand Workflow Automation content	43
Considerations while importing OnCommand Workflow Automation content	43
Migrate the OnCommand Workflow Automation installation	44
Uninstall OnCommand Workflow Automation	45
Managing OnCommand Workflow Automation SSL certificate	45
Replace the default Workflow Automation SSL certificate	45
Create a certificate signing request for Workflow Automation	46
Managing Perl and Perl modules	48
Configure your preferred Perl distribution	48
Troubleshooting installation and configuration issues	48
Cannot view Performance Advisor data in WFA	48
Related documentation for OnCommand Workflow Automation	49
Other references	49
Tool references	49
Installation and setup for Windows	50
Overview of OnCommand Workflow Automation	50
WFA features	50
WFA license information	50
OnCommand Workflow Automation deployment architecture	51
OnCommand Workflow Automation installation and setup overview	51
Known limitations and enhancements	52
System requirements for installing OnCommand Workflow Automation	53
Hardware requirements for installing WFA	53
Software requirements for installing WFA	53
Ports required for Workflow Automation	54
Prerequisites for installing Workflow Automation	55
Required configuration information	56
Managing high availability	57
Set up Workflow Automation in MSCS for high availability	57
Configure earlier versions of OnCommand Workflow Automation for high availability	64
Uninstall Workflow Automation in an MSCS environment	65
Setting up OnCommand Workflow Automation	65
Access OnCommand Workflow Automation	65
OnCommand Workflow Automation data sources	66
Create local users	71
Configure the credentials of a target system	72
Configuring OnCommand Workflow Automation	73
Disable the default password policy	78
Modify the default password policy for Windows	79
Enable remote access to the OnCommand Workflow Automation database on Windows	79

Restrict access rights of OnCommand Workflow Automation on the host	80
Modify the transaction timeout setting of OnCommand Workflow Automation	80
Configure the timeout value for Workflow Automation	81
Enabling ciphers and adding new ciphers	81
Upgrade OnCommand Workflow Automation	82
Upgrade from OnCommand Workflow Automation 3.1 or later versions	82
Upgrading third-party products	84
Upgrade OpenJDK	84
Upgrade MySQL	84
Upgrade ActiveState Perl	85
Backing up the OnCommand Workflow Automation database	86
Backup and restoration of user credentials	86
Back up the WFA database from the web portal	86
Back up the WFA database using the PowerShell script	87
Backing up the WFA database using the CLI	87
Backing up the WFA database using REST APIs	89
Restoring the OnCommand Workflow Automation database	90
Restore the WFA database	90
Restoring the WFA database using the CLI	91
Restoring the WFA database using REST APIs	92
Reset the admin password created during installation	93
Import OnCommand Workflow Automation content	94
Considerations while importing OnCommand Workflow Automation content	94
Migrate the OnCommand Workflow Automation installation	95
Uninstall OnCommand Workflow Automation	96
Managing OnCommand Workflow Automation SSL certificate	96
Replace the default Workflow Automation SSL certificate	97
Create a certificate signing request for Workflow Automation	97
Managing Perl and Perl modules	99
Configure your preferred Perl distribution	99
Manage site-specific Perl modules	99
Repair the ActivePerl installation	102
Troubleshooting installation and configuration issues	102
Cannot open the OnCommand Workflow Automation login page	102
Cannot view Performance Advisor data in WFA	103
OnCommand Workflow Automation (WFA) displays a blank page on Windows 2012	103
Related documentation for OnCommand Workflow Automation	103
Other references	103
Tool references	104
Workflow	105
Overview of OnCommand Workflow Automation	105
WFA features	105
WFA license information	105
Understanding Workflow Automation designer	106
Working with the building blocks in OnCommand Workflow Automation	106

What a playground database is	113
Managing workflows	115
Customize predefined workflows	115
Creating workflows	116
Create WFA workflow packs	142
Integrating WFA workflow packs with the SCM repository	147
Creating building blocks for workflows	150
Create a data source type	150
Create a command	151
Create a finder	154
Create a filter	155
Create a dictionary entry	156
Create a function	156
Create a template	157
Create a cache query	158
Create recurring schedules	159
Define filter rules	160
Add approval points	161
Coding guidelines for WFA	162
Guidelines for variables	162
Guidelines for indentation	166
Guidelines for comments	166
Guidelines for logging	168
Guidelines for error handling	170
General PowerShell and Perl conventions for WFA	173
Considerations for adding custom PowerShell and Perl modules	174
WFA cmdlets and functions	174
PowerShell and Perl WFA modules	175
Considerations while converting PowerShell commands to Perl	178
Guidelines for WFA building blocks	180
Reserved words	193
How you use REST APIs	194
References to learning material	194
Windows PowerShell	194
Data ONTAP PowerShell toolkit	195
Perl	195
NetApp Manageability SDK	196
Structured Query Language (SQL)	196
MVFLEX Expression Language (MVEL)	196
Regular expressions	196
Related documentation for OnCommand Workflow Automation	196
Other references	197
Tool references	197
Manage and configure	198
Overview of OnCommand Workflow Automation	198

WFA features	198
WFA license information	198
Create local users	199
Configuring OnCommand Workflow Automation	200
Configure authentication settings	200
Configure email notifications	201
Configure SNMP	201
Configure Syslog	203
Configure AutoSupport	203
Configure email notifications for data source acquisition failures	204
Configure resource reservation for workflows	205
Configure the credentials of a target system	205
Configure protocols for connecting to remote systems	207
OnCommand Workflow Automation Designer features	207
How repeat row works	207
What approval points are	209
How continue on failure works	210
How resource selection works	211
How reservation works	212
What incremental naming is	213
What conditional execution is	215
How return parameters work	215
What schemes are	216
What remote system types are	217
How entity versioning works	218
How you define workflows	220
How you map command parameters	221
All command categories	221
Commands that create objects	222
Commands that update objects	222
Commands that remove objects	222
Commands that deal with optional parent and child objects	223
Commands that update associations between objects	223
How user inputs are defined	223
User input type options	223
How you define constants	226
How you use REST APIs	227
Set up a data source	227
Configure a database user by running ocsetup on Windows	229
Configure a database user by running ocsetup on Linux	230
Configure a database user on Active IQ Unified Manager	231
Create workflow help content	231
Reserved words	232
Where to find information about MVEL	233
MVEL-enabled fields in OnCommand Workflow Automation	233

Examples of MVEL syntax	234
References to learning material	237
Windows PowerShell	237
Data ONTAP PowerShell toolkit	238
Perl	238
NetApp Manageability SDK	238
Structured Query Language (SQL)	239
MVFLEX Expression Language (MVEL)	239
Regular expressions	239
Supported workflows in ONTAP	239
Import OnCommand Workflow Automation content	242
Considerations while importing OnCommand Workflow Automation content	243
Export OnCommand Workflow Automation content	244
Disable cache acquisition for dictionary entries	245
Create WFA workflow packs	245
Delete OnCommand Workflow Automation packs	246
Add approval points	246
Define filter rules	247
Create a scheme	248
Edit a scheme	248
Delete a scheme	249
Add a new remote system type	249
Log Viewer window	250
Backup and Restore window	250
Backup section	250
Restore section	250
Users Window	250
Users table	251
Toolbar	253
New User dialog box	253
Edit User dialog box	254
Preferences window	255
Command button	256
Active Directory Groups window	256
Active Directory Groups table	256
Toolbar	258
New Active Directory Group dialog box	258
Edit Active Directory Group dialog box	260
Approval Portal window	261
Data Sources window	262
Data Sources table	262
History table	264
Toolbar	266
New Data Source dialog box	267
Edit Data Source dialog box	268

Credentials window	270
Credentials table	270
Toolbar	271
Test Connectivity dialog box	272
New Credentials dialog box	272
Edit Credentials dialog box	273
About dialog box	275
WFA version information	275
Workflows Window	275
Category pane	275
Schemes	275
Workflow pane	275
Executions window	276
Workflow table	276
Toolbar	279
Recurring Executions window	280
Recurring Executions table	280
Toolbar	282
Reservations window	282
Reservations table	282
Toolbar	284
Schedules window	284
Schedules table	285
Toolbar	286
New Schedule dialog box	286
Workflows window	287
Workflows table	287
Toolbar	289
New Workflow window	290
Workflow <workflow name> window	297
Execute Workflow dialog box	302
Edit Variable dialog box	303
Preview Workflow dialog box	305
Monitoring window	305
New Approval Point dialog box	308
Edit Approval Point dialog box	309
Finders window	310
Finders table	310
Toolbar	311
New Finder dialog box	313
Edit Finder dialog box	314
Clone Finder dialog box	316
Filters window	318
Filters table	318
Toolbar	320

New Filter dialog box	321
Edit Filter dialog box	322
Clone Filter dialog box	323
Commands window	325
Commands table	325
Toolbar	326
New Command Definition dialog box	327
Edit Command Definition dialog box	331
Clone Command Definition dialog box	334
Functions window	337
Functions table	337
Toolbar	338
Templates window	339
Templates table	340
Toolbar	341
New Template dialog box	342
Edit Template dialog box	343
Clone Template dialog box	343
Schemes window	344
Schemes table	344
Toolbar	346
Dictionary window	347
Dictionary table	347
Toolbar	349
New Dictionary Entry dialog box	350
Edit Dictionary Entry dialog box	352
Clone Dictionary Entry dialog box	353
Data Source Types window	355
Data Source Types table	355
Toolbar	356
Remote System Types window	358
Remote System Types table	358
Toolbar	359
New Remote System Type dialog box	360
Edit Remote System Type dialog box	361
Cache Queries window	362
Cache Queries list	363
Toolbar	364
Add Cache Query dialog box	365
Edit Cache Query dialog box	366
Clone Cache Query dialog box	366
Packs window	367
Packs table	367
Toolbar	368
New Pack dialog box	370

Edit Pack dialog box	370
Categories window	372
Categories table	372
Toolbar	373
New Category dialog box	374
Edit Category dialog box	375
Clone Category dialog box	376
Legal notices	378
Copyright	378
Trademarks	378
Patents	378
Privacy policy	378
Open source	378

OnCommand Workflow Automation Documentation

OnCommand Workflow Automation Release Notes

The release notes describe new features, upgrade notes, fixed issues, known limitations, and known issues.

[OnCommand Workflow Automation 5.1.1 Release Notes](#)

[OnCommand Workflow Automation 5.1 Release Notes](#)

Installation and setup for Linux

Overview of OnCommand Workflow Automation

OnCommand Workflow Automation (WFA) is a software solution that helps to automate storage management tasks, such as provisioning, migration, decommissioning, data protection configurations, and cloning storage. You can use WFA to build workflows to complete tasks that are specified by your processes. WFA supports ONTAP.

A workflow is a repetitive and procedural task that consists of sequential steps, including the following types of tasks:

- Provisioning, migrating, or decommissioning storage for databases or file systems
- Setting up a new virtualization environment, including storage switches and datastores
- Setting up storage for an application as part of an end-to-end orchestration process

Storage architects can define workflows to follow best practices and meet organizational requirements, such as the following:

- Using required naming conventions
- Setting unique options for storage objects
- Selecting resources
- Integrating internal configuration management database (CMDB) and ticketing applications

WFA features

- Workflow design portal to build workflows

The workflow design portal includes several building blocks, such as commands, templates, finders, filters, and functions, that are used to create workflows. The designer enables you to include advanced capabilities to workflows such as automated resource selection, row repetition (looping), and approval points.

The workflow design portal also includes building blocks, such as dictionary entries, cache queries, and data source types, for caching data from external systems.

- Execution portal to execute workflows, verify status of workflow execution, and access logs
- Administration/Settings option for tasks such as setting up WFA, connecting to data sources, and configuring user credentials
- Web service interfaces to invoke workflows from external portals and data center orchestration software
- Storage Automation Store to download WFA packs. The ONTAP 9.7.0 pack is bundled with WFA 5.1.

WFA license information

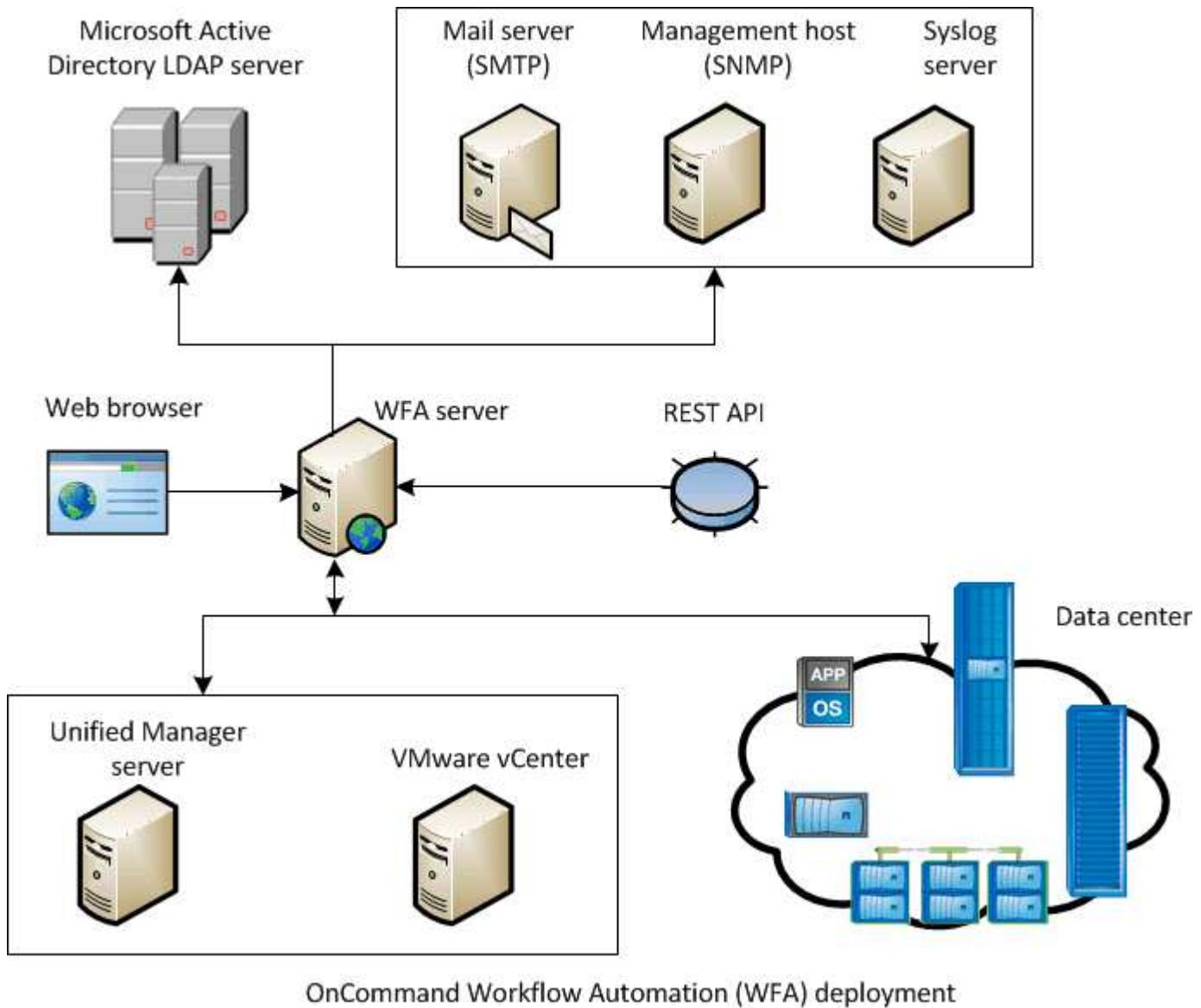
No license is required for using the OnCommand Workflow Automation server.

OnCommand Workflow Automation deployment architecture

OnCommand Workflow Automation (WFA) server is installed to orchestrate the workflow operations across several datacenters.

You can centrally manage your automation environment by connecting your WFA server to several Active IQ Unified Manager deployments and VMware vCenters.

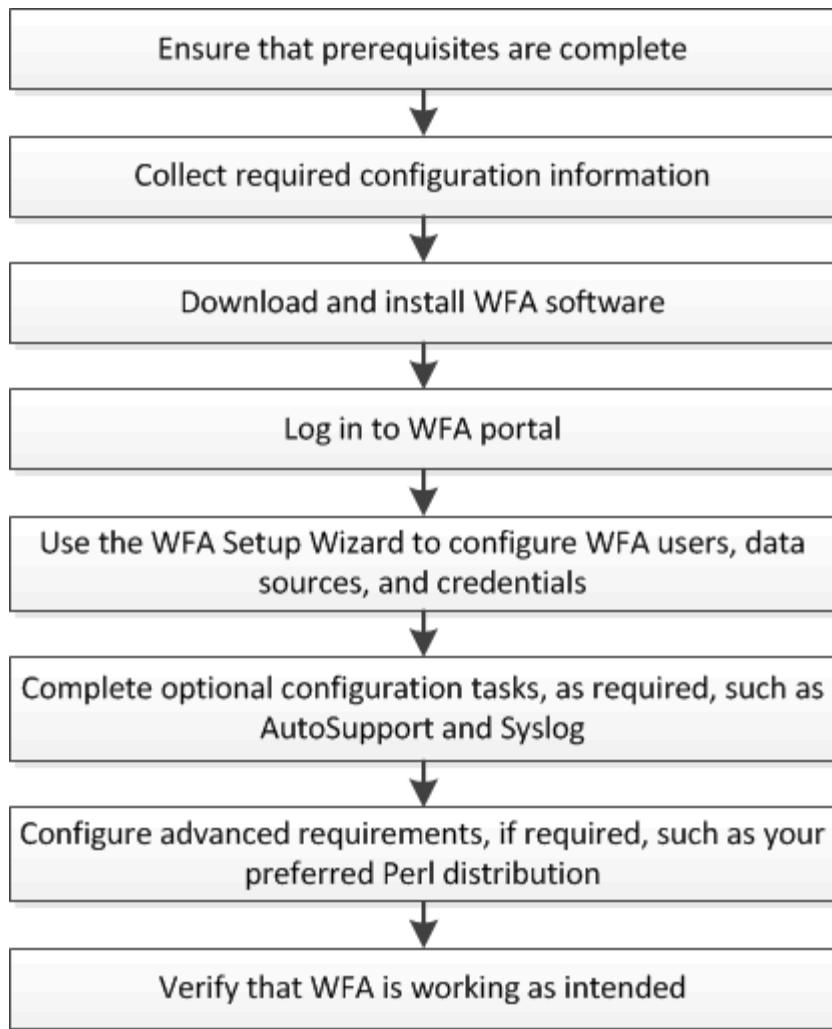
The following illustration shows a deployment example:



OnCommand Workflow Automation installation and setup overview

Installing OnCommand Workflow Automation (WFA) includes performing tasks such as preparing for the installation, downloading the WFA installer, and running the installer. After the installation is complete, you can configure WFA to meet your requirements.

The following flowchart illustrates the installation and configuration tasks:



Known limitations for OnCommand Workflow Automation

OnCommand Workflow Automation (WFA) 5.1 includes some limitations and unsupported features that you should be aware of before you install and configure WFA.

- **LDAP authentication**

- You can use only Microsoft Active Directory Lightweight Directory Access Protocol (LDAP) server for LDAP authentication.
- You must be a member of an LDAP group in order to use LDAP authentication.
- You must not use an LDAP user name that is part of a hierarchical structure of multiple domains for authentication or notification.



Microsoft Active Directory Lightweight Directory Services (AD LDS) is not supported.

- **WFA data sources types**

OnCommand Unified Manager 6.0, 6.1, and 6.2 data source types are deprecated in the WFA 4.1 release, and these data source types will not be supported in future releases.

- **WFA installed on Linux**

- Certified content from Data ONTAP operating in 7-Mode is currently not available.
- Commands that include only PowerShell code are not compatible with Linux.
- Certified commands for 7-Mode and VMware are currently not ported to Perl, and are therefore not compatible with Linux.

- **Creating category names**

- When a hyphen (-) is used in a category name, it is replaced with a space once the category is saved. For example, if the category name "abc-xyz" is provided, the category name is saved as "abc xyz", with the hyphen removed. To avoid this issue, do not use hyphens in category names.
- When a colon (:) is used in a category name, the text string before the colon is ignored once the category is saved. For example, if the category name "abc : xyz" is provided, the category name is saved as "xyz", with the "abc" string removed. To avoid this issue, do not use colons in category names.
- There is no check to prevent two categories having the same name. However, this causes a problem when selecting these categories from the navigation pane. To avoid this issue, make sure that each category name is unique.

System requirements for installing OnCommand Workflow Automation

You must be aware of the OnCommand Workflow Automation (WFA) hardware and software requirements before installing WFA.

Hardware requirements for installing WFA

The following table lists the minimum hardware requirements and the recommended hardware specifications for the WFA server.

Component	Minimum requirements	Recommended specifications
CPU	2.27 GHz or faster, 4 core, 64-bit	2.27 GHz or faster, 4 core, 64-bit
RAM	4 GB	8 GB
Free disk space	5 GB	20 GB



If you are installing WFA on a virtual machine (VM), you should reserve the required memory and CPU so that the VM has sufficient resources. The installer does not verify the CPU speed.

Software requirements for installing WFA

The following table lists all the operating system versions compatible with the WFA server.

Operating System	Version
Red Hat Enterprise Linux	7.0, 7.1, 7.2, 7.3, 7.4, 7.5 and 7.6 64-bit operating systems

Operating System	Version
CentOS	7.1, 7.2, 7.3, 7.4, 7.5, and 7.6 64-bit operating systems



WFA should be installed on dedicated physical machines or VMs. You must not install any other application on the server that runs WFA.

- One of the following supported browsers:
 - Mozilla Firefox
 - Microsoft Internet Explorer
 - Google Chrome
- Perl v5.x

You can obtain information about the Perl modules that should be installed by using the `./WFA-version_number.bin -l` command.

For more details, see the [Interoperability Matrix Tool](#).



Antivirus applications might prevent WFA services from starting.

To avoid this issue, configure antivirus scanning exclusions for the following WFA directories:

- The directory where you have installed WFA
- The directory where you have installed Perl
- The directory where you have installed OpenJDK
- The MySQL Data Directory

Related information

[NetApp Interoperability Matrix Tool](#)

Ports required for Workflow Automation

If you are using a firewall, you must be aware of the required ports for Workflow Automation (WFA).

The default port numbers are listed in this section. If you want to use a non-default port number, you must open that port for communication. For more details, see the documentation on your firewall.

The following table lists the default ports that should be open on the WFA server:

Port	Protocol	Direction	Purpose
80, 443	HTTP, HTTPS	Incoming	Opening WFA and logging in

Port	Protocol	Direction	Purpose
80, 443, 22	HTTP, HTTPS, SSH	Outgoing	Command execution (ZAPI, PowerCLI)
445, 139, 389, 636	Microsoft-DS, NetBIOS-ssn, AD LDAP, AD LDAPS	Outgoing	Microsoft Active Directory LDAP authentication
161	SNMP	Outgoing	Sending SNMP messages on the status of workflows
3306	MySQL	Incoming	Caching read-only user
25	SMTP	Outgoing	Mail notification
80, 443, 25	HTTP, HTTPS, SMTP	Outgoing	Sending AutoSupport messages
514	Syslog	Outgoing	Sending logs to a syslog server

The following table lists the default ports that should be open on the Unified Manager server:

Port	Protocol	Direction	Purpose
3306	MySQL	Incoming	Caching data from Active IQ Unified Manager 6.0 and later

The following table lists the default port that should be open on the VMware vCenter:

Port	Protocol	Direction	Purpose
443	HTTPS	Incoming	Caching data from VMware vCenter

The following table lists the default port that should be open on the SNMP host machine:

Port	Protocol	Direction	Purpose
162	SNMP	Incoming	Receiving SNMP messages on the status of workflows

Prerequisites for installing Workflow Automation

Before installing OnCommand Workflow Automation (WFA), you must ensure that you have the required information and you have completed certain tasks.

Before you install WFA on a system, you must have completed the following tasks:

- Downloading the WFA installation file from the NetApp Support Site and copying the file to the server on which you want to install WFA



You must have valid credentials to log in to the NetApp Support Site. If you do not have valid credentials, you can register on the NetApp Support Site to obtain the credentials.

- Verifying that the system has access to the following, as applicable:

- Storage controllers
- Active IQ Unified Manager



If your environment requires Secure Shell (SSH) accessibility, you must ensure that SSH is enabled on the target controllers.

- Verifying that Perl v5.10.1 is installed

Required configuration information

Unit or system	Details	Purpose
Arrays	<ul style="list-style-type: none">• IP address• User name and password	Perform operations on storage systems Root or admin account credentials are required for storage (arrays).
External repositories such as OnCommand Balance and custom databases	<ul style="list-style-type: none">• IP address• User name and password of a read-only user account	Acquire dataYou must create the relevant WFA content, such as dictionary entries and cache queries for the external repositories, in order to acquire data from the external repositories.

Unit or system	Details	Purpose
Mail server	<ul style="list-style-type: none"> • IP address • User name and password <p> User name and password are required if your mail server requires authentication.</p>	Receive WFA notifications through email
AutoSupport server	<ul style="list-style-type: none"> • Mail host 	Send AutoSupport messages through SMTP. If you do not have a mail host configured, you can use HTTP or HTTPS to send AutoSupport messages.
Microsoft Active Directory (AD) LDAP server	<ul style="list-style-type: none"> • IP address • User name and password • Group name 	Authenticate and authorize using AD LDAP or AD LDAPS
SNMP management application	<ul style="list-style-type: none"> • IP address • Port 	Receive WFA SNMP notifications
Syslog server	<ul style="list-style-type: none"> • IP address 	Send log data

Related information

[NetApp Support](#)

Install Perl modules on CentOS and RHEL

Some Perl modules are not included by default with the Perl package for Linux.

During WFA installation, the WFA installer verifies that all of the Perl modules are available in the system, and then proceeds when this requirement is met. You must install the Perl modules before installing OnCommand Workflow Automation (WFA).

 The WFA installer attempts to install the Perl modules automatically if the perl-modules are available in the yum repositories configured on the system. If the Perl modules are not available, the installer prompts the user to install the Perl modules manually. The one exception is the "perl-core" module. This module is not installed on the system automatically even if it is available in the yum repositories configured on the system. This is a known issue.

Perl Module	RPM Package Name
Perl core modules	perl-core

Perl Module	RPM Package Name
DBI	perl-DBI
XML::DOM	perl-XML-DOM
Term::ReadKey	perl-TermReadKey
HTTP::Request	perl-libwww-perl
XML::LibXML perl-XML-LibXML	perl-XML-LibXML
DBD::mysql	perl-DBD-MySQL
URI::URL	perl-URI
HTTP::Response	perl-libwww-perl
HTTP::Headers	perl-libwww-perl
Net::SSLeay	perl-Net-SSLeay
URI::Escape	perl-URI
LWP::Protocol::https perl-LWP-Protocol-https	perl-LWP-Protocol-https
XML::Parser	perl-XML-Parser
LWP::UserAgent	perl-libwww-perl
Net::LDAP	perl-LDAP
Date::Calc	perl-Date-CalcXML

Steps

1. Log in to the Linux server as a **root** user.
2. Verify that all of the Perl modules required for WFA are installed on the system:

```
./WFA-4.2.0.0.0.bin -l
```

3. If any Perl modules are not detected, check whether they are available in the configured repository:

```
yum search Perl-module-name
```

If the DBD::mysql module is not detected:

```
yum search perl-DBD-MySQL
```

4. If any Perl modules are not in the repository, configure the repository that contains the Perl module, or download the Perl module from the Internet.
5. Install any missing Perl modules from the configured repository:

```
yum -y install Perl-module-name
```

Install the DBD::mysql module from the configured repository:

```
yum -y install perl-DBD-MySQL
```

Managing high availability

You can configure a high-availability setup to provide constant support for network operations. If one of the components fail, the mirrored component in the setup takes over the operation and provides uninterrupted network resources. You can also back up the WFA database and supported configurations so that you can recover the data in case of a disaster.

Set up Workflow Automation in VCS for high availability

You must install and configure Workflow Automation (WFA) in a Veritas Cluster Server (VCS) environment to set up high availability and provide failover. Before you install WFA, you must verify that all required components are configured correctly.

A high-availability setup provides constant support for application operations. If one of the components fails, the mirrored component in the setup takes over the operation and provides uninterrupted network resources.



VCS is the only clustering solution that is supported by WFA on Linux.

Configure VCS to install OnCommand Workflow Automation

Before you install OnCommand Workflow Automation (WFA) in Veritas Cluster Server (VCS), you must ensure that the cluster nodes are properly configured to support WFA.

- VCS must be installed on both nodes of the cluster according to the instructions in the *Veritas Cluster Server 6.1.1 Installation Guide*.
- To receive notifications about cluster events, VCS software must be configured for SNMP and SMTP according to the instructions in the *Veritas Cluster Server Administrator's Guide*.
- All requirements and guidelines for configuring cluster servers must be met according to the VCS documentation.
- SnapDrive for UNIX must be installed if you want to create LUNs using SnapDrive for UNIX.
- Both the cluster nodes must be running a supported version of the operating system.

The minimum supported operating systems are Red Hat Enterprise Linux 7.0 and VCS 6.1.1.

- The same version of WFA must be installed using the same path on both the cluster nodes.
- The WFA server must be connected to the storage system through Fibre Channel (FC) or iSCSI.

- The latency must be minimum between the WFA server and the storage system.
- The FC link must be active, and the LUNs that are created must be accessible to both the cluster nodes.
- A minimum of two network interfaces must be set up on each system: one for node-to-node communication and the other for node-to-client communication.
- The name of the network interface used for node-to-client communication should be the same on both the systems.
- A separate heartbeat link must be established between the cluster nodes; otherwise, the network interface is used to communicate between the cluster nodes.
- A shared location must be created for high availability.

You can use SnapDrive for UNIX to create the shared location.

You can also manage the LUNs using SnapDrive or the storage system command-line interface. See the SnapDrive for UNIX compatibility matrix for more information.

Steps

1. Verify that VCS is installed correctly: `hastatus -summary`

Both the nodes must be online, and the VCS service must be running on both the nodes.

2. Verify that the LUNs are accessible to both the nodes by using one of the following options:

- Manage the LUNs natively.
- Use SnapDrive for UNIX:
 - i. Install SnapDrive for UNIX on both the nodes.
 - ii. Configure SnapDrive for UNIX on both nodes.
 - iii. From the first node, run the `snapdrive storage create` command to create a LUN.
 - iv. Verify that the LUN created on the first node is visible on the second node by running the `snapdrive storage show -all` command.

Install OnCommand Workflow Automation on Linux

You can install OnCommand Workflow Automation (WFA) by using the command-line interface (CLI).

- You must have reviewed the installation prerequisites.

Installation prerequisites

- You must have downloaded the WFA installer from the NetApp Support Site.

If you are installing WFA on a virtual machine (VM), the name of the VM must not include the underscore (_) character.

You can change the default installation location at the shell prompt: `./WFA-version_number.bin [-i WFA_install_directory] [-d mysql_data_directory]`

If you change the default installation location, the MySQL data directory is not deleted when you uninstall WFA. You must manually delete the directory.



Before reinstalling WFA 4.2 or later, you must delete the MySQL data directory if you have uninstalled MySQL.

Steps

1. Log in to the Linux server as a `root` user.
2. Navigate to the directory where the executable `.bin` file is located.
3. Install WFA by choosing one of the following methods:
 - Interactive installation
 - i. Start the interactive session: `./WFA-version_number.bin`
 - ii. Enter the credentials for the default admin user, and then press Enter.

You must note the credentials for the admin user and ensure that the password meets the following criteria:

- Minimum of eight characters
- One uppercase character
- One lowercase character
- One numeral
- One special character
- The following special characters are not supported in a password and cause installation failure:
`'" ; < > , = & ^ |`

iii. Accept the default ports for WFA configuration or provide custom ports, and then press Enter.

iv. Specify your company name and a site name, and then press Enter.

The site name can include the location of the WFA installation, for example, Pittsburgh, PA.

v. Verify that WFA is installed successfully by choosing one of the following actions:

- Access WFA through a web browser.
- Verify that the NetApp WFA Server service and the NetApp WFA Database service are running:

```
service wfa-server status
service wfa-db status
```

◦ Silent installation

At the shell prompt:

```
./WFA-version_number.bin [-u admin_user_name] [-p admin_user_password] [-m https_port] [-n http_port] [-c company_name] [-s site_name] [-i install_directory] [-d mysql_data_directory] [-y] [-b]
```

If you want to perform a silent installation, you must specify values for all of the command options. The command options are as follows:

Option	Description
-y	Option to skip Skips the installation confirmation
-b	Option to skip Skips creating a backup of the WFA database during an upgrade
-u	Admin user name
-p	Admin user password The admin user password must satisfy the following criteria: <ul style="list-style-type: none"> • Minimum of eight characters • One uppercase character • One lowercase character • One numeral • One special character • The following characters are not allowed and cause password input to fail: <ul style="list-style-type: none"> '' ; < > , = & ^
-m	HTTPS port
-n	HTTP port
-s	Site name
-c	Company name
-i	Installation directory path
-d	MySQL data directory
-h	Option to display Displays the Help

Related information

[NetApp Support](#)

Configure Workflow Automation in VCS

After you install Workflow Automation (WFA) in VCS, you must configure WFA in VCS using configuration scripts for high availability.

- You must have installed the same version of WFA on both the cluster nodes.
- You must have the same installation path for both the nodes.
- You must create a backup of WFA.

Steps

1. Log in to the first node of the cluster.
2. Use Cluster Manager to verify that the HA state of both nodes is running.
3. At the shell prompt, run the ha_setup.pl script to move the WFA data to the shared location and to configure WFA with VCS for failover: `perl ha_setup.pl --first [-t type_of_cluster_vcs] [-g cluster_group_name] [-e NIC_card_name] [-i IP_address] [-m Netmask] [-n cluster_name] [-f mount_point_of_shared_LUN] [-v name_of_logical_volume] [-d disk_group_name] [-l install_directory]`

For the default installation location, the script is available at /opt/netapp/wfa/bin/ha/.

```
perl ha_setup.pl --first -t vcs -g WFA -e eth0 -i 10.238.170.3 -m
255.255.255.0 -n wfa_cluster -f /mnt/wfa_mount/ -v lun_volume -d lun_dg -l
/opt/netapp/wfa
```

4. Use Cluster Manager to verify that the WFA services, mount point, virtual IP, NIC, and volume group are added to the cluster group.
5. Use Cluster Manager to move the WFA resources to the secondary node:
 - a. Select and right-click the cluster group.
 - b. Select **Switch To > Secondary Node**.
6. Verify that the data mount, virtual IP, volume group, and NIC cards are up on the second node of the cluster.
7. Take the WFA services offline by using Cluster Manager:
 - a. Select **WFA > Application > wfa-server**.
 - b. Right-click and select **Offline**.
 - c. Select **WFA > Application > wfa-db**.
 - d. Right-click and select **Offline**.
8. At the shell prompt, run the ha_setup.pl script on the secondary node of the cluster to configure WFA to use the data from the shared location: `perl ha_setup.pl --join [-t type_of_cluster_vcs] [-f mount_point_of_shared_LUN]`

For the default installation location, the script is available at /opt/netapp/wfa/bin/ha/.

```
perl ha_setup.pl --join -t vcs -f /mnt/wfa_mount/
```

9. Go to Cluster Manager and click **Cluster Group > Online > Server**.

It might take a while before Cluster Manager shows that the application resources are online. You can also right-click the application resources and verify whether the resources are online.

10. Ensure that WFA is accessible through the IP address used during this configuration.

Configure an earlier version of OnCommand Workflow Automation for high availability

You can configure OnCommand Workflow Automation (WFA) versions earlier than 3.1 for high availability.

Steps

1. Upgrade the existing version of WFA to the latest available version of WFA.

[Upgrade WFA](#)

This upgraded version of WFA is the primary node of the cluster.

2. Create a backup of the WFA database.

[Backing up the WFA database](#)

If any of the parameters were changed manually, you must create a backup of the WFA database, uninstall the existing WFA installation, install the latest available version of WFA, restore the backup, and then proceed with the Veritas Cluster Server (VCS) configuration.

3. Configure VCS to install WFA on the primary node.

[Configure VCS to install WFA](#)

4. Install the latest available version of WFA on the secondary node.

[Install WFA](#)

5. Configure WFA in VCS.

[Configure WFA in VCS](#)

The WFA server is configured for high availability.

Uninstall Workflow Automation in a VCS environment

You can uninstall Workflow Automation (WFA) from a cluster by deleting all the WFA services from the cluster nodes.

Steps

1. Take the services offline by using Cluster Manager:
 - a. Right-click the cluster group.
 - b. Select **Offline**, and then select the node.
2. Uninstall WFA on the first node, and then uninstall WFA on the second node.

[Uninstall OnCommand Workflow Automation](#)

3. Delete the cluster resources from Cluster Manager:
 - a. Right-click the cluster group.
 - b. Select **Delete**.

4. Manually delete the data in the shared location.

Back up and restore the OnCommand Workflow Automation database and configurations on Linux

You can back up and restore the OnCommand Workflow Automation (WFA) database and supported configurations so that you can recover the data in case of a disaster. The supported configurations include data access, HTTP timeout, and SSL certificates.

You must have administrator privileges or architect credentials.

You must create the backup in a secure location because restoring the backup will provide access to all the storage systems that are accessed by WFA.

- A comprehensive backup of WFA databases and configurations is required during disaster recovery and can be used in both standalone and high-availability environments.
- You can use only the CLI commands or REST APIs for comprehensive backup and restore operations during disaster recovery.

 You cannot use the web UI for backing up or restoring the WFA database during disaster recovery.

Steps

1. Back up the OnCommand Workflow Automation database.

[Backing up the OnCommand Workflow Automation database](#)

2. Restore a previous backup of the OnCommand Workflow Automation database.

[Restoring the OnCommand Workflow Automation database](#)

Setting up OnCommand Workflow Automation

After you complete installing OnCommand Workflow Automation (WFA), you must complete several configuration settings. You have to access WFA, configure users, set up data sources, configure credentials, and configure WFA.

Access OnCommand Workflow Automation

You can access OnCommand Workflow Automation (WFA) through a web browser from any system that has access to the WFA server.

You must have installed Adobe Flash Player for your web browser.

Steps

1. Open a web browser and enter one of the following in the address bar:

- `https://wfa_server_ip`

`wfa_server_ip` is the IP address (IPv4 or IPv6 address) or the fully qualified domain name (FQDN) of

the WFA server.

- If you are accessing WFA on the WFA server: `https://localhost/wfa`
If you have specified a non-default port for WFA, you must include the port number as follows:
 - `https://wfa_server_ip:port`
 - `https://localhost:port`
port is the TCP port number you have used for the WFA server during installation.

2. In the Sign in section, enter the credentials of the admin user that you have entered during installation.
3. In the **Settings > Setup** menu, set up the credentials and a data source.
4. Bookmark the WFA web GUI for ease of access.

OnCommand Workflow Automation data sources

OnCommand Workflow Automation (WFA) operates on data that is acquired from data sources. Various versions of Active IQ Unified Manager and VMware vCenter Server are provided as predefined WFA data source types. You must be aware of the predefined data source types before you set up the data sources for data acquisition.

A data source is a read-only data structure that serves as a connection to the data source object of a specific data source type. For example, a data source can be a connection to an Active IQ Unified Manager database of an Active IQ Unified Manager 6.3 data source type. You can add a custom data source to WFA after defining the required data source type.

For more information about the predefined data source types, see the Interoperability Matrix.

Related information

[NetApp Interoperability Matrix Tool](#)

Configuring a database user on DataFabric Manager

You must create a database user on DataFabric Manager 5.x to configure read-only access of the DataFabric Manager 5.x database to OnCommand Workflow Automation.

Configure a database user by running `ocsetup` on Windows

You can run the `ocsetup` file on the DataFabric Manager 5.x server to configure read-only access of the DataFabric Manager 5.x database to OnCommand Workflow Automation.

Steps

1. Download the `wfa_ocsetup.exe` file to a directory in the DataFabric Manager 5.x server from the following location:

`https://WFA_Server_IP/download/wfa_ocsetup.exe`.

`WFA_Server_IP` is the IP address (IPv4 or IPv6 address) of your WFA server.

If you have specified a non-default port for WFA, you must include the port number as follows:

`https://wfa_server_ip:port/download/wfa_ocsetup.exe`.

port is the TCP port number that you have used for the WFA server during installation.

If you are specifying an IPv6 address, you must enclose it with square brackets.

2. Double-click the `wfa_ocsetup.exe` file.
3. Read the information in the setup wizard and click **Next**.
4. Browse or type the OpenJDK location and click **Next**.
5. Enter a user name and password to override the default credentials.

A new database user account is created with access to the DataFabric Manager 5.x database.



If you do not create a user account, the default credentials are used. You must create a user account for security purposes.

6. Click **Next** and review the results.
7. Click **Next**, and then click **Finish** to complete the wizard.

Configure a database user by running `ocsetup` on Linux

You can run the `ocsetup` file on the DataFabric Manager 5.x server to configure read-only access of the DataFabric Manager 5.x database to OnCommand Workflow Automation.

Steps

1. Download the `wfa_ocsetup.sh` file to your home directory on the DataFabric Manager 5.x server using the following command in the terminal:

```
wget https://WFA_Server_IP/download/wfa_ocsetup.sh
```

WFA_Server_IP is the IP address (IPv4 or IPv6 address) of your WFA server.

If you have specified a non-default port for WFA, you must include the port number as follows:

```
wget https://wfa_server_ip:port/download/wfa_ocsetup.sh
```

port is the TCP port number that you have used for the WFA server during installation.

If you are specifying an IPv6 address, you must enclose it with square brackets.

2. Use the following command in the terminal to change the `wfa_ocsetup.sh` file to an executable:

```
chmod +x wfa_ocsetup.sh
```

3. Run the script by entering the following in the terminal:

```
./wfa_ocsetup.sh OpenJDK_path
```

OpenJDK_path is the path to OpenJDK.

`/opt/NTAPdfm/java`

The following output is displayed in the terminal, indicating a successful setup:

```
Verifying archive integrity... All good.  
Uncompressing WFA OnCommand Setup.....  
*** Welcome to OnCommand Setup Utility for Linux ***  
    <Help information>  
*** Please override the default credentials below ***  
Override DB Username [wfa] :
```

4. Enter a user name and password to override the default credentials.

A new database user account is created with access to the DataFabric Manager 5.x database.



If you do not create a user account, the default credentials are used. You must create a user account for security purposes.

The following output is displayed in the terminal, indicating a successful setup:

```
***** Start of response from the database *****  
>>> Connecting to database  
<<< Connected  
*** Dropped existing 'wfa' user  
==> Created user 'username'  
>>> Granting access  
<<< Granted access  
***** End of response from the database *****  
***** End of Setup *****
```

Set up a data source

You must set up a connection with a data source in OnCommand Workflow Automation (WFA) to acquire data from the data source.

- For Active IQ Unified Manager 6.0 and later, you must have created a database user account on the Unified Manager server.

See the *OnCommand Unified Manager Online Help* for details.

- The TCP port for incoming connections on the Unified Manager server must be open.

See the documentation on your firewall for details.

The following are the default TCP port numbers:

TCP port number	Unified Manager server version	Description
3306	6.x	MySQL database server

- For Performance Advisor, you must have created an Active IQ Unified Manager user account with a minimum role of GlobalRead.

See the *OnCommand Unified Manager Online Help* for details.

- The TCP port for incoming connections on the VMware vCenter Server must be open.

The default TCP port number is 443. See the documentation on your firewall for details.

You can add multiple Unified Manager server data sources to WFA using this procedure. However, you must not use this procedure if you want to pair Unified Manager server 6.3 and later with WFA and use the protection functionality in Unified Manager server.

For more information about pairing WFA with Unified Manager server 6.x, see the *OnCommand Unified Manager Online Help*.

 While setting up a data source with WFA, you must be aware that Active IQ Unified Manager 6.0, 6.1, and 6.2 data source types are deprecated in the WFA 4.0 release, and these data source types will not be supported in future releases.

Steps

1. Access WFA using a web browser.
2. Click **Settings**, and under **Setup** click **Data Sources**.
3. Choose the appropriate action:

To...	Do this...
Create a new data source	Click  on the toolbar.
Edit a restored data source if you have upgraded WFA	Select the existing data source entry, and click  on the toolbar.

If you have added a Unified Manager server data source to WFA and then upgraded the version of the Unified Manager server, WFA will not recognize the upgraded version of the Unified Manager server. You must delete the previous version of the Unified Manager server and then add the upgraded version of the Unified Manager server to WFA.

4. In the New Data Source dialog box, select the required data source type, and enter a name for the data source and the host name.

Based on the selected data source type, the port, user name, password, and timeout fields might be automatically populated with the default data, if available. You can edit these entries as required.

5. Choose an appropriate action:

For...	Do this...
Active IQ Unified Manager 6.3 and later	<p>Enter the credentials of the Database User account that you created on the Unified Manager server. See <i>OnCommand Unified Manager Online Help</i> for details on creating a database user account.</p> <p> You must not provide the credentials of an Active IQ Unified Manager Database User account that was created using the command-line interface or the ocsetup tool.</p>

6. Click **Save**.
7. In the Data Sources table, select the data source, and click  on the toolbar.
8. Verify the status of the data acquisition process.

Add an upgraded Unified Manager server as a data source

If Unified Manager server (5.x or 6.x) is added as a data source to WFA and then the Unified Manager server is upgraded, you must add the upgraded Unified Manager server as a data source because the data that is associated with the upgraded version is not populated in WFA unless it is manually added as a data source.

Steps

1. Log into the WFA web GUI as an admin.
2. Click **Settings** and under **Setup**, click **Data Sources**.
3. Click  on the toolbar.
4. In the New Data Source dialog box, select the required data source type, and then enter a name for the data source and the host name.

Based on the selected data source type, the port, user name, password, and timeout fields might be automatically populated with the default data, if available. You can edit these entries as required.

5. Click **Save**.
6. Select the previous version of the Unified Manager server, and click  on the toolbar.
7. In the Delete Data Source Type confirmation dialog box, click **Yes**.
8. In the Data Sources table, select the data source, and then click  on the toolbar.
9. Verify the data acquisition status in the History table.

Create local users

OnCommand Workflow Automation (WFA) enables you to create and manage local WFA users with specific permissions for various roles, such as guest, operator, approver, architect, admin, and backup.

You must have installed WFA and logged in as an admin.

WFA enables you to create users for the following roles:

- **Guest**

This user can view the portal and the status of a workflow execution, and can be notified of a change in the status of a workflow execution.

- **Operator**

This user is allowed to preview and execute workflows for which the user is given access.

- **Approver**

This user is allowed to preview, execute, approve, and reject workflows for which the user is given access.



It is recommended to provide the email ID of the approver. If there are multiple approvers, you can provide a group email ID in the **E-mail** field.

- **Architect**

This user has full access to create workflows, but is restricted from modifying global WFA server settings.

- **Admin**

This user has complete access to the WFA server.

- **Backup**

This is the only user who can remotely generate backups of the WFA server. However, the user is restricted from all other access.

Steps

1. Click **Settings**, and under **Management** click **Users**.
2. Create a new user by clicking on the toolbar.
3. Enter the required information in the New User dialog box.
4. Click **Save**.

Configure the credentials of a target system

You can configure the credentials of a target system in OnCommand Workflow Automation (WFA) and use the credentials to connect to that specific system and execute commands.

After initial data acquisition, you must configure the credentials for the arrays on which the commands are run. PowerShell WFA controller connection works in two modes:

- With credentials

WFA tries to establish a connection using HTTPS first, and then tries using HTTP. You can also use Microsoft Active Directory LDAP authentication to connect to arrays without defining credentials in WFA. To use Active Directory LDAP, you must configure the array to perform authentication with the same Active Directory LDAP server.

- Without credentials (for storage systems operating in 7-Mode)

WFA tries to establish a connection using domain authentication. This mode uses the remote procedure call protocol, which is secured using the NTLM protocol.

- WFA checks the Secure Sockets Layer (SSL) certificate for ONTAP systems. Users might be prompted to review and accept/deny the connection to ONTAP systems if the SSL certificate is not trusted.
- You must reenter the credentials for ONTAP, NetApp Active IQ and Lightweight Directory Access Protocol (LDAP) after you restore a backup or complete an in-place upgrade.

Steps

- Log in to WFA through a web browser as an admin.
- Click **Settings**, and under **Setup** click **Credentials**.
- Click  on the toolbar.
- In the New Credentials dialog box, select one of the following options from the **Match** list:
 - Exact**
Credentials for a specific IP address or host name
 - Pattern**
Credentials for the entire subnet or IP range
- Select the remote system type from the **Type** list.
- Enter either the host name or the IPv4 or IPv6 address of the resource, the user name, and the password.



WFA 5.1 verifies the SSL certificates of all resources added to WFA. As certificate verification might prompt you to accept the certificates, using wildcards in credentials is not supported. If you have multiple clusters using the same credentials, you cannot add them all at once.

- Test the connectivity by performing the following action:

If you selected the following match type...	Then...
Exact	Click Test .
Pattern	<p>Save the credentials and choose one of the following:</p> <ul style="list-style-type: none"> Select the credential and click  on the toolbar. Right-click and select Test Connectivity.

- Click **Save**.

Configuring OnCommand Workflow Automation

OnCommand Workflow Automation (WFA) enables you to configure various settings—for example, AutoSupport and notifications.

When configuring WFA, you can set up one or more of the following, as required:

- AutoSupport for sending AutoSupport messages to technical support
- Microsoft Active Directory Lightweight Directory Access Protocol (LDAP) server for LDAP authentication and authorization for WFA users
- Mail for email notifications about workflow operations and sending AutoSupport messages
- Simple Network Management Protocol (SNMP) for notifications about workflow operations
- Syslog for remote data logging

Configure AutoSupport

You can configure several AutoSupport settings such as the schedule, content of the AutoSupport messages, and the proxy server. AutoSupport sends weekly logs of the content that you selected to technical support for archiving and issue analysis.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Setup** click **AutoSupport**.
3. Ensure that the **Enable AutoSupport** box is selected.
4. Enter the required information.
5. Select one of the following from the **Content** list:

If you want to include...	Then choose this option...
Only configuration details such as users, workflows, and commands of your WFA installation	send only configuration data
WFA configuration details and data in WFA cache tables such as the scheme	send configuration and cache data (default)
WFA configuration details, data in WFA cache tables, and data in the installation directory	send configuration and cache extended data



The password of any WFA user is *not* included in the AutoSupport data.

6. Test that you can download an AutoSupport message:
 - a. Click **Download**.
 - b. In the dialog box that opens, select the location to save the .7z file.
7. Test the sending of an AutoSupport message to the specified destination by clicking **Send Now**.
8. Click **Save**.

Configure authentication settings

You can configure OnCommand Workflow Automation (WFA) to use a Microsoft Active Directory (AD) Lightweight Directory Access Protocol (LDAP) server for authentication and authorization.

You must have configured a Microsoft AD LDAP server in your environment.

Only Microsoft AD LDAP authentication is supported for WFA. You cannot use any other LDAP authentication methods, including Microsoft AD Lightweight Directory Services (AD LDS) or Microsoft Global Catalog.



During communication, LDAP sends the user name and password in plain text. However, LDAPS (LDAP secure) communication is encrypted and secure.

Steps

1. Log in to WFA through a web browser as an admin.
2. Add a list of Active Directory group names to the required roles.



You can add a list of AD group names to the required roles in the Active Directory Groups Window.

[Active Directory Groups window](#)

3. Click **Administration > WFA Configuration**.
4. In the WFA Configuration dialog box, click the **Authentication** tab, and then select the **Enable Active Directory** check box.
5. Enter the required information in the fields:
 - a. If you want to use the user@domain format for domain users, replace sAMAccountName with userPrincipalName in the **User name attribute** field.
 - b. If unique values are required for your environment, edit the required fields.
6. Click **Add** to add the Active Directory in the Active Directory Servers table with a URI format:
`ldap://active_directory_server_address[:port]`
`ldap://NB-T01.example.com[:389]`

If you have enabled LDAP over SSL, you can use the following URI format:

`ldaps://active_directory_server_address[:port]`

7. Provide the credentials to bind the LDAP server and the base DN.
8. Test the authentication of the given user:

- a. Enter the user name and password.
- b. Click **Test Authentication**.



You must have added the Active Directory Group to test the authentication of the given user in WFA.

9. Click **Save**.

Add Active Directory groups

You can add Active Directory groups in OnCommand Workflow Automation (WFA).

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings** and under **Management**, click **Active Directory Groups**.
3. In the Active Directory Groups window, click the **New** icon.
4. In the New Active Directory Group dialog box, enter the required information.

If you select **Approver** from the **Role** drop down list, it is recommended provide the email ID of the approver. If there are multiple approvers, you can provide a group email ID in the **E-mail** field. Select the different events of the workflow for which the notification is to be sent to the particular Active Directory group.

5. Click **Save**.

Configure email notifications

You can configure OnCommand Workflow Automation (WFA) to send you email notifications about workflow operations—for example, workflow started or workflow failed.

You must have configured a mail host in your environment.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Setup** click **Mail**.
3. Enter the required information in the fields.
4. Test the mail settings by performing the following steps:
 - a. Click **Send test mail**.
 - b. In the Test Connection dialog box, enter the email address to which you want to send the email.
 - c. Click **Test**.
5. Click **Save**.

Configure SNMP

You can configure OnCommand Workflow Automation (WFA) to send Simple Network Management Protocol (SNMP) traps about the status of workflow operations.

WFA now supports SNMP v1 and SNMP v3 protocols. SNMP v3 provides additional security features.

The WFA .mib file provides information about the traps that are sent by the WFA server. The .mib file is located in the <WFA_install_location>\wfa\bin\wfa.mib directory on the WFA server.



The WFA server sends all the trap notifications with a generic object identifier (1.3.6.1.4.1.789.1.1.12.0).

You cannot use SNMP community strings such as community_string@SNMP_host for SNMP configuration.

Configure Syslog

You can configure OnCommand Workflow Automation (WFA) to send log data to a specific Syslog server for purposes such as event logging and log information analysis.

You must have configured the Syslog server to accept data from the WFA server.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Maintenance** click **Syslog**.
3. Select the **Enable Syslog** check box.
4. Enter the Syslog host name and select the Syslog log level.
5. Click **Save**.

Configure protocols for connecting to remote systems

You can configure the protocol used by OnCommand Workflow Automation (WFA) to connect to remote systems. You can configure the protocol based on your organization's security requirements and the protocol supported by the remote system.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Data Source Design > Remote System Types**.
3. Perform one of the following actions:

If you want to...	Do this...
Configure a protocol for a new remote system	<ol style="list-style-type: none">a. Click .b. In the New Remote System Type dialog box, specify the details such as name, description, and version.
Modify the protocol configuration of an existing remote system	<ol style="list-style-type: none">a. Select and double-click the remote system that you want to modify.b. Click .

4. From the Connection Protocol list, select one of the following:
 - HTTPS with fallback to HTTP (default)
 - HTTPS only
 - HTTP only
 - Custom
5. Specify the details for the protocol, default port, and default timeout.
6. Click **Save**.

Disable the default password policy

OnCommand Workflow Automation (WFA) is configured to enforce a password policy for local users. If you do not want to use the password policy, you can disable it.

You must have logged in to the WFA host system as a root user.

The default WFA installation path is used in this procedure. If you changed the default location during installation, you must use the changed WFA installation path.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server: WFA_install_location/wfa/bin/
2. Enter the following command:

```
./wfa --password-policy=none --restart=WFA
```

Modify the default password policy

OnCommand Workflow Automation (WFA) is configured to enforce a password policy for local users. You can modify the default password policy.

You must have logged in to the WFA host system as a root user.

- The default WFA installation path is used in this procedure.

If you changed the default location during installation, you must use the changed WFA installation path.

- The command for the default password policy is ./wfa --password-policy=default.

The default is

“minLength=true,8;specialChar=true,1;digitalChar=true,1;lowercaseChar=true,1;uppercaseChar=true,1;whitespaceChar=false” . This indicates that the default password policy must have a minimum length of eight characters, must contain at least 1 special character, 1 digit, 1 lowercase character, 1 uppercase character, and no spaces.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server: WFA_install_location/wfa/bin/
2. Modify the default password policy by entering the following command:

```
./wfa --password-policy=PasswordPolicyString --restart=WFA
```

Enable or disable remote access to the OnCommand Workflow Automation database

By default, the OnCommand Workflow Automation (WFA) database can be accessed only by clients running on the WFA host system. You can change the default settings if you want to enable access to the WFA database from a remote system.

- You must have logged in to the WFA host system as a root user.

- If a firewall is installed on the WFA host system, you must have configured your firewall settings to allow access to the MySQL port (3306) from the remote system.

The default WFA installation path is used in this procedure. If you changed the default location during installation, you must use the changed WFA installation path.

Steps

1. Navigate to the following directory on the WFA server: WFA_install_location/wfa/bin/.
2. Perform one of the following actions:

To...	Enter the following command...
Enable remote access	./wfa --db-access=public --restart
Disable remote access	./wfa --db-access=default --restart

Modify the transaction timeout setting of OnCommand Workflow Automation

The OnCommand Workflow Automation (WFA) database transaction times out in 300 seconds by default. You can increase the default timeout duration when restoring a large-sized WFA database from a backup to avoid potential failure of the database restoration.

You must have logged in to the WFA host system as a root user.

The default WFA installation path is used in this procedure. If you changed the default location during installation, you must use the changed WFA installation path.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server: WFA_install_location/wfa/bin/
2. Enter the following command:

```
./wfa --txn-timeout [=TIMEOUT] --restart=WFA
```

```
./wfa --txn-timeout=1000 --restart=WFA
```

Configure the timeout value for Workflow Automation

You can configure the timeout value for the Workflow Automation (WFA) web GUI, instead of using the default timeout value of 180 seconds.

The timeout value that you set is an absolute timeout rather than a timeout related to inactivity. For example, if you set this value to 30 minutes, then you are logged out after 30 minutes, even if you are active at the end of this time. You cannot set the timeout value from the WFA web GUI.

Steps

1. Log in as a root user on the WFA host machine.
2. Set the timeout value:

```
installdir bin/wfa -S=timeout value in minutes
```

Enabling ciphers and adding new ciphers

OnCommand Workflow Automation 5.1 supports a number of ciphers out of the box. Also, you can add additional ciphers as required.

The following ciphers can be enabled out of the box:

```
enabled-cipher-suites=
"TLS_DHE_DSS_WITH_AES_128_GCM_SHA256, TLS_DHE_DSS_WITH_AES_256_GCM_SHA384, TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256,
TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384, TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384,
TLS_RSA_WITH_AES_128_GCM_SHA256, TLS_RSA_WITH_AES_256_GCM_SHA384"
```

Additional ciphers can be added to this configuration in the `standalone-full.xml` file. This file is located at: `<installdir>/jboss/standalone/configuration/standalone-full.xml`.

The file can be modified to support additional ciphers as follows:

```
<https-listener name="https" socket-binding="https" max-post-size="1073741824" security-realm="SSLRealm"
enabled-cipher-suites="*/*< --- add additional ciphers here ---\>**"
enabled-protocols="TLSv1.1, TLSv1.2"/>
```

Upgrade from OnCommand Workflow Automation 3.1 or later

You can perform an in-place upgrade from OnCommand Workflow Automation (WFA) 3.1 or later to the latest available version of WFA to use the new features and enhancements.

You must have downloaded the .bin binary file from the NetApp Support Site to the WFA host machine.

You can restore to WFA 5.1 from either WFA 5.0 or 4.2 only. A WFA database backup can only be restored to a system that is running the same version or a later version of WFA.

The WFA 5.1 cluster connection needs to accept the SSL certificate. When updating from an earlier version of WFA to WFA 5.1, you need to certify the cluster connection. Save the cluster connection details for cluster certification after the in-place upgrade.

You cannot install MYSQL on your own when upgrading from earlier versions of WFA. You can install MySQL

on your own:

- When you are upgrading from WFA 4.2 to later versions of WFA.
- Upgrade from WFA 3.1 or later by choosing one of the following methods:
 - Interactive installation
 - i. Navigate to the .bin binary file in the WFA host machine, and then run the file.
 - ii. Follow the on-screen instructions to complete the upgrade.
 - Silent installation

At the shell prompt:

```
./WFA-version_number-build_number.bin [-y] [-u admin_user_name] [-p admin_user_password]
```

Example:

```
./WFA-3.1-Z3234343435.bin -y -u admin -p Company*234
```

During the silent upgrade, you must include values for all of the following command options:

- -y skips the installation confirmation.
- -u specifies the admin user name.
- -p specifies the admin user password.

If you have not provided the admin user password, then you must enter the password when prompted.

Pack identification during upgrade

During the upgrade process, OnCommand Workflow Automation (WFA) identifies and classifies the entities into a pack. If you had deleted any entity of a pack before the upgrade, the pack will not be identified during the upgrade.

During the upgrade process, WFA compares the packs in the database with the list of packs that were released in the Storage Automation Store to identify the packs that were installed before the upgrade. Pack identification thus classifies existing packs in the database.

WFA performs the following processes to identify and classify packs:

- Maintains a list of packs released in the Storage Automation Store to compare and identify the packs that were installed before the upgrade.
- Classifies the entities in a pack as part of the Storage Automation Store synchronization, if Storage Automation Store is enabled.
- Classifies the entities into packs using the updated list.

Pack identification is applicable only to NetApp-certified packs that were downloaded from the Storage Automation Store.

If a pack is not identified during upgrade, you can re-import the pack to get it identified in WFA. The wfa.log

files provide details about the entities that were not identified as a pack during the upgrade.

Upgrading third-party products

You can upgrade third-party products on Workflow Automation (WFA) such as OpenJDK and MySQL in Linux.

Upgrade OpenJDK

Oracle JRE is no longer supported in OnCommand Workflow Automation. In this release, OpenJDK replaces Oracle JRE for Linux. You can upgrade to a newer version of OpenJDK on the Linux server on which OnCommand Workflow Automation is installed to obtain fixes for security vulnerabilities.

You must have root privileges for the Linux system on which WFA is installed.

You can update OpenJDK releases within release families. For example, you can upgrade from OpenJDK 11.0.1 to OpenJDK 11.0.2, but you cannot update directly from OpenJDK 11 to OpenJDK 12.

Steps

1. Log in as a root user on the WFA host machine.
2. Install the latest version of OpenJDK 11 from the yum repository on the target system.
3. At the shell prompt, stop the WFA server.
4. Restart the WFA server.

Upgrade MySQL on Linux

You can upgrade to a newer version of MySQL on the Linux server on which OnCommand Workflow Automation is installed to obtain fixes for security vulnerabilities.

You must have root privileges for the Linux system on which WFA is installed.



Before you reinstall WFA 4.2, you must delete the MySQL data directory if you have uninstalled MySQL.

You can only upgrade to minor updates of MySQL 5.7, for example, 5.7.22 to 5.7.26. You cannot upgrade to major versions of MySQL, for example, version 5.8.

Steps

1. Log in as a root user on the WFA host machine.
2. Download the latest MySQL Community Server .rpm bundle on the target system.
3. Untar the bundle to a directory on the target system.
4. You will get multiple .rpm packages in the directory after untarring the bundle, but WFA only needs the following rpm packages:
 - mysql-community-client-5.7.x
 - mysql-community-libs-5.7.x

- mysql-community-server-5.7.x
- mysql-community-common-5.7.x
- mysql-community-libs-compat-5.7.x

Delete all other .rpm packages. Installing all packages in an rpm bundle will not cause any problems.

5. At the shell prompt, stop the WFA database and server services:

```
service wfa-db stop
```

```
service wfa-server stop
```

6. Invoke the upgrade of MySQL by using the following command:

```
rpm -uvh *.rpm
```

*.rpm refers to the .rpm packages in the directory where you downloaded the newer version of MySQL.

7. Start the WFA services:

```
service wfa-db start
```

```
service wfa-server start
```

Backing up the OnCommand Workflow Automation database

A backup of the OnCommand Workflow Automation (WFA) database includes the system configuration settings and cache information, including the playground database. You can use the backup for restoration purposes on the same system or on a different system.

An automatic backup of the database is created daily at 2 a.m. and is saved as a .zip file in the following location: wfa_install_location/WFA-Backups.

WFA saves up to five backups in the WFA-Backups directory, and replaces the oldest backup with the latest backup. The WFA-Backups directory is not deleted when you uninstall WFA. You can use the automatically created backup for restoration if you did not create a backup of the WFA database while uninstalling WFA.

You can also manually back up the WFA database when you have to save specific changes for restoration; for example, if you want to back up the changes that you have made before the automatic backup occurs.

- You can restore a WFA database backup only to a system that is running the same version or a later version of WFA.



For example, if you created a backup on a system that is running WFA 4.2, the backup can be restored only to systems that are running WFA 4.2 or later.

- You cannot use the web UI to back up the WFA database during disaster recovery in a high-availability setup.

Backup and restoration of user credentials

The backup of the WFA database includes the WFA user credentials.



The WFA database is also included in the AutoSupport data; however, the password of any WFA user is not included in the AutoSupport data.

When a WFA database is restored from a backup, the following items are preserved:

- The admin user credentials that were created during the current WFA installation.
- If a user with admin privileges other than the default admin user restores the database, the credentials of both the admin users.
- All other user credentials of the current WFA installation are replaced with the user credentials from the backup.

Back up the WFA database from the web portal

You can back up the OnCommand Workflow Automation (WFA) database from the web portal and use the backup file for data recovery purposes. You cannot perform a full backup from the web portal.

You must have admin or architect credentials to perform this task.

A WFA user with backup role cannot log in to the web portal to perform a backup. The WFA users with backup role can only perform remote or scripted backups.

Steps

1. Log in to the WFA web GUI as an admin.
2. Click **Settings** and under **Maintenance**, click **Backup & Restore**.
3. Click **Backup**.
4. In the dialog box that opens, select a location, and then save the file.

Backing up the WFA database using the CLI

If you want to back up the OnCommand Workflow Automation (WFA) database frequently, you can use the WFA command-line interface (CLI) provided with the WFA installation package.

The following are the two backup types:

- Full backup
- Regular backup

Back up (full) the WFA database using the CLI

You can perform a full backup of the OnCommand Workflow Automation (WFA) database by using the WFA command-line interface (CLI). In a full backup, the WFA database, WFA configuration, and key are backed up.

You must have admin user credentials or architect credentials.

In a high-availability environment, you should create scheduled backups by using REST APIs. You cannot create backups by using the CLI when WFA is in failover mode.

For more information, see the REST documentation.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server:

`WFA_install_location/wfa/bin/`

`WFA_install_location` is the WFA installation directory.

2. Back up the WFA database: `.\wfa --backup --user=USER [--password=PASS] [--location=PATH] [--full]`

- user is the user name of the backup user.
- password is the password of the backup user.

If you have not provided the password, you must enter the password when prompted.

- path is the complete directory path to the backup file.

3. Verify that the backup file was created at the specified location.

Back up (regular) the WFA database using the CLI

You can perform a regular backup of the OnCommand Workflow Automation (WFA) database by using the WFA command-line interface (CLI). In a regular backup, only the WFA database is backed up.

You must have admin user credentials, architect credentials, or backup user credentials.

In a high-availability environment, you should create scheduled backups by using REST APIs. You cannot create backups by using the CLI when WFA is in failover mode.

For more information, see the REST documentation.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server: `WFA_install_location/wfa/bin/`.

`WFA_install_location` is the WFA installation directory.

2. Back up the WFA database: `.\wfa --backup --user=USER [--password=PASS] [--location=PATH]`

- user is the user name of the backup user.
- password is the password of the backup user.

If you have not provided the password, you must enter the password when prompted.

- path is the complete directory path to the backup file.

3. Verify that the backup file was created at the specified location.

Backing up the WFA database using REST APIs

You can back up the OnCommand Workflow Automation (WFA) database by using the REST APIs. If WFA is in the failover mode in a high-availability environment, you can use the REST APIs to create scheduled backups. You cannot use the command-line interface (CLI) to create backups during a failover.

The following are the two types of backup:

- Full backup
- Regular backup

Perform a full backup of the WFA database using REST APIs

You can perform a full back up of the OnCommand Workflow Automation (WFA) database by using the REST APIs. In a full backup, the WFA database, WFA configuration, and key are backed up.

You must have admin or architect credentials.

Step

1. Enter the following URL in your web browser: `https://IP address of the WFA server/rest/backups?full=true`

For more information, see the REST documentation.

Perform a regular backup of the WFA database using REST APIs

You can perform a regular backup of the OnCommand Workflow Automation (WFA) database by using the REST APIs. In a regular backup, only the WFA database is backed up.

You must have admin, architect, or backup credentials.

Step

1. Enter the following URL in your web browser: `https://IP address of the WFA server/rest/backups`

For more information, see the REST documentation.

Restoring the OnCommand Workflow Automation database

Restoring the OnCommand Workflow Automation (WFA) database includes restoring the system configuration settings and cache information, including the playground database.

- Restoring a WFA database erases the current WFA database.
- You can restore a WFA database backup only to a system that is running the same version or a later version of WFA.

For example, if you created a backup on a system that is running WFA 4.2, the backup can be restored only to systems that are running WFA 4.2 or later.

- After the restore operation is complete, the WFA SSL certificate is replaced with the SSL certificate in the backup file.

- A comprehensive restore operation of WFA databases and configurations is required during disaster recovery, and can be used in both standalone and high-availability environments.



- A comprehensive backup cannot be created by using the web UI.

You can use only the CLI commands or REST APIs to backup and restore the WFA database comprehensively during disaster recovery.

Restore the WFA database

You can restore the OnCommand Workflow Automation (WFA) database that you backed up previously.

- You must have created a backup of the WFA database.
- You must have admin or architect credentials.
- Restoring a WFA database erases the current database.
- You can restore a WFA database backup only to a system running the same or a later version of OnCommand Workflow Automation.

For example, if you created a backup on a system running OnCommand Workflow Automation 4.2, the backup can be restored only to systems running OnCommand Workflow Automation 4.2 or later.

Steps

1. Log in to the WFA web GUI as an admin.
2. Click **Settings** and under **Maintenance**, click **Backup & Restore**.
3. Click **Choose file**.
4. In the dialog box that opens, select the WFA backup file, and click **Open**.
5. Click **Restore**.

You can review the restored content for completeness in functionality—for example, the functioning of your custom workflows.

Restoring the WFA database using the CLI

During a disaster, while recovering data you can restore the OnCommand Workflow Automation (WFA) database and supported configurations that you backed up previously using the command-line interface (CLI). The supported configurations include data access, HTTP timeout, and SSL certificates.

The following are the two types of restore:

- Full restore

- Regular restore

Restore (full) the WFA database using the CLI

You can do a full restore of the OnCommand Workflow Automation (WFA) database and supported configurations that you backed up previously by using the command-line interface (CLI). In a full restore, you can restore the WFA database, WFA configuration, and key.

- You must have created a backup of the WFA database.
- You must have admin or architect credentials.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server: WFA_install_location/wfa/bin

wfa_install_location is the WFA installation directory.

2. Restore the WFA database:

```
wfa --restore --full --user=user_name [--password=password] [--location=path]
--restart
```

- user_name is the user name of the admin or architect user.
- password is the password of the user.

If you have not provided the password, you must enter the password when prompted.

- path is the complete directory path to the restore file.

3. Verify that the restore operation is successful and WFA is accessible.

Restore (regular) the WFA database using the CLI

You can do a regular restore of the OnCommand Workflow Automation (WFA) database that you backed up previously by using the command-line interface (CLI). In a regular restore, you can only restore the WFA database.

- You must have created a backup of the WFA database.
- You must have admin or architect credentials.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server: WFA_install_location/wfa/bin

wfa_install_location is the WFA installation directory.

2. Restore the WFA database:

```
wfa --restore --user=user_name [--password=password] [--location=path]
```

- user_name is the user name of the admin or architect user.
- password is the password of the user.

If you have not provided the password, you must enter the password when prompted.

- path is the complete directory path to the restore file.

3. Verify that the restore operation is successful and WFA is accessible.

Restoring the WFA database using REST APIs

You can restore the OnCommand Workflow Automation (WFA) database by using REST APIs. You cannot use the command-line interface (CLI) to restore the WFA database during a failover.

The following are the two types of restore:

- Full restore
- Regular restore

Restore (full) the WFA database using REST APIs

You can do a full restore of the OnCommand Workflow Automation (WFA) database by using REST APIs. In a full restore, you can restore the WFA database, WFA configuration, and key.

- You must have created a .zip backup of the WFA database.
- You must have admin or architect credentials.
- If you are restoring the database as a part of the migration procedure, you must do a full restore.

Steps

1. Enter the following URL in the REST client browser: `https://IP address of WFA server/rest/backups?full=true`
2. In the Backup window, select the **POST** method.
3. In the **Part** drop-down list, select **Multipart Body**.
4. In the **File** field, enter the following information:
 - a. In the **Content type** drop-down list, select **multi-part/form-data**.
 - b. In the **Charset** drop-down list, select **ISO-8859-1**.
 - c. In the **File name** field, enter the name of the backup file you created and that you want to restore.
 - d. Click **Browse**.
 - e. Select the location of the .zip backup file.
5. Navigate to the `/opt/netapp/wfa/bin` directory, and restart the WFA services:
6. Restart the **NetApp WFA Database** and **NetApp WFA Server** service:

```
wfa --restart
```
7. Verify that the restore operation is successful and WFA is accessible.

Restore (regular) the WFA database using REST APIs

You can do a regular restore of the OnCommand Workflow Automation (WFA) database by using REST APIs. In a regular restore, you can only restore the WFA database.

- You must have created a .zip backup of the WFA database.
- You must have admin or architect credentials.
- If you are restoring the database as a part of the migration procedure, you must do a full restore.

Steps

1. Enter the following URL in the REST client browser: `https://IP address of WFA server/rest/backups`
2. In the Backup window, select the **POST** method.
3. In the **Part** drop-down list, select **Multipart Body**.
4. In the **File** field, enter the following information:
 - a. In the **Content type** drop-down list, select **multi-part/form-data**.
 - b. In the **Charset** drop-down list, select **ISO-8859-1**.
 - c. In the **File name** field, enter the name of the backup file as `backupFile`.
 - d. Click **Browse**.
 - e. Select the location of the .zip backup file.
5. Navigate to the `/opt/netapp/wfa/bin` directory, and restart the WFA services.
6. Verify that the restore operation is successful and WFA is accessible.

Reset the admin password created during installation

If you have forgotten the password of the admin user that you created when installing the OnCommand Workflow Automation (WFA) server, you can reset it.

- You must have root privileges for the Linux system on which you have installed WFA.
- The WFA services must be running.
- This procedure resets only the password of the admin user created during the WFA installation.

You cannot reset the password of other WFA admin users that you created after the WFA installation.

- This procedure does not enforce the password policy you have configured.

Therefore, you must enter a password that complies with your password policy or change the password from the WFA user interface after you have reset the password.

Steps

1. As a root user, log in to the Linux system on which WFA is installed.
2. At the shell prompt, navigate to the following directory on the WFA server: `WFA_install_location/wfa/bin`
3. Enter the following command:

```
./wfa --admin-password [--password=PASS]
```

If you have not provided a password, you must enter the password when prompted.

4. At the shell prompt, follow the on-screen instructions.

Import OnCommand Workflow Automation content

You can import user-created OnCommand Workflow Automation (WFA) content such as workflows, finders, and commands. You can also import content that is exported from another WFA installation, content that is downloaded from the Storage Automation Store or the WFA community, as well as packs, including Data ONTAP PowerShell toolkits and Perl NMSDK toolkits.

- You must have access to the WFA content that you want to import.
- The content that you want to import must have been created on a system that is running the same version or an earlier version of WFA.

For example, if you are running WFA 2.2, you cannot import content that was created using WFA 3.0.

- You can import content developed on N-2 versions of WFA only into WFA 5.1.
- If the .dar file references NetApp-certified content, the NetApp-certified content packs must be imported.

The NetApp-certified content packs can be downloaded from the Storage Automation Store. You must refer to the documentation of the pack to verify that all requirements are met.

Steps

1. Log in to WFA through a web browser.
2. Click **Settings**, and under **Maintenance** click **Import Workflows**.
3. Click **Choose File** to select the .dar file that you want to import, and then click **Import**.
4. In the Import Success dialog box, click **OK**.

Related information

[NetApp community: OnCommand Workflow Automation](#)

Considerations while importing OnCommand Workflow Automation content

You must be aware of certain considerations when you import user-created content, content that is exported from another OnCommand Workflow Automation (WFA) installation, or content that is downloaded from the Storage Automation Store or the WFA community.

- WFA content is saved as a .dar file and can include the entire user-created content from another system or specific items such as workflows, finders, commands, and dictionary terms.
- When an existing category is imported from a .dar file, the imported content is merged with the existing content in the category.

For example, consider there are two workflows WF1 and WF2 in category A in the WFA server. If workflows WF3 and WF4 in category A are imported to the WFA server, category A will contain workflows WF1, WF2, WF3, and WF4 after the import.

- If the .dar file contains dictionary entries, then the cache tables corresponding to the dictionary entries are automatically updated.

If the cache tables are not updated automatically, an error message is logged in the wfa.log file.

- When importing a .dar file that has a dependency on a pack that is not present in the WFA server, WFA tries to identify whether all the dependencies on the entities are met.

- If one or more entities are missing or if a lower version of an entity is found, the import fails and an error message is displayed.

The error message provides details of the packs that should be installed in order to meet the dependencies.

- If a higher version of an entity is found or if the certification has changed, a generic dialog box about the version mismatch is displayed, and the import is completed.

The version mismatch details are logged in a wfa.log file.

- Questions and support requests for the following must be directed to the WFA community:

- Any content downloaded from the WFA community
 - Custom WFA content that you have created
 - WFA content that you have modified

Migrate the OnCommand Workflow Automation installation

You can migrate an OnCommand Workflow Automation (WFA) installation to maintain the unique WFA database key that is installed during the WFA installation.

- You must perform this procedure only when you want to migrate a WFA installation that includes the WFA database key to a different server.
- A WFA database restore does not migrate the WFA key.
- Migrating a WFA installation does not migrate the SSL certificates.
- The default WFA installation path is used in this procedure.

If you changed the default location during installation, you must use the changed WFA installation path.

Steps

1. Access WFA through a web browser as an admin.
2. Back up the WFA database.
3. Open a shell prompt on the WFA server and change directories to the following location:
WFA_install_location/wfa/bin/
4. Enter the following at the shell prompt to obtain the database key:

```
./wfa -key
```

5. Note the database key that is displayed.
6. Uninstall WFA.

7. Install WFA on the required system.
8. Open a shell prompt on the WFA server and change the directories to the following location:
WFA_install_location/wfa/bin/
9. At the shell prompt, install the database key by entering the following command:

```
./wfa -key=yourdatabasekey
```

yourdatabasekey is the key that you noted from the previous WFA installation.

10. Restore the WFA database from the backup that you created.

Uninstall OnCommand Workflow Automation

You can uninstall OnCommand Workflow Automation (WFA) from your Red Hat Enterprise Linux machine using a single command.

You must have root user access to the Red Hat Enterprise Linux machine from which want to uninstall WFA.

Steps

1. Log in as root user to the Red Hat Enterprise Linux machine from which you want to uninstall WFA.
2. At the shell prompt, enter the following command:

```
rpm -e wfa
```

If the default installation location was changed, the MySQL data directory is not deleted when you uninstall WFA. You must manually delete the directory.

Managing OnCommand Workflow Automation SSL certificate

You can replace the default OnCommand Workflow Automation (WFA) SSL certificate with a self-signed certificate or a certificate signed by a Certificate Authority (CA).

The default self-signed WFA SSL certificate is generated during the installation of WFA. When you are upgrading, the certificate for the previous installation is replaced with the new certificate. If you are using a non-default self-signed certificate or a certificate signed by a CA, you must replace the default WFA SSL certificate with your certificate.

Replace the default Workflow Automation SSL certificate

You can replace the default Workflow Automation (WFA) SSL certificate if the certificate has expired or if you want to increase the validity period of the certificate.

You must have root privileges for the Linux system on which you have installed WFA.

The default WFA installation path is used in this procedure. If you changed the default location during installation, you must use the custom WFA installation path.

Steps

1. Log in as a root user on the WFA host machine.
2. At the shell prompt, navigate to the following directory on the WFA server: WFA_install_location/wfa/bin
3. Stop the WFA database and server services:

```
./wfa --stop=WFA
```

```
./wfa --stop=DB
```

4. Delete the wfa.keystore file from the following location:
WFA_install_location/wfa/jboss/standalone/configuration/keystore.
5. Open a shell prompt on the WFA server, and then change directories to the following location:
<OpenJDK_install_location>/bin
6. Obtain the database key:

```
keytool -keysize 2048 -genkey -alias "ssl keystore" -keyalg RSA -keystore  
"WFA_install_location/wfa/jboss/standalone/configuration/keystore/wfa.keystore  
" -validity xxxx
```

xxxx is the number of days for the validity of the new certificate.

7. When prompted, provide the password (default or new).

The default password is a randomly generated encrypted password.

To obtain and decrypt the default password, follow the steps in the Knowledge Base article [How to renew the self-signed certificate on WFA 5.1.1.0.4](#)

To use a new password, follow the steps in the Knowledge Base article [How to update a new password for the keystore in WFA](#).

8. Enter the required details for the certificate.
9. Review the displayed information, and then enter Yes.
10. Press **Enter** when prompted by the following message: Enter key password for <SSL keystore> <RETURN if same as keystore password>.
11. Restart the WFA services:

```
./wfa --start=DB
```

```
./wfa --start=WFA
```

Create a certificate signing request for Workflow Automation

You can create a certificate signing request (CSR) in Linux so that you can use the SSL certificate that is signed by a Certificate Authority (CA) instead of the default SSL certificate for Workflow Automation (WFA).

- You must have root privileges for the Linux system on which you have installed WFA.
- You must have replaced the default SSL certificate that is provided by WFA.

The default WFA installation path is used in this procedure. If you have changed the default path during installation, then you must use the custom WFA installation path.

Steps

1. Log in as a root user on the WFA host machine.
2. Open a shell prompt on the WFA server, and then change directories to the following location:
<OpenJDK_install_location>/bin
3. Create a CSR file:

```
keytool -certreq -keystore  
WFA_install_location/wfa/jboss/standalone/configuration/keystore/wfa.keystore  
-alias "ssl keystore" -file /root/file_name.csr
```

file_name is the name of the CSR file.

4. When prompted, provide the password (default or new).

The default password is a randomly generated encrypted password.

To obtain and decrypt the default password, follow the steps in the Knowledge Base article [How to renew the self-signed certificate on WFA 5.1.1.0.4](#)

To use a new password, follow the steps in the Knowledge Base article [How to update a new password for the keystore in WFA](#).

5. Send the file_name.csr file to the CA to obtain a signed certificate.

See the CA web site for details.

6. Download a chain certificate from the CA, and then import the chain certificate to your keystore:

```
keytool -import -alias "ssl keystore CA certificate" -keystore  
WFA_install_location/wfa/jboss/standalone/configuration/keystore/wfa.keystore"  
-trustcacerts -file chain_cert.cer
```

chain_cert.cer is the chain certificate file that is received from the CA. The file must be in the X.509 format.

7. Import the signed certificate that you received from the CA:

```
keytool -import -alias "ssl keystore" -keystore  
WFA_install_location/wfa/jboss/standalone/configuration/keystore/wfa.keystore"  
-trustcacerts -file certificate.cer
```

certificate.cer is the chain certificate file that is received from the CA.

8. Start the WFA services:

```
./wfa --start=DB
```

```
./wfa --start=WFA
```

Managing Perl and Perl modules

OnCommand Workflow Automation (WFA) supports Perl commands for workflow operations. You can install and configure your preferred Perl distribution and Perl modules.

The required Perl modules from the NetApp Manageability SDK are also installed when you install WFA. The NetApp Manageability SDK Perl modules are required for successful execution of Perl commands.

You can install additional Perl modules, if required, from the Red Hat package repositories or from the CPAN repositories.

Configure your preferred Perl distribution

The Perl package installed on your system is used by OnCommand Workflow Automation. If you want to use another Perl distribution, you can configure your preferred Perl distribution to work with WFA.

You must have installed the required Perl distribution on the WFA server.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server: WFA_install_location/wfa/bin/
2. Enter the following command:

```
/wfa --custom-perl [=PERL_PATH] --restart=WFA  
/wfa --custom-perl=/usr/local/perl5-11/bin/perl --restart=WFA
```

Troubleshooting installation and configuration issues

You can troubleshoot issues that might occur while installing and configuring OnCommand Workflow Automation (WFA).

Cannot view Performance Advisor data in WFA

If you cannot view Performance Advisor data in WFA or if the data acquisition process from the Performance Advisor data source fails, you should perform certain actions to troubleshoot the issue.

- Ensure that you have specified the credentials of an Active IQ Unified Manager user with a minimum role of GlobalRead when configuring Performance Advisor as a data source in WFA.
- Ensure that you have specified the correct port when configuring Performance Advisor as a data source in WFA.

By default, Active IQ Unified Manager uses port 8088 for an HTTP connection and port 8488 for an HTTPS connection.

- Ensure that performance data is collected by the Active IQ Unified Manager server.

Related documentation for OnCommand Workflow Automation

There are additional documents and tools to help you learn to perform more advanced configuration of your OnCommand Workflow Automation (WFA) server.

Other references

The Workflow Automation space within the NetApp community provides additional learning resources, including the following:

- **NetApp community**

[NetApp community: Workflow Automation \(WFA\)](#)

Tool references

- **Interoperability Matrix**

Lists supported combinations of hardware components and software versions.

[Interoperability Matrix](#)

Installation and setup for Windows

Overview of OnCommand Workflow Automation

OnCommand Workflow Automation (WFA) is a software solution that helps to automate storage management tasks, such as provisioning, migration, decommissioning, data protection configurations, and cloning storage. You can use WFA to build workflows to complete tasks that are specified by your processes. WFA supports both ONTAP and Data ONTAP operating in 7-Mode.

A workflow is a repetitive and procedural task that consists of sequential steps, including the following types of tasks:

- Provisioning, migrating, or decommissioning storage for databases or file systems
- Setting up a new virtualization environment, including storage switches and datastores
- Setting up storage for an application as part of an end-to-end orchestration process

Storage architects can define workflows to follow best practices and meet organizational requirements, such as the following:

- Using required naming conventions
- Setting unique options for storage objects
- Selecting resources
- Integrating internal configuration management database (CMDB) and ticketing applications

WFA features

- Workflow design portal to build workflows

The workflow design portal includes several building blocks, such as commands, templates, finders, filters, and functions, that are used to create workflows. The designer enables you to include advanced capabilities to workflows such as automated resource selection, row repetition (looping), and approval points.

The workflow design portal also includes building blocks, such as dictionary entries, cache queries, and data source types, for caching data from external systems.

- Execution portal to execute workflows, verify status of workflow execution, and access logs
- Administration/Settings option for tasks such as setting up WFA, connecting to data sources, and configuring user credentials
- Web service interfaces to invoke workflows from external portals and data center orchestration software
- Storage Automation Store to download WFA packs. The ONTAP 9.7.0 pack is bundled with WFA 5.1.

WFA license information

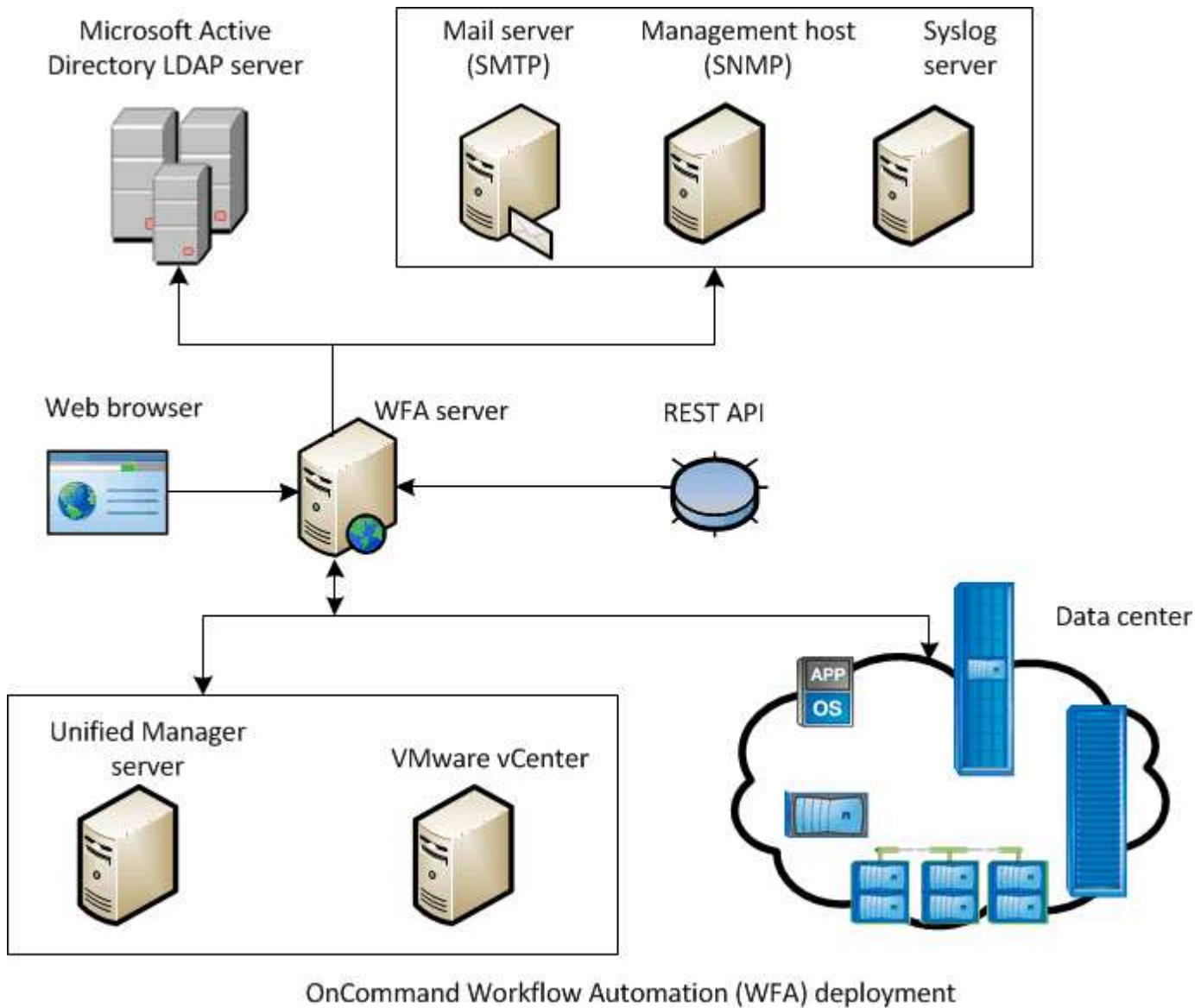
No license is required for using the OnCommand Workflow Automation server.

OnCommand Workflow Automation deployment architecture

OnCommand Workflow Automation (WFA) server is installed to orchestrate the workflow operations across several datacenters.

You can centrally manage your automation environment by connecting your WFA server to several Active IQ Unified Manager deployments and VMware vCenters.

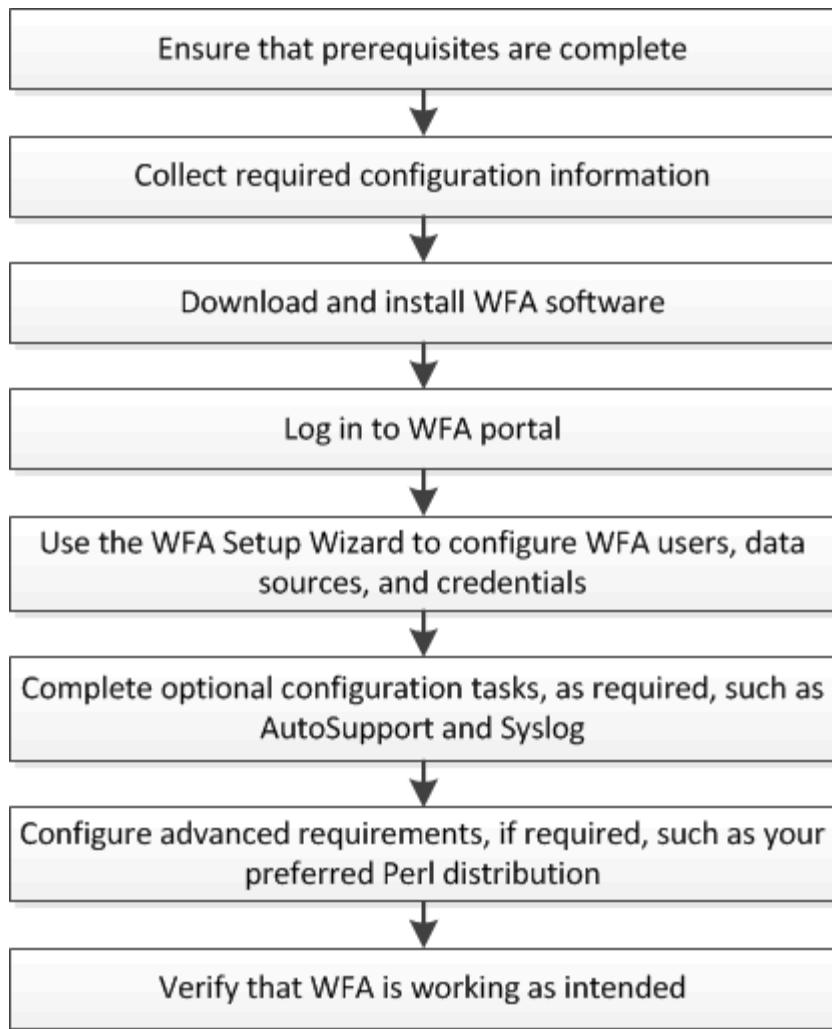
The following illustration shows a deployment example:



OnCommand Workflow Automation installation and setup overview

Installing OnCommand Workflow Automation (WFA) includes performing tasks such as preparing for the installation, downloading the WFA installer, and running the installer. After the installation is complete, you can configure WFA to meet your requirements.

The following flowchart illustrates the installation and configuration tasks:



Known limitations and enhancements

OnCommand Workflow Automation (WFA) 5.1 includes some limitations and unsupported features that you should be aware of before you install and configure WFA.

- **Creating category names**

- When a hyphen (-) is used in a category name, it is replaced with a space once the category is saved. For example, if the category name "abc-xyz" is provided, the category name is saved as "abc xyz", with the hyphen removed. To avoid this issue, do not use hyphens in category names.
- When a colon (:) is used in a category name, the text string before the colon is ignored once the category is saved. For example, if the category name "abc : xyz" is provided, the category name is saved as "xyz", with the "abc" string removed. To avoid this issue, do not use colons in category names.
- There is no check to prevent two categories having the same name. However, this causes a problem when selecting these categories from the navigation pane. To avoid this issue, make sure that each category name is unique.

System requirements for installing OnCommand Workflow Automation

You must be aware of the OnCommand Workflow Automation (WFA) hardware and software requirements before installing WFA.

Hardware requirements for installing WFA

The following table lists the minimum hardware requirements and the recommended hardware specifications for the WFA server.

Component	Minimum requirements	Recommended specifications
CPU	2.27 GHz or faster, 4 core, 64-bit	2.27 GHz or faster, 4 core, 64-bit
RAM	4 GB	8 GB
Free disk space	5 GB	20 GB

If you are installing WFA on a virtual machine (VM), you should reserve the required memory and CPU so that the VM has sufficient resources. The installer does not verify the CPU speed.

Software requirements for installing WFA

WFA runs on a 64-bit Windows operating system, and should be installed on dedicated physical machines or VMs. You must not install any other application on the server that runs WFA.

WFA runs from Microsoft Windows Server 2012 Enterprise Edition to Microsoft Windows Server 2016 (all editions). Enterprise Edition is the recommended Windows operating system.

For Windows 2012 servers, .NET Framework version 4.5.2 must be installed on your Windows system. If .NET Framework version 4.5.2 is not installed, WFA 5.1 installation fails.

- One of the following supported browsers:
 - Mozilla Firefox
 - Microsoft Internet Explorer
 - Google Chrome
- PowerShell 3.0
- VMware PowerCLI version 5



The PowerShell extension for VMware APIs is required only if you are using WFA to execute workflows on VMware vSphere.



Antivirus applications might prevent WFA services from starting.

To avoid this issue, configure antivirus scanning exclusions for the following WFA directories:

- The directory where you have installed WFA
- The directory where you have installed Perl
- The directory where you have installed OpenJDK
- The MySQL Data Directory

For more details, see the Interoperability Matrix Tool.

Related information

[NetApp Interoperability Matrix Tool](#)

Ports required for Workflow Automation

If you are using a firewall, you must be aware of the required ports for Workflow Automation (WFA).

The default port numbers are listed in this section. If you want to use a non-default port number, you must open that port for communication. For more details, see the documentation on your firewall.

The following table lists the default ports that should be open on the WFA server:

Port	Protocol	Direction	Purpose
80, 443	HTTP, HTTPS	Incoming	Opening WFA and logging in
80, 443, 22	HTTP, HTTPS, SSH	Outgoing	Command execution (ZAPI, PowerCLI)
445, 139, 389, 636	Microsoft-DS, NetBIOS-ssn, AD LDAP, AD LDAPS	Outgoing	Microsoft Active Directory LDAP authentication
161	SNMP	Outgoing	Sending SNMP messages on the status of workflows
3306	MySQL	Incoming	Caching read-only user
25	SMTP	Outgoing	Mail notification
80, 443, 25	HTTP, HTTPS, SMTP	Outgoing	Sending AutoSupport messages
514	Syslog	Outgoing	Sending logs to a syslog server

The following table lists the default ports that should be open on the Unified Manager server:

Port	Protocol	Direction	Purpose
3306	MySQL	Incoming	Caching data from Active IQ Unified Manager 6.0 and later

The following table lists the default port that should be open on the VMware vCenter:

Port	Protocol	Direction	Purpose
443	HTTPS	Incoming	Caching data from VMware vCenter

The following table lists the default port that should be open on the SNMP host machine:

Port	Protocol	Direction	Purpose
162	SNMP	Incoming	Receiving SNMP messages on the status of workflows

Prerequisites for installing Workflow Automation

Before installing OnCommand Workflow Automation (WFA), you must ensure that you have the required information and you have completed certain tasks.

Before you install WFA on a system, you must have completed the following tasks:

- Downloading the WFA installation file from the NetApp Support Site and copying the file to the server on which you want to install WFA



You must have valid credentials to log in to the NetApp Support Site. If you do not have valid credentials, you can register on the NetApp Support Site to obtain the credentials.

- Verifying that the system has access to the following, as applicable:

- Storage controllers
- Active IQ Unified Manager
- VMware vCenter



If your environment requires Secure Shell (SSH) accessibility, you must ensure that SSH is enabled on the target controllers.

- Verifying that PowerShell 3.0 or later is installed
- Ensuring that VMware Power CLI is installed, if you are using WFA to execute workflows on VMware vSphere
- Collecting the required configuration information
- Ensuring that the MySQL .Net connector is installed, if you are using the Invoke-MysqlQuery cmdlet

Required configuration information

Unit or system	Details	Purpose
Arrays	<ul style="list-style-type: none"> IP address User name and password 	<p>Perform operations on storage systems</p> <p> Root or admin account credentials are required for storage (arrays).</p>
vSphere	<ul style="list-style-type: none"> IP address User name and password of an admin for vCenter server 	<p>Acquire data Perform operations by using VMware APIs</p> <p> You must have installed VMware Power CLI.</p>
External repositories such as OnCommand Balance and custom databases	<ul style="list-style-type: none"> IP address User name and password of a read-only user account 	<p>Acquire data You must create the relevant WFA content, such as dictionary entries and cache queries for the external repositories, in order to acquire data from the external repositories.</p>
Mail server	<ul style="list-style-type: none"> IP address User name and password <p> User name and password are required if your mail server requires authentication.</p>	Receive WFA notifications through email
AutoSupport server	<ul style="list-style-type: none"> Mail host 	<p>Send AutoSupport messages through SMTPIf you do not have a mail host configured, you can use HTTP or HTTPS to send AutoSupport messages.</p>
Microsoft Active Directory (AD) LDAP server	<ul style="list-style-type: none"> IP address User name and password Group name 	Authenticate and authorize using AD LDAP or AD LDAPS
SNMP management application	<ul style="list-style-type: none"> IP address Port 	Receive WFA SNMP notifications

Unit or system	Details	Purpose
Syslog server	<ul style="list-style-type: none"> IP address 	Send log data

Related information

[NetApp Support](#)

Managing high availability

You can configure a high-availability setup to provide constant support for network operations. If one of the components fail, the mirrored component in the setup takes over the operation and provides uninterrupted network resources. You can also back up the WFA database and supported configurations so that you can recover the data in case of a disaster.

Set up Workflow Automation in MSCS for high availability

You can install and configure Workflow Automation (WFA) in a Microsoft Cluster Service (MSCS) environment to set up high availability and provide failover. Before you install WFA, you must verify that all the required components are configured correctly.

A high-availability setup provides constant support for application operations. If one of the components fails, the mirrored component in the setup takes over the operation and provides uninterrupted network resources.



MSCS is the only clustering solution that is supported by WFA in Windows.

Configure MSCS to install Workflow Automation

Before you install Workflow Automation (WFA) in Microsoft Cluster Server (MSCS), you must configure your MSCS environment.

- MSCS must be installed from the server manager.
- Optional: SnapDrive for Windows must be installed.

The minimum supported version is Windows 2012.

- The same version of WFA must be installed using the same path on both the cluster nodes.
- Both the cluster nodes must be added to the same domain.

You must complete this task by using Cluster Manager in the MSCS interface.

Steps

- Log in to Cluster Manager as a domain admin.
- Verify that the LUNs are accessible to both the nodes using one of the following options:
 - Managing the LUNs natively.
 - By using SnapDrive for Windows:

- i. Install and configure SnapDrive for Windows on both the nodes.
- ii. Create a LUN using SnapDrive for Windows and configure the LUN for both the nodes.

3. From Failover Cluster Manager, add the disk to the cluster.

Install OnCommand Workflow Automation on Windows

You can install OnCommand Workflow Automation (WFA) to create and customize storage workflows for automating the storage tasks that are performed in your environment.

- You must have reviewed the installation prerequisites.

[Prerequisites for installing Workflow Automation](#)

- If you are installing WFA on a system where WFA was previously installed and then uninstalled, you must ensure that there are no WFA services on that system.
- You must have downloaded the WFA installer from the NetApp Support Site.
- If you are installing WFA on a virtual machine (VM), the name of the VM must not include the underscore (_) character.
- ActiveState ActivePerl is installed before you install WFA.

This installation does not affect any other instances of ActivePerl that you have installed on your WFA server.

- Before you reinstall WFA 4.2 or later, you must delete MySQL data directory if you have uninstalled MySQL.

Steps

1. Log in to Windows using an account with administrative permissions.
2. Open Windows Explorer, and then navigate to the directory where the installation file is located.
3. Install WFA:
 - Interactive installation
 - i. Right-click and run the WFA installer executable (.exe) file as an admin user.
 - ii. Click **Next**.
 - iii. Enter the credentials for the default admin user, and then click **Next**.

The default admin password must satisfy the following criteria:

- Minimum of eight characters
- One uppercase character
- One lowercase character
- One numeral
- One special character
- The following special characters are not supported in a password and cause installation failure:
`" ; < > , = & ^ |



You must note the credentials of the admin user.

- iv. Enter a user name and password for the WFA service logon. For a domain user provide a user name in the format DOMAIN\USER. For a local system user the format is just a user name. The default user name is "wfa".

WFA installer creates a local user if one does not exist. If a local user exists and the password entered is different than the existing password, WFA updates the password.



Ensure the password conforms to the password policy configured for local users on the system. If the password does not conform to the password policy, the install fails.

- v. Select the ports for the WFA configuration, and then click **Next**.

- vi. Enter a site name and your company name, and then click **Next**.

The site name can include the location of the WFA installation, for example, Pittsburgh, PA.

- vii. If you want to change the default installation location, select the location where you want to install WFA, and then click **Next**.

- viii. If you want to change the default installation location for third party products, select the location where you want to install third party products, and then click **Next**.

- ix. If you do not want to change the default location of the WFA database, click **Next**.

- x. Click **Install** to continue the installation.

- xi. Click **Finish** to complete the installation.

- xii. Verify that WFA was installed successfully by choosing one of the following actions:

- Access WFA through a web browser.
- Use the Windows Services console to verify that the NetApp WFA Server service and the NetApp WFA Database service are running.

- Silent installation (from the command prompt):

```
WFA-version_number-build_number.exe /s
/v"WFA_ADMIN_USERNAME=wfa_username WFA_ADMIN_PASSWORD=password
WFA_ADMIN_CONFIRM_PASSWORD=confirm admin password /
WFA_MYSQL_PASS=password CONFIRM_WFA_MYSQL_PASS=confirm MySQL password
WFA_INSTALL_SITE=site WFA_INSTALL_ORGANIZATION=organization_name
WFA_HTTP_PORT=port WFA_HTTPS_PORT=port INSTALLDIR=install_directory
JDKINSTALLDIR=jdk_directory PerlDir=perl_directory
MySqlInstallDir=mysql_directory WFA_SERVICE_LOGON_USERNAME=wfa
service logon username WFA_SERVICE_LOGON_PASSWORD=wfa service logon
user password MYSQL_DATA_DIR= mysql data directory /qr /l*v
C:\install.log"
```

Example

```
WFA-x64-V5.1.0.0.1-B5355278.exe /s /v"WFA_ADMIN_USERNAME=admin
WFA_ADMIN_PASSWORD=Company*123 WFA_ADMIN_CONFIRM_PASSWORD=Company*123
```

```

WFA_MYSQL_PASS=MySQL*123 CONFIRM_WFA_MYSQL_PASS=MySQL*123
WFA_INSTALL_SITE=nb WFA_INSTALL_SITE=nb WFA_INSTALL_ORGANIZATION=netapp
WFA_HTTP_PORT=80 WFA_HTTPS_PORT=443 INSTALLDIR=|"C:\Program Files\NetApp\WFA\
JDKINSTALLDIR=|"C:\Program Files\NetApp\" PerlDir=|"C:\Perl64\" MySqlInstallDir=|"C:\Program
Files\MySQL\"WFA_SERVICE_LOGON_USERNAME=wfa
WFA_SERVICE_LOGON_PASSWORD=Wfa*1234\NetApp\WFA\" MYSQL_DATA_DIR=|"C:\Program
Files\NetApp\WFA\Database\" /qr /l*v C:\install.log"

```



The /qn option is not supported by WFA.

The command parameters are as follows:

Parameter	Description
WFA_ADMIN_USERNAME	Admin user name Optional parameter. If you do not specify a value, the value defaults to admin.
WFA_ADMIN_PASSWORD	Admin user password Mandatory parameter. The default admin password must satisfy the following criteria: <ul style="list-style-type: none"> • Minimum of eight characters • One uppercase character • One lowercase character • One numeral • One special character • The following characters are not allowed and cause password input to fail: `` ; < > , = & ^
WFA_ADMIN_CONFIRM_PASSWORD	Admin user password Mandatory parameter
WFA_MYSQL_PASS	MySQL user password Mandatory parameter
CONFIRM_WFA_MYSQL_PASS	MySQL user password Mandatory parameter
WFA_INSTALL_SITE	Organizational unit where WFA is being installed Mandatory parameter
WFA_INSTALL_ORGANIZATION	Organization or company name where WFA is being installed Mandatory parameter
WFA_HTTP_PORT	HTTP port Optional parameter. If you do not specify a value, the value defaults to 80.

Parameter	Description
WFA_HTTPS_PORT	HTTPS port Optional parameter. If you do not specify a value, the value defaults to 443.
INSTALLDIR	Installation directory path Optional parameter. If you do not specify a value, the path defaults to "C:\Program Files\NetApp\WFA\".
JDKINSTALLDIR	JDK installation directory path Optional parameter. If you do not specify a value, the path defaults to "C:\Program Files\NetApp\".
PerlDir	Perl installation directory path Optional parameter. If you do not specify a value, the path defaults to "C:\Perl64\".
MySQLInstallDir	MySQL installation directory path Optional parameter. If you do not specify a value, the path defaults to "C:\Program Files\MySQL\".
WFA_SERVICE_LOGON_USERNAME	<p>User name for WFA service logon Optional parameter. If you do not specify a value, the default user name is "wfa".</p> <p>For a domain user provide a user name in the format DOMAIN\USER. For a local system user the format is just a user name.</p> <p>WFA installer creates a local user if one does not exist. If a local user exists and the password entered is different than the existing password, WFA updates the password.</p> <p> Ensure the password conforms to the password policy configured for local users on the system. If the password does not conform to the password policy, the install fails.</p>
WFA_SERVICE_LOGON_PASSWORD	Password for WFA service logon Mandatory parameter
MYSQL_DATA_DIR	<p>Directory for MYSQL data Optional parameter. If you do not specify a value, the path defaults to "C:\ProgramData\MySQL\MySQLServerData"</p> <p>Available for fresh install only.</p>

Related information

Configure Workflow Automation with MSCS

After you install Workflow Automation (WFA) in Microsoft Cluster Server (MSCS), you must configure WFA for high availability in MSCS using configuration scripts.

You must have created a backup of WFA.



Before you start your configuration make sure that the WFA encryption key is set consistently on both MSCS cluster nodes. If it is not set on both nodes, then when failover occurs, the credentials cannot be decrypted on the second node causing workflow failures.

Steps

1. Log in to the first node of the MSCS cluster, and perform the following steps:

For...	Do this...
Windows 2012, Windows 2016, Windows 2019	<ol style="list-style-type: none"> From the Failover Cluster Manager, right-click Service Roles. Select Create Empty Service Role, and then rename the role to “WFA”. Add the IP address resource to the newly created “WFA” role: <ol style="list-style-type: none"> From the Failover Cluster Manager, right-click the newly created “WFA” role. Select Resource > More Resources > IP Address. Configure the cluster IP address.

2. Edit the `mscs_data_parameters.xml` file and set the relative path to the MySQL data directory:

```

<dir>
  <description>Data directory</description>
  <srcpath>...\\..\\..\\..\\..\\ProgramData\\MySQL\\MySQLServerData</srcpath>
  <destpath>wfa</destpath>
</dir>

```

3. Edit the `mscs_resource_properties.xml` file and make the following updates:

- Perform a find/replace for the `NA_WFA_DB` service name and update it to `MYSQL57`.
- Set the `vip_res <prettyname>` to the virtual IP address name:

```

<resource>
  <type>essential</type>
  <id>vip_res</id>
  <prettyname>WFA IP address</prettyname>
</resource>

```

c. Set the `data_res` `<prettyname>` to the disk name assigned to the share disk resource:

```

<resource>
  <type>essential</type>
  <id>datadisk_res</id>
  <prettyname>Cluster Disk 2</prettyname>
</resource>

```

d. Copy the XML files from the first node to the second node:

```
copy "\\\node1\\D$\\Program Files\\NetApp\\WFA\\bin\\ha\\*xml" "D:\\Program Files\\NetApp\\WFA\\bin\\ha"
```

e. Run the command to join the second node to the cluster:

```
D:\\Program Files\\NetApp\\WFA\\bin\\ha>perl ha_setup.pl --join -t mscs -f E:\\
```

4. At the command prompt, run the `ha_setup.pl` script to move the WFA data to the shared location and to configure WFA with MSCS for failover. The script is available at `WFA_install_location\\WFA\\bin\\ha`.

```
perl ha_setup.pl --first [-t type_of_cluster_vcs] [-g cluster_group_name] [-i IP_address_name] [-n cluster_name] [-k shared_disk_resource_name] [-f shared_drive_path]
```

The `ha_setup.pl` script expects an input using the IP Address resource for the MSCS cluster. When installing on MSCS 2016, the resource needs to be added by name, not IP address, WFA IP address. For example:

```
perl ha_setup.pl --first -t mscs -g WFA -i "WFA IP address" -n wfa_cluster -k "Cluster Disk 2" -f E:\\
```

5. Verify that the MSCS resources are created, by checking for the successfully configured message in the output.

Successfully configured MSCS cluster resources on this node

6. Stop the WFA services from the Failover Cluster Manager:

For...	Do this...
Windows 2012, Windows 2016, Windows 2019	<ol style="list-style-type: none"> Select Service Roles, and then select the newly created “WFA” role. In the Resource pane, right-click MYSQL57, and then select Take Offline. In the Resource pane, right-click NA_WFA_SRV, and then select Take Offline.

The WFA database service and the WFA server service must be taken offline. The WFA services must not be stopped from the Windows services.

1. Manually move the WFA resources to the secondary node.
2. Verify that the shared disk is accessible from the second node.
3. At the command prompt, run the ha_setup.pl script on the secondary node of the cluster to configure WFA for using the data from the shared location:

```
perl ha_setup.pl --join [-t type_of_cluster_msds] [-f shared_drive_path]
```

The ha_setup.pl script is available at WFA_install_location\WFA\bin\ha\.

```
perl ha_setup.pl --join -t msds -f E:\
```

4. From the Failover Cluster Manager, bring the WFA resources online:

For...	Do this...
Windows 2012, Windows 2016, Windows 2019	<ol style="list-style-type: none"> Right-click the newly created “WFA” role, and then select Start Role. The role must be in the Running status, and the individual resources must be in the Online state.

5. Manually switch to the second node of the MSCS cluster.
6. Verify that the WFA services start properly on the second node of the cluster.

Configure earlier versions of OnCommand Workflow Automation for high availability

You can configure OnCommand Workflow Automation (WFA) versions earlier than 3.1 for high availability.

Steps

1. Upgrade the existing version of WFA to the latest available version of WFA.

[Upgrade WFA](#)

This upgraded version of WFA is the primary node of the cluster.

2. Create a backup of the WFA database.

Backing up the WFA database

If any of the parameters were changed manually, you must create a backup of the WFA database, uninstall the existing WFA installation, install the latest available version of WFA, restore the backup, and then proceed with the Microsoft Cluster Service (MSCS) configuration.

3. Configure MSCS to install WFA on the primary node.

[Configure MSCS to install WFA](#)

4. Install the latest available version of WFA on the secondary node.

[Install WFA](#)

5. Configure WFA in MSCS.

[Configure WFA in MSCS](#)

The WFA server is configured for high availability.

Uninstall Workflow Automation in an MSCS environment

You can uninstall Workflow Automation (WFA) from a cluster by deleting all the WFA services from the cluster nodes.

This task applies to Windows Server 2012.

Steps

1. Take the services offline by using Failover Cluster Manager:
 - a. Right-click the role.
 - b. Select **Stop Role**.
2. Uninstall WFA on the first node, and then uninstall WFA on the second node.

[Uninstall OnCommand Workflow Automation](#)

3. Delete the cluster resources from Failover Cluster Manager:
 - a. Right-click the role.
 - b. Select **Remove**.
4. Manually delete the data in the shared location.

Setting up OnCommand Workflow Automation

After you complete installing OnCommand Workflow Automation (WFA), you must complete several configuration settings. You have to access WFA, configure users, set up data sources, configure credentials, and configure WFA.

Access OnCommand Workflow Automation

You can access OnCommand Workflow Automation (WFA) through a web browser from

any system that has access to the WFA server.

You must have installed Adobe Flash Player for your web browser.

Steps

1. Open a web browser and enter one of the following in the address bar:

- `https://wfa_server_ip`

`wfa_server_ip` is the IP address (IPv4 or IPv6 address) or the fully qualified domain name (FQDN) of the WFA server.

- If you are accessing WFA on the WFA server: `https://localhost/wfa`

If you have specified a non-default port for WFA, you must include the port number as follows:

- `https://wfa_server_ip:port`

- `https://localhost:port`

`port` is the TCP port number you have used for the WFA server during installation.

2. In the Sign in section, enter the credentials of the admin user that you have entered during installation.

3. In the **Settings > Setup** menu, set up the credentials and a data source.

4. Bookmark the WFA web GUI for ease of access.

OnCommand Workflow Automation data sources

OnCommand Workflow Automation (WFA) operates on data that is acquired from data sources. Various versions of Active IQ Unified Manager and VMware vCenter Server are provided as predefined WFA data source types. You must be aware of the predefined data source types before you set up the data sources for data acquisition.

A data source is a read-only data structure that serves as a connection to the data source object of a specific data source type. For example, a data source can be a connection to an Active IQ Unified Manager database of an Active IQ Unified Manager 6.3 data source type. You can add a custom data source to WFA after defining the required data source type.

For more information about the predefined data source types, see the Interoperability Matrix.

Related information

[NetApp Interoperability Matrix Tool](#)

Configuring a database user on DataFabric Manager

You must create a database user on DataFabric Manager 5.x to configure read-only access of the DataFabric Manager 5.x database to OnCommand Workflow Automation.

Configure a database user by running `ocsetup` on Windows

You can run the `ocsetup` file on the DataFabric Manager 5.x server to configure read-only access of the DataFabric Manager 5.x database to OnCommand Workflow Automation.

Steps

1. Download the `wfa_ocsetup.exe` file to a directory in the DataFabric Manager 5.x server from the following location: https://WFA_Server_IP/download/wfa_ocsetup.exe.

`WFA_Server_IP` is the IP address (IPv4 or IPv6 address) of your WFA server.

If you have specified a non-default port for WFA, you must include the port number as follows: https://wfa_server_ip:port/download/wfa_ocsetup.exe.

`port` is the TCP port number that you have used for the WFA server during installation.

If you are specifying an IPv6 address, you must enclose it with square brackets.

2. Double-click the `wfa_ocsetup.exe` file.
3. Read the information in the setup wizard and click **Next**.
4. Browse or type the OpenJDK location and click **Next**.
5. Enter a user name and password to override the default credentials.

A new database user account is created with access to the DataFabric Manager 5.x database.



If you do not create a user account, the default credentials are used. You must create a user account for security purposes.

6. Click **Next** and review the results.
7. Click **Next**, and then click **Finish** to complete the wizard.

Configure a database user by running `ocsetup` on Linux

You can run the `ocsetup` file on the DataFabric Manager 5.x server to configure read-only access of the DataFabric Manager 5.x database to OnCommand Workflow Automation.

Steps

1. Download the `wfa_ocsetup.sh` file to your home directory on the DataFabric Manager 5.x server using the following command in the terminal:

```
wget https://WFA_Server_IP/download/wfa_ocsetup.sh
```

`WFA_Server_IP` is the IP address (IPv4 or IPv6 address) of your WFA server.

If you have specified a non-default port for WFA, you must include the port number as follows:

```
wget https://wfa_server_ip:port/download/wfa_ocsetup.sh
```

`port` is the TCP port number that you have used for the WFA server during installation.

If you are specifying an IPv6 address, you must enclose it with square brackets.

2. Use the following command in the terminal to change the `wfa_ocsetup.sh` file to an executable: `chmod +x wfa_ocsetup.sh`
3. Run the script by entering the following in the terminal:

```
./wfa_ocsetup.sh OpenJDK_path
```

OpenJDK_path is the path to OpenJDK.

/opt/NTAPdfm/java

The following output is displayed in the terminal, indicating a successful setup:

```
Verifying archive integrity... All good.
Uncompressing WFA OnCommand Setup.....  
*** Welcome to OnCommand Setup Utility for Linux ***  
<Help information>  
*** Please override the default credentials below ***  
Override DB Username [wfa] :
```

4. Enter a user name and password to override the default credentials.

A new database user account is created with access to the DataFabric Manager 5.x database.



If you do not create a user account, the default credentials are used. You must create a user account for security purposes.

The following output is displayed in the terminal, indicating a successful setup:

```
***** Start of response from the database *****  
>>> Connecting to database  
<<< Connected  
*** Dropped existing 'wfa' user  
==> Created user 'username'  
>>> Granting access  
<<< Granted access  
***** End of response from the database *****  
***** End of Setup *****
```

Configure a database user on Active IQ Unified Manager

You must create a database user on Active IQ Unified Manager to configure read-only access of the Active IQ Unified Manager database to OnCommand Workflow Automation.

Steps

1. Log in to Active IQ Unified Manager with administrator credentials.
2. Click **Settings > Users**.
3. Click **Add a New User**.
4. Select **Database User** as the type of user.

The same user should be used in OnCommand Workflow Automation while adding Active IQ Unified Manager as a data source in OnCommand Workflow Automation.

Set up a data source

You must set up a connection with a data source in OnCommand Workflow Automation (WFA) to acquire data from the data source.

- For Active IQ Unified Manager 6.0 and later, you must have created a database user account on the Unified Manager server.

See the *OnCommand Unified Manager Online Help* for details.

- The TCP port for incoming connections on the Unified Manager server must be open.

See the documentation on your firewall for details.

The following are the default TCP port numbers:

TCP port number	Unified Manager server version	Description
3306	6.x	MySQL database server

- For Performance Advisor, you must have created an Active IQ Unified Manager user account with a minimum role of GlobalRead.

See the *OnCommand Unified Manager Online Help* for details.

- For VMware vCenter Server, you must have created a user account on the vCenter Server.

See the VMware vCenter Server documentation for details.



You must have installed VMware PowerCLI. If you want to execute workflows only on vCenter Server data sources, setting up Unified Manager server as a data source is not required.

- The TCP port for incoming connections on the VMware vCenter Server must be open.

The default TCP port number is 443. See the documentation on your firewall for details.

You can add multiple Unified Manager server data sources to WFA using this procedure. However, you must not use this procedure if you want to pair Unified Manager server 6.3 and later with WFA and use the protection functionality in Unified Manager server.

For more information about pairing WFA with Unified Manager server 6.x, see the *OnCommand Unified Manager Online Help*.



While setting up a data source with WFA, you must be aware that Active IQ Unified Manager 6.0, 6.1, and 6.2 data source types are deprecated in the WFA 4.0 release, and these data source types will not be supported in future releases.

Steps

1. Access WFA using a web browser.
2. Click **Settings**, and under **Setup** click **Data Sources**.

3. Choose the appropriate action:

To...	Do this...
Create a new data source	Click  on the toolbar.
Edit a restored data source if you have upgraded WFA	Select the existing data source entry, and click  on the toolbar.

If you have added a Unified Manager server data source to WFA and then upgraded the version of the Unified Manager server, WFA will not recognize the upgraded version of the Unified Manager server. You must delete the previous version of the Unified Manager server and then add the upgraded version of the Unified Manager server to WFA.

4. In the New Data Source dialog box, select the required data source type, and enter a name for the data source and the host name.

Based on the selected data source type, the port, user name, password, and timeout fields might be automatically populated with the default data, if available. You can edit these entries as required.

5. Choose an appropriate action:

For...	Do this...
Active IQ Unified Manager 6.3 and later	<p>Enter the credentials of the Database User account that you created on the Unified Manager server. See <i>OnCommand Unified Manager Online Help</i> for details on creating a database user account.</p> <p> You must not provide the credentials of an Active IQ Unified Manager Database User account that was created using the command-line interface or the ocsetup tool.</p>
VMware vCenter Server (only for windows)	(only for windows) Enter the user name and password of the user that you created on the VMware vCenter server.

6. Click **Save**.

7. In the Data Sources table, select the data source, and click  on the toolbar.

8. Verify the status of the data acquisition process.

Add an upgraded Unified Manager server as a data source

If Unified Manager server (5.x or 6.x) is added as a data source to WFA and then the Unified Manager server is upgraded, you must add the upgraded Unified Manager server as a data source because the data that is associated with the upgraded version is not populated in WFA unless it is manually added as a data source.

Steps

1. Log into the WFA web GUI as an admin.
2. Click **Settings** and under **Setup**, click **Data Sources**.
3. Click  on the toolbar.
4. In the New Data Source dialog box, select the required data source type, and then enter a name for the data source and the host name.

Based on the selected data source type, the port, user name, password, and timeout fields might be automatically populated with the default data, if available. You can edit these entries as required.

5. Click **Save**.
6. Select the previous version of the Unified Manager server, and click  on the toolbar.
7. In the Delete Data Source Type confirmation dialog box, click **Yes**.
8. In the Data Sources table, select the data source, and then click  on the toolbar.
9. Verify the data acquisition status in the History table.

Create local users

OnCommand Workflow Automation (WFA) enables you to create and manage local WFA users with specific permissions for various roles, such as guest, operator, approver, architect, admin, and backup.

You must have installed WFA and logged in as an admin.

WFA enables you to create users for the following roles:

- **Guest**

This user can view the portal and the status of a workflow execution, and can be notified of a change in the status of a workflow execution.

- **Operator**

This user is allowed to preview and execute workflows for which the user is given access.

- **Approver**

This user is allowed to preview, execute, approve, and reject workflows for which the user is given access.



It is recommended to provide the email ID of the approver. If there are multiple approvers, you can provide a group email ID in the **E-mail** field.

- **Architect**

This user has full access to create workflows, but is restricted from modifying global WFA server settings.

- **Admin**

This user has complete access to the WFA server.

- **Backup**

This is the only user who can remotely generate backups of the WFA server. However, the user is restricted from all other access.

Steps

1. Click **Settings**, and under **Management** click **Users**.
2. Create a new user by clicking  on the toolbar.
3. Enter the required information in the New User dialog box.
4. Click **Save**.

Configure the credentials of a target system

You can configure the credentials of a target system in OnCommand Workflow Automation (WFA) and use the credentials to connect to that specific system and execute commands.

After initial data acquisition, you must configure the credentials for the arrays on which the commands are run. PowerShell WFA controller connection works in two modes:

- With credentials

WFA tries to establish a connection using HTTPS first, and then tries using HTTP. You can also use Microsoft Active Directory LDAP authentication to connect to arrays without defining credentials in WFA. To use Active Directory LDAP, you must configure the array to perform authentication with the same Active Directory LDAP server.

- Without credentials (for storage systems operating in 7-Mode)

WFA tries to establish a connection using domain authentication. This mode uses the remote procedure call protocol, which is secured using the NTLM protocol.

- WFA checks the Secure Sockets Layer (SSL) certificate for ONTAP systems. Users might be prompted to review and accept/deny the connection to ONTAP systems if the SSL certificate is not trusted.
- You must reenter the credentials for ONTAP, NetApp Active IQ and Lightweight Directory Access Protocol (LDAP) after you restore a backup or complete an in-place upgrade.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Setup** click **Credentials**.
3. Click  on the toolbar.
4. In the New Credentials dialog box, select one of the following options from the **Match** list:

- **Exact**

Credentials for a specific IP address or host name

- **Pattern**

Credentials for the entire subnet or IP range



The use of regular expression syntax is not supported for this option.

5. Select the remote system type from the **Type** list.
6. Enter either the host name or the IPv4 or IPv6 address of the resource, the user name, and the password.



WFA 5.1 verifies the SSL certificates of all resources added to WFA. As certificate verification might prompt you to accept the certificates, using wildcards in credentials is not supported. If you have multiple clusters using the same credentials, you cannot add them all at once.

7. Test the connectivity by performing the following action:

If you selected the following match type...	Then...
Exact	Click Test .
Pattern	Save the credentials and choose one of the following: <ul style="list-style-type: none">• Select the credential and click  on the toolbar.• Right-click and select Test Connectivity.

8. Click **Save**.

Configuring OnCommand Workflow Automation

OnCommand Workflow Automation (WFA) enables you to configure various settings—for example, AutoSupport and notifications.

When configuring WFA, you can set up one or more of the following, as required:

- AutoSupport for sending AutoSupport messages to technical support
- Microsoft Active Directory Lightweight Directory Access Protocol (LDAP) server for LDAP authentication and authorization for WFA users
- Mail for email notifications about workflow operations and sending AutoSupport messages
- Simple Network Management Protocol (SNMP) for notifications about workflow operations
- Syslog for remote data logging

Configure AutoSupport

You can configure several AutoSupport settings such as the schedule, content of the AutoSupport messages, and the proxy server. AutoSupport sends weekly logs of the content that you selected to technical support for archiving and issue analysis.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Setup** click **AutoSupport**.

3. Ensure that the **Enable AutoSupport** box is selected.
4. Enter the required information.
5. Select one of the following from the **Content** list:

If you want to include...	Then choose this option...
Only configuration details such as users, workflows, and commands of your WFA installation	send only configuration data
WFA configuration details and data in WFA cache tables such as the scheme	send configuration and cache data (default)
WFA configuration details, data in WFA cache tables, and data in the installation directory	send configuration and cache extended data



The password of any WFA user is *not* included in the AutoSupport data.

6. Test that you can download an AutoSupport message:
 - a. Click **Download**.
 - b. In the dialog box that opens, select the location to save the .7z file.
7. Test the sending of an AutoSupport message to the specified destination by clicking **Send Now**.
8. Click **Save**.

Configure authentication settings

You can configure OnCommand Workflow Automation (WFA) to use a Microsoft Active Directory (AD) Lightweight Directory Access Protocol (LDAP) server for authentication and authorization.

You must have configured a Microsoft AD LDAP server in your environment.

Only Microsoft AD LDAP authentication is supported for WFA. You cannot use any other LDAP authentication methods, including Microsoft AD Lightweight Directory Services (AD LDS) or Microsoft Global Catalog.



During communication, LDAP sends the user name and password in plain text. However, LDAPS (LDAP secure) communication is encrypted and secure.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Setup** click **Authentication**.
3. Select the **Enable Active Directory** check box.
4. Enter the required information in the fields:
 - a. If you want to use the user@domain format for domain users, replace sAMAccountName with userPrincipalName in the **User name attribute** field.
 - b. If unique values are required for your environment, edit the required fields.

c. Enter the AD server URI as follows: `ldap://active_directory_server_address[:port\]`

`ldap://NB-T01.example.com[:389]`

If you have enabled LDAP over SSL, you can use the following URI format:

`ldaps://active_directory_server_address[:port\]`

d. Add a list of AD group names the required roles.



You can add a list of AD group names to the required roles in the Active Directory Groups Window.

5. Click **Save**.

6. If LDAP connectivity to an array is required, configure the WFA service to log on as the required domain user:

a. Open the Windows services console by using `services.msc`.

b. Double-click the **NetApp WFA Server** service.

c. In the NetApp WFA Server Properties dialog box, click the **Log On** tab, and then select **This account**.

d. Enter the domain user name and password, and then click **OK**.

Add Active Directory groups

You can add Active Directory groups in OnCommand Workflow Automation (WFA).

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings** and under **Management**, click **Active Directory Groups**.
3. In the Active Directory Groups window, click the **New** icon.
4. In the New Active Directory Group dialog box, enter the required information.

If you select **Approver** from the **Role** drop down list, it is recommended provide the email ID of the approver. If there are multiple approvers, you can provide a group email ID in the **E-mail** field. Select the different events of the workflow for which the notification is to be sent to the particular Active Directory group.

5. Click **Save**.

Configure email notifications

You can configure OnCommand Workflow Automation (WFA) to send you email notifications about workflow operations—for example, workflow started or workflow failed.

You must have configured a mail host in your environment.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Setup** click **Mail**.
3. Enter the required information in the fields.

4. Test the mail settings by performing the following steps:
 - a. Click **Send test mail**.
 - b. In the Test Connection dialog box, enter the email address to which you want to send the email.
 - c. Click **Test**.
5. Click **Save**.

Configure SNMP

You can configure OnCommand Workflow Automation (WFA) to send Simple Network Management Protocol (SNMP) traps about the status of workflow operations.

WFA now supports SNMP v1 and SNMP v3 protocols. SNMP v3 provides additional security features.

The WFA .mib file provides information about the traps that are sent by the WFA server. The .mib file is located in the <WFA_install_location>\wfa\bin\wfa.mib directory on the WFA server.



The WFA server sends all the trap notifications with a generic object identifier (1.3.6.1.4.1.789.1.1.12.0).

You cannot use SNMP community strings such as community_string@SNMP_host for SNMP configuration.

Configure SNMP Version 1

Steps

1. Log in to WFA through a web browser as an admin user, and then access the WFA server.
2. Click **Settings**, and under **Setup** click **SNMP**.
3. Select the **Enable SNMP** check box.
4. In the **Version** drop-down list, select **Version 1**.
5. Enter an IPv4 or IPv6 address or the host name, and the port number of the management host.

WFA sends SNMP traps to the specified port number. The default port number is 162.

6. In the **Notify On** section, select one or more of the following check boxes:
 - Workflow execution started
 - Workflow execution completed successfully
 - Workflow execution failed/partially successful
 - Workflow execution waiting for approval
 - Acquisition failure
7. Click **Send Test Notification** to verify the settings.
8. Click **Save**.

Configure SNMP Version 3

You can also configure OnCommand Workflow Automation (WFA) to send Simple Network Management Protocol (SNMP) Version 3 traps about the status of workflow operations.

Version 3 offers two additional security options:

- Version 3 with Authentication

Traps are sent unencrypted over the network. SNMP management applications, which are configured by the same authentication parameters as SNMP trap messages, can receive traps.

- Version 3 with Authentication and Encryption

Traps are sent encrypted over the network. To receive and decrypt these traps, you must configure SNMP management applications with the same authentication parameters and encryption key as the SNMP traps.

Steps

1. Log in to WFA through a web browser as an admin user, and then access the WFA server.
2. Click **Settings**, and under **Setup** click **SNMP**.
3. Select the **Enable SNMP** check box.
4. In the **Version** drop-down list, select one of the following options:
 - Version 3
 - Version 3 with Authentication
 - Version 3 with Authentication and Encryption
5. Select the SNMP configuration options that correspond to the specific SNMP Version 3 option you chose in Step 4.
6. Enter an IPv4 or IPv6 address or the host name, and the port number of the management host. WFA sends SNMP traps to the specified port number. The default port number is 162.
7. In the Notify On section, select one or more of the following check boxes:
 - Workflow planning started/failed/completed
 - Workflow execution started
 - Workflow execution completed successfully
 - Workflow execution failed/ partially successful
 - Workflow execution waiting for approval
 - Acquisition failure
8. Click **Send Test Notification** to verify the settings.
9. Click **Save**.

Configure Syslog

You can configure OnCommand Workflow Automation (WFA) to send log data to a specific Syslog server for purposes such as event logging and log information analysis.

You must have configured the Syslog server to accept data from the WFA server.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Maintenance** click **Syslog**.

3. Select the **Enable Syslog** check box.
4. Enter the Syslog host name and select the Syslog log level.
5. Click **Save**.

Configure protocols for connecting to remote systems

You can configure the protocol used by OnCommand Workflow Automation (WFA) to connect to remote systems. You can configure the protocol based on your organization's security requirements and the protocol supported by the remote system.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Data Source Design > Remote System Types**.
3. Perform one of the following actions:

If you want to...	Do this...
Configure a protocol for a new remote system	<ol style="list-style-type: none"> a. Click . b. In the New Remote System Type dialog box, specify the details such as name, description, and version.
Modify the protocol configuration of an existing remote system	<ol style="list-style-type: none"> a. Select and double-click the remote system that you want to modify. b. Click .

4. From the Connection Protocol list, select one of the following:
 - HTTPS with fallback to HTTP (default)
 - HTTPS only
 - HTTP only
 - Custom
5. Specify the details for the protocol, default port, and default timeout.
6. Click **Save**.

Disable the default password policy

OnCommand Workflow Automation (WFA) is configured to enforce a password policy for local users. If you do not want to use the password policy, you can disable it.

You must have logged in to the WFA host system as an admin.

The default WFA installation path is used in this procedure. If you changed the default location during installation, you must use the changed WFA installation path.

Steps

1. Open Windows Explorer and navigate to the following directory: `WFA_install_location\WFA\bin\`.

2. Double-click the ps.cmd file.

A PowerShell command-line interface (CLI) prompt opens with ONTAP and WFA modules loaded in it.

3. At the prompt, enter the following:

```
Set-WfaConfig -Name PasswordPolicy -Enable $false
```

4. When prompted, restart the WFA services.

Modify the default password policy for Windows

OnCommand Workflow Automation (WFA) enforces a password policy for local users. You can modify the default password policy to set a password as per your requirement.

You must be logged in to the WFA host system as a root user.

- The default WFA installation path is used in this procedure.

If you changed the default location during installation, you must use the custom WFA installation path.

- The command for modifying the default password policy is .\wfa --password-policy=default.

The default setting is

“minLength=true,8;specialChar=true,1;digitalChar=true,1;lowercaseChar=true,1;uppercaseChar=true,1;whitespaceChar=false”. Per this setting for the default password policy, the password must have a minimum length of eight characters, must contain at least one special character, one digit, one lowercase character, and one uppercase character, and must not contain spaces.

Steps

1. At the command prompt, navigate to the following directory on the WFA server:

```
WFA_install_location/wfa/bin/
```

2. Modify the default password policy:

```
.\wfa --password-policy=PasswordPolicyString --restart=WFA
```

Enable remote access to the OnCommand Workflow Automation database on Windows

By default, the OnCommand Workflow Automation (WFA) database can be accessed only by clients that are running on the WFA host system. You can change the default settings if you want to access the WFA database from a remote system.

- You must have logged in to the WFA host system as an admin user.
- If a firewall is installed on the WFA host system, you must have configured your firewall settings to allow access from the remote system.

The default WFA installation path is used in this procedure. If you changed the default location during installation, you must use the custom WFA installation path.

Steps

1. Open Windows Explorer, and navigate to the following directory: WFA_install_location\WFA\bin
2. Perform one of the following actions:

To...	Enter the following command...
Enable remote access	.\wfa --db-access=public --restart
Disable remote access	.\wfa --db-access=default --restart

Restrict access rights of OnCommand Workflow Automation on the host

By default, OnCommand Workflow Automation (WFA) executes the workflows as the admin of the host system. You can restrict WFA rights on the host system by changing the default settings.

You must have logged in to the WFA host system as an admin.

Steps

1. Create a new Windows user account with permissions to open sockets and to write to the WFA home directory.
2. Open the Windows services console by using services.msc and double-click **NetApp WFA Database**.
3. Click the **Log On** tab.
4. Select **This account** and enter the credentials of the new user you have created, and then click **OK**.
5. Double-click **NetApp WFA Server**.
6. Click the **Log On** tab.
7. Select **This account** and enter the credentials of the new user you have created, and then click **OK**.
8. Restart the **NetApp WFA Database** and the **NetApp WFA Server** services.

Modify the transaction timeout setting of OnCommand Workflow Automation

The OnCommand Workflow Automation (WFA) database transaction times out in 300 seconds by default. You can increase the default timeout duration when restoring a large-sized WFA database from a backup to avoid potential failure of the database restoration.

You must have logged in to the WFA host system as an admin.

The default WFA installation path is used in this procedure. If you changed the default location during installation, you must use the changed WFA installation path.

Steps

1. Open Windows Explorer and navigate to the following directory:
WFA_install_location\WFA\bin
2. Double-click the ps.cmd file.

A PowerShell command-line interface (CLI) prompt opens with ONTAP and WFA modules loaded in it.

3. At the prompt, enter the following:

```
Set-WfaConfig -Name TransactionTimeOut -Seconds NumericValue
```

```
Set-WfaConfig -Name TransactionTimeOut -Seconds 1000
```

4. When prompted, restart the WFA services.

Configure the timeout value for Workflow Automation

You can configure the timeout value for the Workflow Automation (WFA) web GUI, instead of using the default timeout value.

The default timeout value for WFA web GUI is 180 minutes. You can configure the timeout value to meet your requirements through CLI. You cannot set the timeout value from the WFA web GUI.

 The timeout value that you set is an absolute timeout rather than a timeout related to inactivity. For example, if you set this value to 30 minutes, then you are logged out after 30 minutes, even if you are active at the end of this time.

Steps

1. Log in as the administrator on the WFA host machine.
2. Set the timeout value:

```
installdir bin/wfa -S=timeout value in minutes
```

Enabling ciphers and adding new ciphers

OnCommand Workflow Automation 5.1 supports a number of ciphers out of the box. Also, you can add additional ciphers as required.

The following ciphers can be enabled out of the box:

```
enabled-cipher-suites=
"TLS_DHE_DSS_WITH_AES_128_GCM_SHA256, TLS_DHE_DSS_WITH_AES_256_GCM_SHA384, TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256,
TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384, TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384,
TLS_RSA_WITH_AES_128_GCM_SHA256, TLS_RSA_WITH_AES_256_GCM_SHA384"
```

Additional ciphers can be added to this configuration in the `standalone-full.xml` file. This file is located at: `<installdir>/jboss/standalone/configuration/standalone-full.xml`.

The file can be modified to support additional ciphers as follows:

```
<https-listener name="https" socket-binding="https" max-post-size="1073741824" security-realm="SSLRealm"  
enabled-cipher-suites="**< --- add additional ciphers here ---\>**"  
enabled-protocols="TLSv1.1,TLSv1.2"/>
```

Upgrade OnCommand Workflow Automation

If you have installed a previous version of OnCommand Workflow Automation (WFA), you can upgrade to the latest version of WFA to use the new features and enhancements.

- You can upgrade to WFA 5.1 from either WFA 5.0 or 4.2 only.

If you are currently running WFA 4.1 or earlier versions of WFA, then you must first upgrade to either WFA 5.0 or 4.2 and then upgrade to WFA 5.1.

- You can restore a backup taken on either WFA 5.0 or 4.2 on WFA 5.1. A WFA database backup can only be restored to a system that is running the same version or a later version of WFA.

For example, if you created a backup on a system that is running WFA 4.2, the backup can be restored only to systems that are running WFA 4.2 or later.

- You cannot install MySQL on your own when upgrading from versions earlier than WFA 4.2.

However, you can install MySQL on your own:

- During a fresh installation of WFA 4.2 and later or
- When you are upgrading from WFA 4.2 to later versions of WFA.
 1. Back up the WFA database by using one of the following options:

- WFA web portal
- PowerShell script

If you want to revert to the same version of WFA, you can use the backup that you created to restore your WFA database.

1. Uninstall the existing version of WFA.
2. Install the latest version of WFA.
3. Restore the WFA database.

You can review the restored content for completeness in functionality—for example, you can check the functioning of your custom workflows.

Upgrade from OnCommand Workflow Automation 3.1 or later versions

You can perform an in-place upgrade from OnCommand Workflow Automation (WFA) 3.1 or later versions to the latest available versions of WFA to use the new features and enhancements.

You must have downloaded the .exe binary file from the NetApp Support Site to the WFA host machine.

The WFA 5.1 cluster connection needs to accept the SSL certificate. When updating from an earlier version of WFA to WFA 5.1, you need to certify the cluster connection. Save the cluster connection details for cluster certification after the in-place upgrade.

You cannot install MySQL on your own when upgrading from earlier versions of WFA. However, you can install MySQL on your own:

- During a fresh installation of WFA 4.2 and later or
- When you are upgrading from WFA 4.2 to later versions of WFA.

Step

1. Upgrade from WFA 3.1 or later versions by choosing one of the following methods:

- Interactive installation
 - i. Navigate to the .exe binary file in the WFA host machine, and then run the file.
 - ii. Follow the on-screen instructions in the wizard to complete the upgrade.
- Silent installation

Enter the following at the command prompt:

```
WFA-version_number-build_number.exe /s /v"WFA_ADMIN_USERNAME_UP=wfa username  
WFA_ADMIN_PASSWORD_UP=password /qr /l*v C:\upgrade.log"
```

Example:

```
WFA-x64-v4.2.0.0.0-B2973881.exe /s /v"WFA_ADMIN_USERNAME_UP=admin  
WFA_ADMIN_PASSWORD_UP=password*123 /qr /l*v C:\upgrade.log"
```



The /qn option is not supported by WFA.

If you want to perform a silent upgrade, then you must include values for all of the command parameters:

- WFA_ADMIN_USERNAME_UP is the user name of a WFA user who has permission to create a WFA database backup.
- WFA_ADMIN_PASSWORD_UP is the password of the user.

Pack identification during upgrade

During the upgrade process, OnCommand Workflow Automation (WFA) identifies and classifies the entities into a pack. If you had deleted any entity of a pack before the upgrade, the pack will not be identified during the upgrade.

During the upgrade process, WFA compares the packs in the database with the list of packs that were released in the Storage Automation Store to identify the packs that were installed before the upgrade. Pack identification thus classifies existing packs in the database.

WFA performs the following processes to identify and classify packs:

- Maintains a list of packs released in the Storage Automation Store to compare and identify the packs that were installed before the upgrade.
- Classifies the entities in a pack as part of the Storage Automation Store synchronization, if Storage Automation Store is enabled.
- Classifies the entities into packs using the updated list.

Pack identification is applicable only to NetApp-certified packs that were downloaded from the Storage Automation Store.

If a pack is not identified during upgrade, you can re-import the pack to get it identified in WFA. The wfa.log files provide details about the entities that were not identified as a pack during the upgrade.

Upgrading third-party products

You can upgrade third-party products on OnCommand Workflow Automation (WFA) such as OpenJDK, MySQL, and ActiveState Perl in Windows. Third-party products like Open JDK, MYSQL, and so on, report security vulnerabilities. Starting from this release of WFA, you can now upgrade third-party products on your own.

Upgrade OpenJDK

Oracle JRE is no longer supported in OnCommand Workflow Automation. In this release, OpenJDK replaces Oracle JRE for Windows. You can upgrade new versions of OpenJDK for OnCommand Workflow Automation (WFA) on the Windows server. You can upgrade to a newer version of OpenJDK to get fixes for security vulnerabilities on the Windows server.

You must have Windows admin privileges on the WFA server.

You can update OpenJDK releases within release families. For example, you can upgrade from OpenJDK 11.0.1 to OpenJDK 11.0.2, but you cannot update directly from OpenJDK 11 to OpenJDK 12.

Steps

1. Log in as an admin user on the WFA host machine.
2. Download the latest version of OpenJDK 11 (64-bit) to the target system.
3. Use the Windows Services console to stop the WFA server and WFA Database services.
4. Extract the downloaded version of OpenJDK 11 to the folder where you have installed open JDK.
5. Use the Windows Services console to start the WFA services.

Upgrade MySQL

You can upgrade new versions of MySQL for OnCommand Workflow Automation (WFA) on the Windows server. You can upgrade to a newer version of MySQL to get fixes for security vulnerabilities on the Windows server.

You must have Windows admin privileges and the password for MySQL root user on the WFA server.



Before reinstalling WFA 4.2 or later, you must have deleted the MySQL data directory if you have uninstalled MySQL.

You should be aware of the following limitations:

- You can upgrade within any version of MySQL 5.7.

For example, you can upgrade from MySQL 5.7.1 to MySQL 5.7.2.

- You cannot upgrade from MySQL 5.7 to MySQL 5.8

Steps

1. Log in as the admin user on the WFA host machine.
2. Download the appropriate version of MySQL to the target system.
3. Use the Windows Services console to stop the following WFA services:
 - NetApp WFA Database or MYSQL57
 - NetApp WFA Server
4. Click on the MYSQL msi package to invoke the upgrade of MySQL.
5. Follow the instructions on the screen to complete MySQL installation.
6. Start the WFA services by using the Windows Services console.

Upgrade ActiveState Perl

OnCommand Workflow Automation (WFA) works with the Enterprise edition of ActiveState Perl on Windows. You can upgrade to a newer version of ActiveState Perl to get fixes for security vulnerabilities on the Windows server.

You must have Windows admin privileges on the WFA server. ActiveState Perl does not support “in-place” upgrades.

WFA 5.1 uses the Enterprise edition of ActiveState Perl.

You can upgrade from ActiveState Perl 5.26.3 to later builds. You cannot upgrade to a major release of ActiveState Perl.

Steps

1. Log in as the admin user on the WFA host machine.
2. Download the latest version of 64-bit ActiveState Enterprise Edition 5.26.3 to the target system.
3. Use the Windows Services console to stop the following WFA services:
 - WFA Database or MYSQL57
 - WFA Server
4. Uninstall the current version of ActiveState Perl on the target system from the control panel.
5. Perform a backup of the C:\Perl64\site\lib folder.
6. Install the new ActiveState Enterprise Edition on the target machine.
7. Restore the \site\lib folder of ActiveState Enterprise Edition whose backup you created in step 5.

8. Restart the WFA services by using the Windows Services console.

Backing up the OnCommand Workflow Automation database

A backup of the OnCommand Workflow Automation (WFA) database includes the system configuration settings and cache information, including the playground database. You can use the backup for restoration purposes on the same system or on a different system.

An automatic backup of the database is created daily at 2 a.m. and is saved as a .zip file in the following location: wfa_install_location/WFA-Backups.

WFA saves up to five backups in the WFA-Backups directory, and replaces the oldest backup with the latest backup. The WFA-Backups directory is not deleted when you uninstall WFA. You can use the automatically created backup for restoration if you did not create a backup of the WFA database while uninstalling WFA.

You can also manually back up the WFA database when you have to save specific changes for restoration; for example, if you want to back up the changes that you have made before the automatic backup occurs.

- You can restore a WFA database backup only to a system that is running the same version or a later version of WFA.



For example, if you created a backup on a system that is running WFA 4.2, the backup can be restored only to systems that are running WFA 4.2 or later.

- You cannot use the web UI to back up the WFA database during disaster recovery in a high-availability setup.

Backup and restoration of user credentials

The backup of the WFA database includes the WFA user credentials.



The WFA database is also included in the AutoSupport data; however, the password of any WFA user is not included in the AutoSupport data.

When a WFA database is restored from a backup, the following items are preserved:

- The admin user credentials that were created during the current WFA installation.
- If a user with admin privileges other than the default admin user restores the database, the credentials of both the admin users.
- All other user credentials of the current WFA installation are replaced with the user credentials from the backup.

Back up the WFA database from the web portal

You can back up the OnCommand Workflow Automation (WFA) database from the web portal and use the backup file for data recovery purposes. You cannot perform a full backup from the web portal.

You must have admin or architect credentials to perform this task.

A WFA user with backup role cannot log in to the web portal to perform a backup. The WFA users with backup role can only perform remote or scripted backups.

Steps

1. Log in to the WFA web GUI as an admin.
2. Click **Settings** and under **Maintenance**, click **Backup & Restore**.
3. Click **Backup**.
4. In the dialog box that opens, select a location, and then save the file.

Back up the WFA database using the PowerShell script

If you want to back up the OnCommand Workflow Automation (WFA) database frequently, you can use the PowerShell script that is provided with the WFA installation package.

You must have admin user credentials, architect credentials, or backup user credentials.

For more information, see the REST documentation.

Steps

1. Open Windows PowerShell as an admin user, and then back up the WFA database:

```
<WFA_install_location\WFA\bin\Backup.ps1> -User user_name -Password password  
-Path backup_file_path
```

- WFA_install_location is the WFA installation directory.
- user_name is the user name of the admin user, architect, or backup user.
- password is the password of the admin user, architect, or backup user.
- backup_file_path is the complete directory path for the backup file.



The backup file is a zip file with the name in the following format: wfa_backup_servername_.zip

- wfa_backup_ is a fixed portion of the file name, which is the name of the backup server.
- servername is extracted from the environment of the Windows server.
- _.zip is a fixed portion of the file name.

```
C:\Program Files\NetApp\WFA\bin\Backup.ps1 -User backup -Password  
MyPassword123 -Path C:\WFA_backups\backup_10_08_12
```

After the backup is complete, the following output is displayed:

```
C:\WFA_backups\backup_1008_12\wfa_backup_myserver.zip  
. Verify that the backup file was created at the specified location.
```

Backing up the WFA database using the CLI

If you want to back up the OnCommand Workflow Automation (WFA) database frequently, you can use the WFA command-line interface (CLI) provided with the WFA installation package.

The following are the two backup types:

- Full backup
- Regular backup

Back up (full) the WFA database using the CLI

You can perform a full backup of the OnCommand Workflow Automation (WFA) database by using the WFA command-line interface (CLI). In a full backup, the WFA database, WFA configuration, and key are backed up.

You must have admin user credentials or architect credentials.

In a high-availability environment, you should create scheduled backups by using REST APIs. You cannot create backups by using the CLI when WFA is in failover mode.

For more information, see the REST documentation.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server:
WFA_install_location\WFA\bin.

WFA_install_location is the WFA installation directory.

2. Back up the WFA database:

```
.\wfa --backup --user=USER [--password=PASS] [--location=PATH] [--full]
```

- user is the user name of the backup user.
- password is the password of the backup user.

If you have not provided the password, you must enter the password when prompted.

- path is the complete directory path to the backup file.

3. Verify that the backup file was created at the specified location.

Back up (regular) the WFA database using the CLI

You can perform a regular backup of the OnCommand Workflow Automation (WFA) database by using the WFA command-line interface (CLI). In a regular backup, only the WFA database is backed up.

You must have admin user credentials, architect credentials, or backup user credentials.

In a high-availability environment, you should create scheduled backups by using REST APIs. You cannot create backups by using the CLI when WFA is in failover mode.

For more information, see the REST documentation.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server:
WFA_install_location\WFA\bin.

WFA_install_location is the WFA installation directory.

2. Back up the WFA database:

```
.\wfa --backup --user=USER [--password=PASS] [--location=PATH]
```

- user is the user name of the backup user.
- password is the password of the backup user.

If you have not provided the password, you must enter the password when prompted.

- path is the complete directory path to the backup file.

3. Verify that the backup file was created at the specified location.

Backing up the WFA database using REST APIs

You can back up the OnCommand Workflow Automation (WFA) database by using the REST APIs. If WFA is in the failover mode in a high-availability environment, you can use the REST APIs to create scheduled backups. You cannot use the command-line interface (CLI) to create backups during a failover.

The following are the two types of backup:

- Full backup
- Regular backup

Perform a full backup of the WFA database using REST APIs

You can perform a full back up of the OnCommand Workflow Automation (WFA) database by using the REST APIs. In a full backup, the WFA database, WFA configuration, and key are backed up.

You must have admin or architect credentials.

Step

1. Enter the following URL in your web browser: <https://IP address of the WFA server/rest/backups?full=true>

For more information, see the REST documentation.

Perform a regular backup of the WFA database using REST APIs

You can perform a regular backup of the OnCommand Workflow Automation (WFA) database by using the REST APIs. In a regular backup, only the WFA database is backed up.

You must have admin, architect, or backup credentials.

Step

1. Enter the following URL in your web browser: <https://IP address of the WFA server/rest/backups>

For more information, see the REST documentation.

Restoring the OnCommand Workflow Automation database

Restoring the OnCommand Workflow Automation (WFA) database includes restoring the system configuration settings and cache information, including the playground database.

- Restoring a WFA database erases the current WFA database.
- You can restore a WFA database backup only to a system that is running the same version or a later version of WFA.

For example, if you created a backup on a system that is running WFA 4.2, the backup can be restored only to systems that are running WFA 4.2 or later.

- After the restore operation is complete, the WFA SSL certificate is replaced with the SSL certificate in the backup file.

- A comprehensive restore operation of WFA databases and configurations is required during disaster recovery, and can be used in both standalone and high-availability environments.
- A comprehensive backup cannot be created by using the web UI.



You can use only the CLI commands or REST APIs to backup and restore the WFA database comprehensively during disaster recovery.

Restore the WFA database

You can restore the OnCommand Workflow Automation (WFA) database that you backed up previously.

- You must have created a backup of the WFA database.
- You must have admin or architect credentials.
- Restoring a WFA database erases the current database.
- You can restore a WFA database backup only to a system running the same or a later version of OnCommand Workflow Automation.

For example, if you created a backup on a system running OnCommand Workflow Automation 4.2, the backup can be restored only to systems running OnCommand Workflow Automation 4.2 or later.

Steps

1. Log in to the WFA web GUI as an admin.
2. Click **Settings** and under **Maintenance**, click **Backup & Restore**.
3. Click **Choose file**.
4. In the dialog box that opens, select the WFA backup file, and click **Open**.
5. Click **Restore**.

You can review the restored content for completeness in functionality—for example, the functioning of your custom workflows.

Restoring the WFA database using the CLI

During a disaster, while recovering data you can restore the OnCommand Workflow Automation (WFA) database and supported configurations that you backed up previously using the command-line interface (CLI). The supported configurations include data access, HTTP timeout, and SSL certificates.

The following are the two types of restore:

- Full restore
- Regular restore

Restore (full) WFA database using the CLI

You can do a full restore of the OnCommand Workflow Automation (WFA) database by using the command-line interface (CLI). In a full restore, you can restore the WFA database, WFA configuration, and key.

- You must have created a backup of the WFA database.
- You must have admin or architect credentials.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server: WFA_install_location\WFA\bin
wfa_install_location is the WFA installation directory.
2. Perform the restore operation:

```
wfa.cmd --restore --full --user=user_name [--password=password] [--location=path] --restart
```

- user_name is the user name of the admin or architect user.
- password is the password of the user.

If you have not provided the password, you must enter the password when prompted.

- path is the complete directory path to the restore file.

3. Verify that the restore operation is successful and WFA is accessible.

Restore (regular) WFA database using the CLI

You can do regular restore of the OnCommand Workflow Automation (WFA) database by using the REST APIs. In a regular restore, you can only backup the WFA database.

- You must have created a backup of the WFA database.
- You must have admin credentials, architect credentials, or backup user credentials.

Steps

1. At the shell prompt, navigate to the following directory on the WFA server: WFA_install_location\WFA\bin

wfa_install_location is the WFA installation directory.

2. Perform the restore operation:

```
wfa.cmd --restore --user=user_name [--password=password] [--location=path]
```

- user_name is the user name of the admin or architect user.
- password is the password of the user.

If you have not provided the password, you must enter the password when prompted.

- path is the complete directory path to the restore file.

3. Verify that the restore operation is successful and WFA is accessible.

Restoring the WFA database using REST APIs

You can restore the OnCommand Workflow Automation (WFA) database by using REST APIs. You cannot use the command-line interface (CLI) to restore the WFA database during a failover.

The following are the two types of restore:

- Full restore
- Regular restore

Restore (full) the WFA database using REST APIs

You can do a full restore of the OnCommand Workflow Automation (WFA) database by using REST APIs. In a full restore, you can restore the WFA database, WFA configuration, and key.

- You must have created a .zip backup of the WFA database.
- You must have admin or architect credentials.
- If you are restoring the database as a part of the migration procedure, you must do a full restore.

Steps

1. Enter the following URL in the REST client browser: <https://IP address of WFA server/rest/backups?full=true>
2. In the Backup window, select the **POST** method.
3. In the **Part** drop-down list, select **Multipart Body**.
4. In the **File** field, enter the following information:
 - a. In the **Content type** drop-down list, select **multi-part/form-data**.
 - b. In the **Charset** drop-down list, select **ISO-8859-1**.
 - c. In the **File name** field, enter the name of the backup file you created and that you want to restore.
 - d. Click **Browse**.
 - e. Select the location of the .zip backup file.

5. Navigate to the WFA_install_location\wfa\bin directory, and restart the WFA services:

6. Restart the **NetApp WFA Database** and **NetApp WFA Server** service:

```
wfa --restart
```

7. Verify that the restore operation is successful and WFA is accessible.

Restore (regular) the WFA database using REST APIs

You can do a regular restore of the OnCommand Workflow Automation (WFA) database by using REST APIs. In a regular restore, you can only restore the WFA database.

- You must have created a .zip backup of the WFA database.
- You must have admin or architect credentials.
- If you are restoring the database as a part of the migration procedure, you must do a full restore.

Steps

1. Enter the following URL in the REST client browser: <https://IP address of WFA server/rest/backups>

2. In the Backup window, select the **POST** method.

3. In the **Part** drop-down list, select **Multipart Body**.

4. In the **File** field, enter the following information:

- a. In the **Content type** drop-down list, select **multi-part/form-data**.
- b. In the **Charset** drop-down list, select **ISO-8859-1**.
- c. In the **File name** field, enter the name of the backup file as **backupFile**.
- d. Click **Browse**.
- e. Select the location of the .zip backup file.

5. Navigate to the WFA_install_location\wfa\bin directory, and restart the WFA services:

6. Verify that the restore operation is successful and WFA is accessible.

Reset the admin password created during installation

If you have forgotten the password of the admin user that you created during OnCommand Workflow Automation (WFA) server installation, you can reset the password.

- You must have admin privileges for the Windows system on which you have installed WFA.
- The WFA services must be running.
- This procedure resets only the password of the admin user that was created during the WFA installation.

You cannot reset the password of other WFA admin users that you created after the WFA installation.

- This procedure does not enforce the password policy that you have configured.

You must either enter a password that complies with your password policy or change the password from

the WFA user interface after you have reset the password.

Steps

1. Open a command prompt and navigate to the following directory: `WFA_install_location\WFA\bin\`
2. Enter the following command:

```
wfa --admin-password [--password=PASS]
```

If you do not provide a password in the command, you must enter the password when prompted.

3. At the command prompt, follow the on-screen instructions to reset the admin password.

Import OnCommand Workflow Automation content

You can import user-created OnCommand Workflow Automation (WFA) content such as workflows, finders, and commands. You can also import content that is exported from another WFA installation, content that is downloaded from the Storage Automation Store or the WFA community, as well as packs, including Data ONTAP PowerShell toolkits and Perl NMSDK toolkits.

- You must have access to the WFA content that you want to import.
- The content that you want to import must have been created on a system that is running the same version or an earlier version of WFA.

For example, if you are running WFA 2.2, you cannot import content that was created using WFA 3.0.

- You can import content developed on N-2 versions of WFA only into WFA 5.1.
- If the .dar file references NetApp-certified content, the NetApp-certified content packs must be imported.

The NetApp-certified content packs can be downloaded from the Storage Automation Store. You must refer to the documentation of the pack to verify that all requirements are met.

Steps

1. Log in to WFA through a web browser.
2. Click **Settings**, and under **Maintenance** click **Import Workflows**.
3. Click **Choose File** to select the .dar file that you want to import, and then click **Import**.
4. In the Import Success dialog box, click **OK**.

Related information

[NetApp community: OnCommand Workflow Automation](#)

Considerations while importing OnCommand Workflow Automation content

You must be aware of certain considerations when you import user-created content, content that is exported from another OnCommand Workflow Automation (WFA) installation, or content that is downloaded from the Storage Automation Store or the WFA community.

- WFA content is saved as a .dar file and can include the entire user-created content from another system or specific items such as workflows, finders, commands, and dictionary terms.
- When an existing category is imported from a .dar file, the imported content is merged with the existing content in the category.

For example, consider there are two workflows WF1 and WF2 in category A in the WFA server. If workflows WF3 and WF4 in category A are imported to the WFA server, category A will contain workflows WF1, WF2, WF3, and WF4 after the import.

- If the .dar file contains dictionary entries, then the cache tables corresponding to the dictionary entries are automatically updated.

If the cache tables are not updated automatically, an error message is logged in the wfa.log file.

- When importing a .dar file that has a dependency on a pack that is not present in the WFA server, WFA tries to identify whether all the dependencies on the entities are met.
 - If one or more entities are missing or if a lower version of an entity is found, the import fails and an error message is displayed.

The error message provides details of the packs that should be installed in order to meet the dependencies.

- If a higher version of an entity is found or if the certification has changed, a generic dialog box about the version mismatch is displayed, and the import is completed.

The version mismatch details are logged in a wfa.log file.

- Questions and support requests for the following must be directed to the WFA community:
 - Any content downloaded from the WFA community
 - Custom WFA content that you have created
 - WFA content that you have modified

Migrate the OnCommand Workflow Automation installation

You can migrate an OnCommand Workflow Automation (WFA) installation to maintain the unique WFA database key that is installed during the WFA installation. For example, you can migrate the WFA installation from a Windows 2012 server to a Windows 2016 server.

- You must perform this procedure only when you want to migrate a WFA installation that includes the WFA database key to a different server.
- A WFA database restore does not migrate the WFA key.
- Migrating a WFA installation does not migrate the SSL certificates.
- The default WFA installation path is used in this procedure.

If you changed the default location during installation, you must use the changed WFA installation path.

Steps

1. Access WFA through a web browser as an admin.
2. Back up the WFA database.

3. Open a command prompt on the WFA server and change directories to the following location:
c:\Program Files\NetApp\WFA\bin
4. Enter the following at the command prompt to obtain the database key:

wfa.cmd -key
5. Note the database key that is displayed.
6. Uninstall WFA.
7. Install WFA on the required system.
8. Open the command prompt on the new WFA server and change directories to the following location:
c:\Program Files\NetApp\WFA\bin
9. Enter the following at the command prompt to install the database key:

```
wfa.cmd -key=yourdatabasekey
```

yourdatabasekey is the key that you noted from the previous WFA installation.

10. Restore the WFA database from the backup that you created.

Uninstall OnCommand Workflow Automation

You can uninstall OnCommand Workflow Automation (WFA) using Microsoft Windows Programs and Features.

Steps

1. Log in to Windows using an account with admin privileges.
2. Click **All Programs > Control Panel > Control Panel > Programs and Features**.
3. Uninstall WFA by choosing one of the following options:
 - Select **NetApp WFA** and click **Uninstall**.
 - Right-click **NetApp WFA** and select **Uninstall**.
4. If the uninstallation process stops responding before it is complete, stop the **NetApp WFA Database** service from the Windows Services console and try to uninstall again.

Managing OnCommand Workflow Automation SSL certificate

You can replace the default OnCommand Workflow Automation (WFA) SSL certificate with a self-signed certificate or a certificate signed by a Certificate Authority (CA).

The default self-signed WFA SSL certificate is generated during the installation of WFA. When you are upgrading, the certificate for the previous installation is replaced with the new certificate. If you are using a non-default self-signed certificate or a certificate signed by a CA, you must replace the default WFA SSL certificate with your certificate.

Replace the default Workflow Automation SSL certificate

You can replace the default Workflow Automation (WFA) SSL certificate if the certificate has expired or if you want to increase the validity period of the certificate.

You must have Windows admin privileges on the WFA server.

The default WFA installation path is used in this procedure. If you changed the default location during installation, you must use the custom WFA installation path.

Steps

1. Log in as the admin user on the WFA host machine.
2. Use the Windows Services console to stop the following WFA services:
 - NetApp WFA Database
 - NetApp WFA Server
3. Delete the wfa.keystore file from the following location:

```
<WFA_install_location>\WFA\jboss\standalone\configuration\keystore
```

4. Open a command prompt on the WFA server, and change directories to the following location:<OpenJDK_install_location>\bin
5. Obtain the database key:

```
keytool -keysize 2048 -genkey -alias "ssl keystore" -keyalg RSA -keystore
"<WFA_install_location>\WFA\jboss\standalone\configuration\keystore\wfa.keystore" -validity xxxx
```

xxxx is the number of days for the new certificate's validity.

6. When prompted, provide the password (default or new).

The default password is a randomly generated encrypted password.

To obtain and decrypt the default password, follow the steps in the Knowledge Base article [How to renew the self-signed certificate on WFA 5.1.1.0.4](#)

To use a new password, follow the steps in the Knowledge Base article [How to update a new password for the keystore in WFA](#).

7. Enter the required details for the certificate.
8. Review the displayed information, and enter Yes.
9. Press **Enter** when prompted by the following message: Enter key password for <SSL keystore> <RETURN if same as keystore password>.
10. Restart the WFA services by using the Windows Services console.

Create a certificate signing request for Workflow Automation

You can create a certificate signing request (CSR) in Windows so that you can use the SSL certificate that is signed by a Certificate Authority (CA) instead of the default SSL

certificate for Workflow Automation (WFA).

- You must have Windows admin privileges on the WFA server.
- You must have replaced the default SSL certificate that is provided by WFA.

The default WFA installation path is used in this procedure. If you have changed the default path during installation, then you must use the custom WFA installation path.

Steps

1. Log in as an admin user on the WFA host machine.
2. Open a command prompt on the WFA server, and then change directories to the following location:
<OpenJDK_install_location>\bin
3. Create a CSR:

```
keytool -certreq -keystore  
WFA_install_location\WFA\jboss\standalone\configuration\keystore\wfa.keystore  
-alias "ssl keystore" -file C:\file_name.csr
```

file_name is the name of the CSR file.

4. When prompted, provide the password (default or new).

The default password is a randomly generated encrypted password.

To obtain and decrypt the default password, follow the steps in the Knowledge Base article [How to renew the self-signed certificate on WFA 5.1.1.0.4](#)

To use a new password, follow the steps in the Knowledge Base article [How to update a new password for the keystore in WFA](#).

5. Send the file_name.csr file to the CA to obtain a signed certificate.

See the CA web site for details.

6. Download a chain certificate from the CA, and then import the chain certificate to your keystore: keytool -import -alias "ssl keystore CA certificate" -keystore "WFA_install_location\WFA\jboss\standalone\configuration\keystore\wfa.keystore" -trustcacerts -file C:\chain_cert.cer

C:\chain_cert.cer is the chain certificate file that is received from the CA. The file must be in the X.509 format.

7. Import the signed certificate that you received from the CA:

```
keytool -import -alias "ssl keystore" -keystore  
"WFA_install_location\WFA\jboss\standalone\configuration\keystore\wfa.keystore  
" -file C:\certificate.cer
```

C:\certificate.cer is the chain certificate file that is received from the CA.

8. Start the following WFA services:

- NetApp WFA Database

- NetApp WFA Server

Managing Perl and Perl modules

OnCommand Workflow Automation (WFA) supports Perl commands for workflow operations. ActivePerl 5.26.3 is installed and configured on the WFA server when you install WFA. You can install and configure your preferred Perl distribution and Perl modules.

In addition to ActivePerl, the required Perl modules from the NetApp Manageability SDK are also installed when you install WFA. The NetApp Manageability SDK Perl modules are required for successful execution of Perl commands.

Configure your preferred Perl distribution

By default, ActivePerl is installed with OnCommand Workflow Automation (WFA). If you want to use another Perl distribution, you can configure your preferred Perl distribution to work with WFA.

You must have installed the required Perl distribution on the WFA server.

You must not uninstall or overwrite the default ActivePerl installation. You must install your preferred Perl distribution at a separate location.

Steps

1. Open Windows Explorer and navigate to the following directory:

```
WFA_install_location\WFA\bin\
```

2. Double-click the ps.cmd file.

A PowerShell command-line interface (CLI) prompt opens with ONTAP and WFA modules loaded in it.

3. At the prompt, enter the following:

```
Set-WfaConfig -Name CustomPerl -PerlPath CustomPerlPath
```

```
Set-WfaConfig -Name CustomPerl -PerlPath C:\myperl\perl.exe
```

4. When prompted, restart the WFA services.

Manage site-specific Perl modules

You can use the ActiveState Perl Package Manager (PPM) to manage your site-specific Perl modules. You must install your site-specific Perl modules outside the OnCommand Workflow Automation (WFA) installation directory to avoid deletion of your Perl modules during a WFA upgrade.

Using the PERL5LIB environment variable, you can configure the Perl interpreter installed on the WFA server to use your Perl modules.

Installation of the Try-Tiny Perl module in the user area at c:\Perl is used as an example in this procedure. This user area is not deleted when you uninstall WFA, and you can reuse the area after WFA is reinstalled or upgraded.

Steps

1. Set the PERL5LIB environment variable to the location where you want to install your Perl modules.

```
c:\>echo %PERL5LIB% c:\Perl
```

2. Verify that the Perl module area is not initialized by using ppm area list.

```
c:\Program Files\NetApp\WFA\Perl64\bin>ppm area list
```

name	pkgs	lib
(user)	n/a	C:/Perl
site*	0	C:/Program Files/NetApp/WFA/Perl64/site/lib
perl	229	C:/Program Files/NetAPP/WFA/Perl64/lib

3. Initialize the Perl module area by using ppm area init user.

```
c:\Program Files\NetApp\WFA\Perl64\bin>ppm area init user
```

```
Syncing user PPM database with .packlists...done
```

4. Verify that the Perl module area is initialized by using ppm area list.

```
c:\Program Files\NetApp\WFA\Perl64\bin>ppm area list
```

name	pkgs	lib
user	0	C:/Perl
site*	0	C:/Program Files/NetApp/WFA/Perl64/site/lib
perl	229	C:/Program Files/NetAPP/WFA/Perl64/lib

5. Add the required repositories and install the required packages.

a. Add the required repository by using ppm repo add.

```
c:\Program Files\NetApp\WFA\Perl64\bin>ppm repo add  
http://ppm4.activestate.com/MSWin32-x64/5.16/1600/package.xml
```

```
Downloading ppm4.activestate.com packlist...done  
Updating ppm4.activestate.com database...done  
Repo 1 added.
```

b. Verify that the required repository is added by using ppm repo list.

```
c:\Program Files\NetApp\WFA\Perl64\bin>ppm repo list
```

id	pkgs	name
1	17180	ppmr.activestate.com

(1 enabled repository)

c. Install the required Perl module by using ppm install.

```
c:\Program Files\NetApp\WFA\Perl64\bin>ppm install Try-Tiny --area user
```

```
Downloading ppm4.activestate.com packlist...done  
Updating ppm4.activestate.com database...done  
Downloading Try-Tiny-0.18...done  
Unpacking Try-Tiny-0.18...done  
Generating HTML for Try-Tiny-0.18...done  
Updating files in user area...done  
2 files installed
```

d. Verify that the required Perl module is installed by using ppm area list.

```
c:\Program Files\NetApp\WFA\Perl64\bin>ppm area list
```

name	pkgs	lib
user	1	C:/Perl
site*	0	C:/Program Files/NetApp/WFA/Perl64/site/lib
perl	229	C:/Program Files/NetAPP/WFA/Perl64/lib

Repair the ActivePerl installation

ActiveState ActivePerl is installed on your OnCommand Workflow Automation (WFA) server when you install WFA. ActivePerl is required for the execution of Perl commands. If you inadvertently uninstall ActivePerl from the WFA server or if the ActivePerl installation is corrupted, you can manually repair the ActivePerl installation.

Steps

1. Back up the WFA database using one of the following options:
 - WFA web portal
 - PowerShell script
2. Uninstall WFA.
3. Install the version of WFA that you uninstalled.

ActivePerl is installed when you install WFA.

4. Restore the WFA database.

You can review the restored content for completeness in functionality—for example, the functioning of your custom workflows.

Troubleshooting installation and configuration issues

You can troubleshoot issues that might occur while installing and configuring OnCommand Workflow Automation (WFA).

Cannot open the OnCommand Workflow Automation login page

If you have installed .Net 3.5, the Internet Information Services (IIS) is installed with it. The IIS occupies port 80, which is used by WFA.

Ensure that either the IIS role is removed or IIS is disabled in the WFA server.

Cannot view Performance Advisor data in WFA

If you cannot view Performance Advisor data in WFA or if the data acquisition process from the Performance Advisor data source fails, you should perform certain actions to troubleshoot the issue.

- Ensure that you have specified the credentials of an Active IQ Unified Manager user with a minimum role of GlobalRead when configuring Performance Advisor as a data source in WFA.
- Ensure that you have specified the correct port when configuring Performance Advisor as a data source in WFA.

By default, Active IQ Unified Manager uses port 8088 for an HTTP connection and port 8488 for an HTTPS connection.

- Ensure that performance data is collected by the Active IQ Unified Manager server.

OnCommand Workflow Automation (WFA) displays a blank page on Windows 2012

A blank page might be displayed if you have downloaded and installed Adobe Flash Player separately from the Adobe website. You must not download and install the Flash Player separately because it is bundled with Internet Explorer in Windows 2012. Updates for the Flash Player are installed through Windows updates.

If you have downloaded and installed the Flash Player separately, you must perform the following steps:

Steps

1. Uninstall the Flash Player that you have already installed.
2. In Windows, open **Server Manager** > **Local Server** > **ROLES AND FEATURES** > **TASKS** and select **Add Roles and Features**.
3. In the Add Roles and Features Wizard, click **Features** > **User Interface and Infrastructure**, select **Desktop Experience** and then complete adding the feature.

Adding Desktop Experience adds the Flash Player to Windows.

4. Restart Windows.

Related documentation for OnCommand Workflow Automation

There are additional documents and tools to help you learn to perform more advanced configuration of your OnCommand Workflow Automation (WFA) server.

Other references

The Workflow Automation space within the NetApp community provides additional learning resources, including the following:

- **NetApp community**

[NetApp community: Workflow Automation \(WFA\)](#)

Tool references

- **Interoperability Matrix**

Lists supported combinations of hardware components and software versions.

[Interoperability Matrix](#)

Workflow

Overview of OnCommand Workflow Automation

OnCommand Workflow Automation (WFA) is a software solution that helps to automate storage management tasks, such as provisioning, migration, decommissioning, data protection configurations, and cloning storage. You can use WFA to build workflows to complete tasks that are specified by your processes.

A workflow is a repetitive and procedural task that consists of sequential steps, including the following types of tasks:

- Provisioning, migrating, or decommissioning storage for databases or file systems
- Setting up a new virtualization environment, including storage switches and datastores
- Setting up storage for an application as part of an end-to-end orchestration process

Storage architects can define workflows to follow best practices and meet organizational requirements, such as the following:

- Using required naming conventions
- Setting unique options for storage objects
- Selecting resources
- Integrating internal configuration management database (CMDB) and ticketing applications

WFA features

- Workflow design portal to build workflows

The workflow design portal includes several building blocks, such as commands, templates, finders, filters, and functions, that are used to create workflows. The designer enables you to include advanced capabilities to workflows such as automated resource selection, row repetition (looping), and approval points.

The workflow design portal also includes building blocks, such as dictionary entries, cache queries, and data source types, for caching data from external systems.

- Execution portal to execute workflows, verify status of workflow execution, and access logs
- Administration/Settings option for tasks such as setting up WFA, connecting to data sources, and configuring user credentials
- Web service interfaces to invoke workflows from external portals and data center orchestration software
- Storage Automation Store to download WFA packs. The ONTAP 9.7.0 pack is bundled with WFA 5.1.

WFA license information

No license is required for using the OnCommand Workflow Automation server.

Understanding Workflow Automation designer

You create workflows in the Workflow Automation (WFA) designer using the building blocks such as finders, filters, and commands. Understanding the building blocks and the workflow creation process is important before you start creating your workflows.

Working with the building blocks in OnCommand Workflow Automation

The Workflow Automation (WFA) workflows consist of several building blocks and WFA includes a library of the predefined building blocks. You can use the building blocks that WFA provides to create workflows that match the requirements of your organization.

WFA provides the structure for storage automation processes. WFA's flexibility is based on how the workflows are constructed by using the workflow building blocks.

The WFA building blocks are as follows:

- Dictionary entries
- Commands
- Filters
- Finders
- Functions
- Templates

You should understand how the building blocks are used in WFA to help you in creating the workflows.

What data sources are

A data source is a method for establishing a connection to other systems, files and databases in order to extract data. For example, a data source can be a connection to an Active IQ Unified Manager database of Active IQ Unified Manager 9.4 data source type.

You can add a custom data source to OnCommand Workflow Automation (WFA) for data acquisition after defining the required data source type by associating the caching scheme, the required port, and the acquisition method with the data source type.

WFA caches information through various data sources. WFA collects resource information from the data sources and formats it for the caching scheme. The cache tables, which are the tables inside the caching schemes, are formatted to match the dictionary entry objects. When you use a finder in workflows, it returns a dictionary object and the data from the dictionary object is populated from the cache tables. The process of acquiring data from the data sources is known as *data source acquisition*. You can use either a script-based method or a driver-based method for data source acquisition. The sources can be different from each other and data source acquisition might sample them at different time intervals. WFA then merges that information into the database and superimposes the reservation data to maintain updated resource information in the database.

The WFA database includes several different caching schemes. A caching scheme is a set of tables and each table includes information from a certain dictionary entry type; however, the tables might include combined information from multiple sources of a specific data source type. WFA uses the database information to understand the status of the resources, perform calculations, and execute commands on the resources.

What dictionary entries are

Dictionary entries are one of the OnCommand Workflow Automation (WFA) building blocks. You can use dictionary entries to represent object types and their relationships in your storage and storage-related environments. You can then use filters in workflows to return the value of the natural keys of the dictionary entries.

A dictionary entry is the definition of an object type that is supported by WFA. Each dictionary entry represents an object type and its relationship in the supported storage and storage-related environments. A dictionary object consists of a list of attributes, which might be type checked. A dictionary object with complete values describes an object instance of a type. In addition, the reference attributes describe the relationship of the object with the environment; for example, a volume dictionary object has many attributes, such as name, size_mb, and volume_guarantee. In addition, the volume dictionary object includes references to the aggregate and the array containing the volume in the form of array_id and aggregate_id.

The cache table of an object is a database containing a few or all of the dictionary entry's attributes that are marked for caching. For a dictionary entry to include a cache table, at least one of the dictionary entry's attributes must be marked for caching. Dictionary entries include natural keys, which are unique identifiers for the objects; for example, 7-Mode volumes are identified uniquely by their name and the IP address of the array containing them. Qtrees are identified by the qtree name, the volume name, and the array IP address. You must identify the dictionary attributes that are part of the dictionary entry's natural keys when creating dictionary entries.

How commands work

OnCommand Workflow Automation commands are the execution blocks for workflows. You can use a command for each step in your workflow.

WFA commands are written using PowerShell and Perl scripts. PowerShell commands use the Data ONTAP PowerShell toolkit and VMware PowerCLI, if the package is installed. Perl commands use the Perl distribution and Perl modules installed on the WFA server. If you include multiple scripting languages in a command, such as PowerShell and Perl, the appropriate script is chosen by WFA based on the operating system on which it is installed and the preferred order of language you have specified in the WFA configuration menu.

The scripts for the WFA commands include several parameters. These parameters might be mapped to dictionary entry attributes.

Note that each WFA command can include several Data ONTAP commands.

Some of the WFA commands are known as wait commands because they can wait for long-running operations and poll periodically—for example, the **Wait for multiple volume moves** command. The waiting interval at which the polling command is executed can be configured to check if the operation has been completed.

A WFA command is initiated by WFA while the workflow is in its execution phase. WFA executes the commands serially, in left-to-right and top-to-bottom order. The planning of the workflow confirms the availability and validity of the parameters that are supplied to the command. The WFA server supplies all the parameters required for the commands before the commands are executed.

Parameters to commands are finalized during workflow planning. The workflow then passes these parameters to the commands during execution time. The commands cannot pass parameters back to the workflow. However, if you want to exchange information that is obtained during execution time between commands in a workflow, you can use the designated WFA PowerShell cmdlets or Perl functions.

WFA PowerShell commands do not use the -ErrorAction stop flag for the PowerShell cmdlets; therefore,

workflow executions continue even when the cmdlets fail because of an error. If you want the `-ErrorAction stop` flag to be included in a specific command, you can clone the command and modify the PowerShell script to add the flag.

The following are the PowerShell cmdlets and Perl functions that are included in WFA to enable exchange of information between commands:

PowerShell cmdlets	Perl functions
Add-WfaWorkflowParameter	addWfaWorkflowParameter
Get-WfaWorkflowParameter	getWfaWorkflowParameter

Parameters added by the “add” cmdlets or functions to a command can be retrieved by a command that is executed subsequently and uses the “get” cmdlets or functions. For example, in a PowerShell WFA command, you can use the following in the code to add a parameter named `volumId`: `Add-WfaWorkflowParameter -Name "VolumeUUID" -Value "12345" -AddAsReturnParameter $true`. Then, you can use the following in a subsequent command to retrieve the value of `volumId`: `$volumeId = Get-WfaWorkflowParameter -Name volumeId`.

WFA commands can query the WFA database and obtain the required result. This enables you to construct a command without using filters and finders. You can use the following functions to query the database:

PowerShell cmdlet	Perl function
<code>Invoke-MySqlQuery</code> For example: <code>Invoke-MySqlQuery -Query "SELECT cluster.name AS 'Cluster Name' FROM cm_storage.cluster"</code>	<code>invokeMySqlQuery</code>

Related information

[Create a filter](#)

[Create a finder](#)

[Create a command](#)

What filters are

You can use WFA filters in your workflows to select the required resources.

A WFA filter is an SQL-based query that works on the WFA database. Each filter returns a list of elements of a specific dictionary type. The returned elements are based on the selection criteria specified in the SQL query. You must be aware of SQL syntaxes to create or edit a filter.

What finders are

A finder is a combination of one or more filters that are used together to identify common results. You can use a finder in your workflows to select the required resources for workflow execution.

Finders might apply a sorting order to differentiate the applicable results. Finders return the best resource based on the selection criteria and sorting.

Finders return either one result or no result; therefore, they can be used to verify the existence of certain storage elements. However, when a finder is used as part of a repeat row definition, the result sets are used to form the list of members in the group. Filters that are used in finders return the natural key of the dictionary type, at a minimum, but might return additional fields, whose value can be referenced. A sorting order might be applied to any returned field of a filter's SQL query.

You can test the results of a finder. When testing a finder, you can view the common results of all the WFA filters, where the effective result of the finder is highlighted in the results. When using a finder in a workflow, you can create a customized error message to convey meaningful information to the storage operator.

What functions are

You can use a function in your workflows for a complex task that has to be completed during the planning phase of the workflow.

You can write functions by using the MVFLEX Expression Language (MVEL). You can use functions to put together commonly used logic as well as more complex logic in a named function and reuse it as values for command parameters or filter parameters. You can write a function once and use it across workflows. You can use functions to handle repetitive tasks and tasks that might be complex, such as defining a complex naming convention.

Functions might use other functions during their execution.

What schemes are

A scheme represents the data model for a system. A data model is a collection of dictionary entries. You can define a scheme and then define a data source type. The data source defines how the data is acquired and the scheme is populated. For example, a vc scheme acquires data about your virtual environment, such as virtual machines, hosts, and datastores.

Schemes can also be populated directly with data through workflows that are customized to solve specific problems.

Dictionary entries are associated with an existing scheme when the dictionary entries are created. Dictionary entries are also associated with cache queries, and cache queries include SQL queries.

Schemes can acquire data using either script based data source type or SQL data source type. The scripts are defined while creating the data source type and SQL queries are defined in the cache queries.

The following schemes are included in WFA:

- 7-Mode (storage)**

Scheme to acquire data through Active IQ Unified Manager from Data ONTAP operating in 7-Mode.

- Clustered Data ONTAP (cm_storage)**

Scheme to acquire data through Active IQ Unified Manager from clustered Data ONTAP.

- **7-Mode Performance (performance)**

Scheme to acquire performance data of Data ONTAP operating in 7-Mode through Performance Advisor.

- **Clustered Data ONTAP Performance (cm_performance)**

Scheme to acquire performance data of clustered Data ONTAP through Performance Advisor.

- **VMware vCenter (vc)**

Scheme to acquire data from VMware vCenter.

- **Playground (playground)**

Scheme that can you can directly populate with data.

What remote system types are

OnCommand Workflow Automation (WFA) communicates with remote system types. A remote system type specifies the type of remote systems with which WFA can communicate. You can configure remote system types in WFA. For example, Data ONTAP system can be configured as a remote system type.

A remote system type has the following attributes:

- Name
- Description
- Version
- Protocol
- Port
- Timeout

You can have a Perl script for each remote system type to validate the credentials of the remote system. You can store the credentials for the remote systems configured on WFA. You can add or edit a new custom remote system type. You can also clone an existing remote system type. You can delete a remote system type only if no systems are associated with it.

How you use templates

You can use WFA templates in your workflows as a reference or for adhering to usage policies.

A WFA template acts as a blueprint of an object definition. You can define a template by including the properties of an object and the values for the object's properties. Then, you can use the template for populating the properties of an object definition in your workflows.

When you use a template, you cannot edit the fields that include the values that are obtained from the template. Therefore, you can use templates for setting up usage policies and creation of objects. If you remove the association of a template with the workflow after you have applied the template, the values populated from the template remain, but you can edit the fields.

How you use categories

You can categorize your workflows to better organize the workflows and to apply access control capability on the workflows.

You can categorize workflows such that they appear in specific groups on the WFA portal. You can also apply access control capability on workflow categories. For example, you can allow only certain storage operators or approvers to view certain categories of workflows. Storage operators or approvers can execute only the workflows within the category for which they have been granted access rights.

Active Directory groups also can be used for access control to categories.

How entity versioning works

The OnCommand Workflow Automation (WFA) entities, such as commands and workflows, are versioned. You can use the version numbers to easily manage changes to the WFA entities.

Each WFA entity includes a version number in the `major.minor.revision` format—for example, 1.1.20. You can include up to three digits in each part of the version number.

Before modifying the version number of a WFA entity, you must be aware of the following rules:

- Version numbers cannot be changed from the current version to an earlier version.
- Each part of the version must be a number from 0 through 999.
- New WFA entities are versioned as 1.0.0, by default.
- An entity's version number is retained when cloning or using **Save As** to save a copy of the entity.
- Multiple versions of an entity cannot exist in a WFA installation.

When you update the version of a WFA entity, the version of its immediate parent entity is updated automatically. For example, updating the version of the **Create Volume** command updates the **Create an NFS Volume** workflow, because the **Create an NFS Volume** workflow is an immediate parent entity of the **Create Volume** command. The automatic update to versions is applied as follows:

- Modifying the major version of an entity updates the minor version of its immediate parent entities.
- Modifying the minor version of an entity updates the revision version of its immediate parent entities.
- Modifying the revision version of an entity does not update any part of the version of its immediate parent entities.

The following table lists the WFA entities and their immediate parent entities:

Entity	Immediate parent entity
Cache query	<ul style="list-style-type: none">• Data source type
Template	<ul style="list-style-type: none">• Workflow

Entity	Immediate parent entity
Function	<ul style="list-style-type: none"> Workflow Template <p> If a function contains special or mixed case characters, the version of its immediate parent entities might not be updated.</p>
Dictionary	<ul style="list-style-type: none"> Template Filter Cache query Command Data source types which are using script method
Command	<ul style="list-style-type: none"> Workflow
Filter	<ul style="list-style-type: none"> Finder Workflow
Finder	<ul style="list-style-type: none"> Workflow
Data source type	None
Workflow	None

You can search for an entity in WFA either using the parts of the version number or the complete version number.

If you delete a parent entity, the child entities are retained and their version is not updated for the deletion.

How versioning works when importing entities

If you import entities from versions earlier than Workflow Automation 2.2, the entities are versioned as 1.0.0, by default. If the imported entity is already present in the WFA server, the existing entity is overwritten with the imported entity.

The following are the potential changes to WFA entities during an import:

- Upgrade of entities

The entities are replaced with a later version.

- Rollback of entities

The entities are replaced with an earlier version.



When you perform a rollback of an entity, the version of its immediate parent entities are updated.

- Import of new entities



You cannot selectively import entities from a .dar file.

If a later version of an entity is imported, the version of its immediate parent entities is updated.

If there are multiple child entities to the imported parent entity, only the highest degree of change (major, minor, or revision) to the child entities is applied to the parent entity. The following examples explain how this rule works:

- For an imported parent entity, if there is one child entity with a minor change and another child entity with a revision change, the minor change is applied to the parent entity.

The revision part of the parent's version is incremented.

- For an imported parent entity, if there is one child entity with a major change and another child entity with a minor change, the major change is applied to the parent entity.

The minor part of the parent's version is incremented.

Example of how the versions of imported child entities affect the parent's version

Consider the following workflow in WFA: "Create Volume and export using NFS - Custom" 1.0.0.

The existing commands included in the workflow are as follows:

- "Create Export Policy - Custom" 1.0.0
- "Create Volume - Custom" 1.0.0

The commands included in the .dar file, which is to be imported, are as follows:

- "Create Export Policy - Custom" 1.1.0
- "Create Volume - Custom" 2.0.0

When you import this .dar file, the minor version of the "Create Volume and export using NFS - Custom" workflow is incremented to 1.1.0.

What a playground database is

The playground database is a MySQL database, which is included in the Workflow Automation (WFA) server installation. You can add tables to the playground database to include information, which can be used by filters and SQL queries for user inputs.

The playground database is a schema that cannot be accessed through the WFA web portal. You can use a MySQL client, such as SQLyog, Toad for MySQL, and MySQL Workbench or a command-line interface (CLI), to access the database.

You must use the following credentials to access the playground database:

- User name: wfa
- Password: Wfa123

The credentials provide complete access to the playground database and read-only access to other schemas defined in the WFA MySQL database. You can create the required tables in the playground database.

You can add the tags or metadata that you are using for storage objects in your environment to a table in the playground database. The tags or metadata can then be used along with the information in other WFA cache tables by WFA filters and user input queries.

For example, you can use the playground database for the following use cases:

- Tagging aggregates with business unit (BU) name and allocating volumes to the BUs based on these tags
- Tagging vFiler units with BU names
- Adding geography or location details to storage objects
- Defining access of database admins to databases

For example, if you are using the name of the BU as a tag for the storage objects, such as aggregates and vFiler units, you can create a table in the playground database that includes the name of the BU. The BU name can then be used by filters and user input queries for your workflows.

The following is an example playground database table (playground.volume_bu):

array_ip	volume_name	BU
10.225.126.23	data_11	Marketing
10.225.126.28	arch_11	HR

The following is an example SQL query that you can use to filter volumes by BU:

```

SELECT
    vol.name,
    array.ip AS 'array.ip'
FROM
    storage.volume AS vol,
    storage.array AS array,
    playground.volume_bu AS vol_bu
WHERE
    vol.array_id = array.id
    AND array.ip = vol_bu.array_ip
    AND vol.name = vol_bu.volume_name
    AND vol_bu.bu = '{$bu}'

```

Related information

[SQLyog](#)

Managing workflows

You can customize predefined workflows or create new workflows as part of managing your workflows. You must also understand the relevant concepts before you start managing your workflows.

Customize predefined workflows

You can customize a predefined Workflow Automation (WFA) workflow if there is no predefined workflow that is suitable for your requirement.

What you'll need

You must have identified the required modifications for the predefined workflow.

About this task

Questions and support request for the following must be directed to the WFA community:

- Any content downloaded from the WFA community
- Custom WFA content that you have created
- WFA content that you have modified

Steps

1. Click **Workflow Design > Workflows**.
2. Select the predefined workflow that closely matches your requirement, and then click  on the toolbar.
3. In the workflow designer, make the required changes in the appropriate tabs, such as editing the description, adding or deleting a command, modifying the command details, and modifying the user input.
4. Click **Preview**, enter the required user inputs to preview the workflow execution, and then click **Preview** to view the planning details of the workflow.
5. Click **OK** to close the preview window.
6. Click **Save**.

After you finish

You can test the workflow that you modified in your test environment, and then mark the workflow as ready for production.

Customize the Create a Volume and a CIFS Share workflow

You can customize your workflows based on your requirements. For example, you can modify the predefined *Create a Volume and a CIFS Share* workflow to include deduplication and compression.

About this task

The customization and illustrations in this task are examples; you can modify the WFA workflows based on your requirements.

Steps

1. Click **Workflow Design > Workflows**.
2. Select the **Create a Volume and a CIFS Share** workflow, and then click  on the toolbar.
3. Click the **Details** tab and edit the description of the workflow in the **Workflow name** field.
4. Click the **Workflow** tab, expand the **storage** schema, and then drag and drop the **Setup deduplication and compression** command in between the **Create volume** and **Create CIFS share** commands.
5. Place your mouse cursor below the **Setup deduplication and compression** command on the first row and then click .
6. In the **Volume** tab of the **Parameters for 'Setup deduplication and compression'** dialog box, select the **by using a previously defined Volume** option, and then select the **share_volume** option in the **Define Volume** field, which is the Volume object variable created by the **Create Volume** command in the workflow.
7. Click the **Other Parameters** tab and perform the following steps:
 - a. Select **true** in the **StartNow** field.
 - b. Select **Inline** in the **Compression** field.
 - c. Enter '`sun-sat@1`' expression in the **Schedule** field, which schedules deduplication and compression on all days of the week at 1 a.m.
8. Click **OK**.
9. Click **Preview** to ensure that the planning of the workflow is completed successfully, and then click **OK**.
10. Click **Save**.

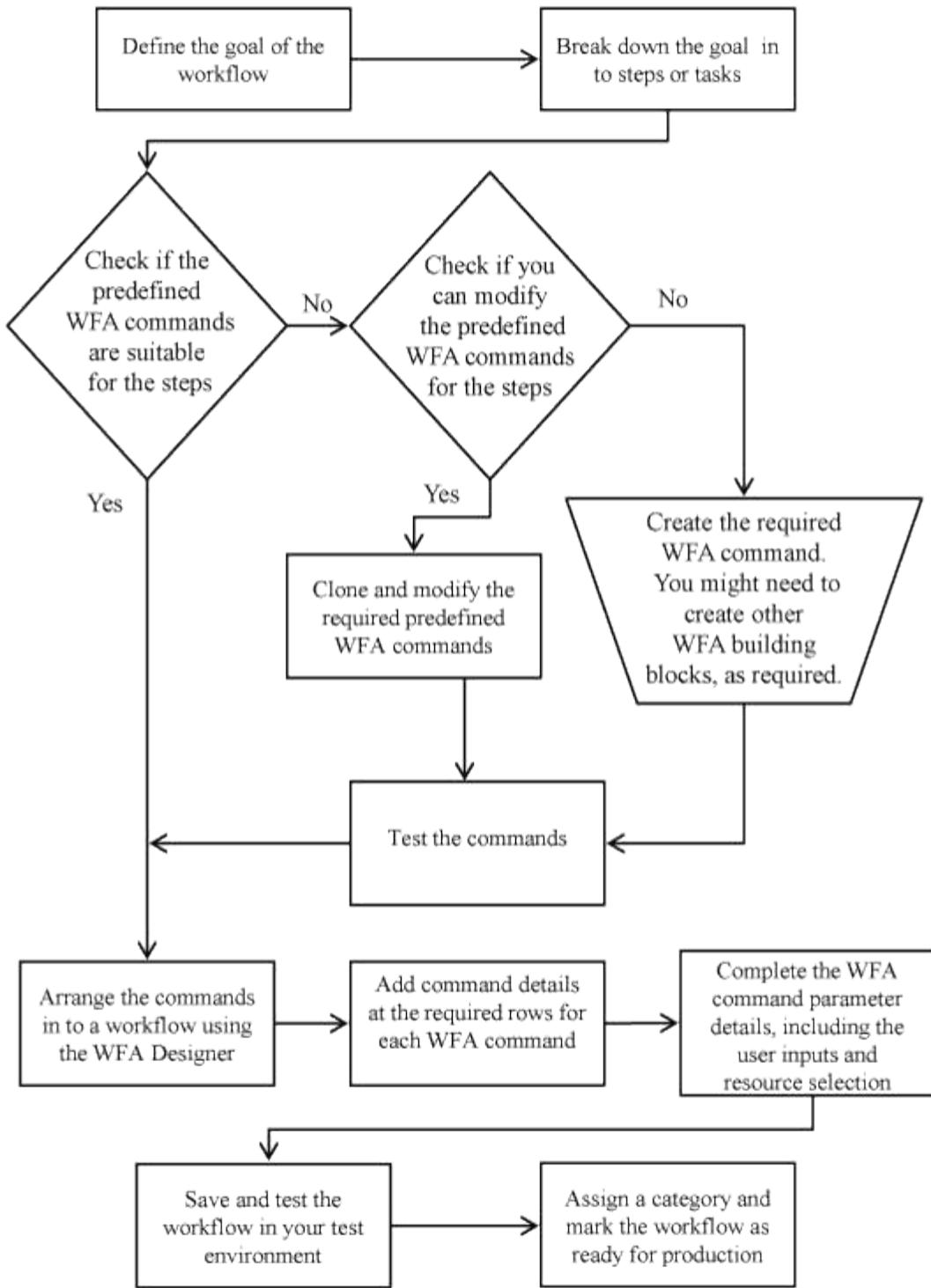
Creating workflows

If the predefined workflows do not match your requirements, you can create the required workflow. Before you create your workflows, you should understand the capabilities available in the WFA designer and create a workflow checklist.

Tasks involved in creating workflows

Creating storage automation workflows in OnCommand Workflow Automation (WFA) includes defining the steps to be performed by a workflow and creating the workflow using the WFA building blocks, such as commands, finders, filters, and dictionary entries.

The following flowchart illustrates the workflow creation process:



How you define workflows

You must break down the goal of a workflow into the steps that should be executed by the workflow. You can then arrange the steps to complete your workflow.

A workflow is an algorithm that includes a series of steps that are required to complete an end-to-end process. The scope of the process might vary, depending on the goal of the workflow. The goal of a workflow might be defined to handle only storage operations or more complex processes such as handling networking, virtualization, IT systems, and other applications as part of a single process. OnCommand Workflow Automation (WFA) workflows are designed by storage architects and are executed by storage operators.

Defining your workflow includes breaking down the goal of your workflow into a series of steps—for example, creating an NFS volume includes the following steps:

1. Creating a volume object
2. Creating a new export policy and associating the policy with the volume

You can use a WFA command or a workflow for each step in your workflow. WFA includes predefined commands and workflows, which are based on common storage use cases. If you do not find a predefined command or workflow that can be used for a particular step, you can do one of the following:

- Choose a predefined command or workflow that closely matches the step, and then clone and modify the predefined command or workflow according to your requirements.
- Create a new command or workflow.

You can then arrange the commands or workflows in a new workflow to create the workflow that accomplishes your goal.

At the beginning of the workflow execution, WFA plans the execution and verifies that the workflow can be executed using the input to the workflow and the commands. When planning the workflow, all resource selection and user input are resolved to create an execution plan. After planning is completed, WFA executes the execution plan, which consists of a set of WFA commands with applicable parameters.

How user inputs are defined

The OnCommand Workflow Automation (WFA) user inputs are data input options that are available during the execution of workflows. You must define the user input parameters for your workflows to enhance the flexibility and usability of your workflows.

User inputs are shown as input fields, which can be filled out with relevant data when previewing or executing workflows. You can create a user input field when specifying the command details in a workflow by prefixing a label or variable with the dollar sign (\$). For example, \$VolumeName creates a Volume Name user input field. WFA automatically populates the User Inputs tab in the Workflow <workflow name> window with the user input labels that you have created. You can also define the type of the user input and customize the input fields by modifying the user input attributes, such as type, display name, default values, and validation values.

User input type options

- **String**

You can use a regular expression for valid values—for example, a*.

Strings, such as 0d and 0f, are evaluated as numbers similar to 0d evaluated as 0 of type double.

- **Number**

You can define a numerical range that can be selected—for example, 1 through 15.

- **Enum**

You can create enumeration values that can be selected when filling the user input field using the enum type. You can optionally lock the enum values that you have created to ensure that only the values you have created are selected for the user input.

- **Query**

You can select the query type when you want the user input to be selected from the values available in the WFA cache. For example, you can use the following query to automatically populate the user input fields with the IP address and name values from the WFA cache: `SELECT ip, name FROM storage.array`. You can optionally lock the values retrieved by a query so that only the results retrieved by the query are selected.

- **Query (multi-select)**

The query (multi-select) type, which is similar to the query type, enables the selection of multiple values during the execution of the workflow. For example, users can select multiple volumes or a volume together with its shares and exports. You can allow the users to select multiple rows, or restrict the selection to a single row. Selecting a row selects the values from all the columns of the selected row.

You can use the following functions when using the query (multi-select) type of user input:

- `getSize`
- `getValueAt`
- `getValueAt2D`
- `getValueFrom2DByRowKey`
 - **Boolean**

You can use the Boolean type to display a check box in the user input dialog box. You must use the Boolean type for user inputs that have “true” and “false” as the possible values.

- **Table**

You can use the table type of user input to specify the column headers of a table that can be used to enter multiple values during the execution of the workflow. For example, a table that can be used to specify a list of node names and port names. You can also specify one of the following user input types for the column headers to validate the values that are entered during run time:

- `String`
- `Number`
- `Enum`
- `Boolean`
- `Query`

`String` is the default user input type for the column headers. You must double-click the Type column to specify a different user input type.

You can open the Create SnapMirror policy and rules workflow in the Designer to see how the user input types are used in the “`SnapMirrorPolicyRule`” user input.

You can use the following functions when using the table type of user input:

- `getSize`
- `getValueAt`
- `getValueAt2D`

- `getValueFrom2DByKey`

You can open the **Create and configure a Storage Virtual Machine with Infinite Volume** workflow in the Designer to see how the table type is used.

- **Password**

You can use the password type for user inputs that are meant for entering passwords. The password entered by the user is encrypted and displayed as a sequence of asterisk characters across the WFA application and in the log files. You can use the following functions to decrypt the password, which can then be used by the command:

- For Perl commands: `WFAUtil::getWfaInputPassword ($password)`
- For PowerShell commands: `Get-WfaInputPassword -EncryptedPassword $password`

Here, `$password` is the encrypted password that is passed by WFA to the command.

- **Dictionary**

You can add the table data for the selected dictionary entry. The dictionary entry attribute selects the attribute that is to be returned. You can select a single value or multiple values while executing the workflow. For example, you can select a single volume or multiple volumes. By default, single values are selected. You can also select Rules for filtering. A rule consists of a dictionary entry attribute, an operator, and a value. The attribute can also include attributes of its references.

For example, you can specify a rule for aggregates by listing all aggregates with name starting with the string “aggr” and have an available size greater than 5 GB. The first rule in the group is the attribute name, with the operator `starts-with`, and the value `aggr`. The second rule for the same group is the attribute `available_size_mb`, with the operator `>` and the value `5000`.

The following table lists the options that you can apply to the user input types:

Option	Description
Validating	<p>You can validate the user inputs type so that only valid values are entered by users:</p> <ul style="list-style-type: none"> • The string and number types of user input can be validated with the values entered during run time of the workflow. • The string type can also be validated with a regular expression. • The number type is a numeric floating-point field and can be validated using a specified numeric range.
Locking values	<p>You can lock the values of the query and enum types to prevent the user from overwriting the drop-down values and to enable the selection of only the displayed values.</p>

Option	Description
Marking as mandatory	You can mark user inputs as mandatory so that the users must enter certain user inputs in order to continue with the execution of the workflow.
Grouping	You can group related user inputs and provide a name for the user input group. The groups can be expanded and collapsed in the user input dialog box. You can select a group that should be expanded by default.
Applying conditions	With the conditional user input capability, you can set the value of a user input based on the value that is entered for another user input. For example, in a workflow that configures the NAS protocol, you can specify the required user input for protocol as NFS to enable the “Read/Write host lists” user input.

How you map command parameters

The parameters in Workflow Automation (WFA) commands are mapped to specific attributes and dictionary entry references based on certain rules. You must be aware of the rules to map command parameters when you create or edit a WFA command.

Command parameter mapping defines how command details are defined in the workflows. Mapped command parameters of a command are displayed in tabs when you are specifying the command details for commands in workflows. The tabs are named based on the group name specified in the Object Name column of the Parameters Mapping tab. The parameters that are not mapped are displayed in the Other Parameters tab when you are specifying the command details in workflows.

The rules for command parameter mapping are applicable based on the command category and how the commands are represented in the workflow editor.

The following are the command categories:

- Commands that create objects
- Commands that update objects
- Commands that remove objects
- Commands that deal with optional parent and child objects
- Commands that update associations between objects

The rules are listed below for each category:

All command categories

When mapping a command parameter, you should use the natural path based on how the command is used in workflows.

The following examples show how you can define a natural path:

- For the ArrayIP parameter, depending on the command, you should use the aggregate.array.ip attribute of the Volume dictionary entry and not the array.ip attribute.

This is important when a workflow creates a volume and then performs an additional step with the created volume by referring to it. The following are similar examples:

- volume.aggregate.array.ip of the Qtree dictionary entry
- volume.aggregate.array.ip of the LUN dictionary entry
- For Cluster used in commands, you should use one of the following:
 - vserver.cluster.primary_address of the Volume dictionary entry
 - volume.vserver.cluster.primary_address of the Qtree dictionary entry

Commands that create objects

This category of commands is used for one of the following:

- Finding a parent object and defining new objects
- Searching for an object and creating the object if the object does not exist

You should use the following parameter mapping rules for this category of commands:

- Map the relevant parameters of the object that is created to the object's dictionary entry.
- Map the parent object through the references of the dictionary entry that is created.
- Ensure that the relevant attribute is present in the dictionary entry when adding a new parameter.

The following are the exception scenarios for this rule:

- Some objects that are created do not have a corresponding dictionary entry and only the parent object is mapped to the relevant parent dictionary entry—for example, the **Create VIF** command—in which only an array can be mapped to array dictionary entry.
- Parameter mapping is not required

For example, the ExecutionTimeout parameter in the **Create or resize aggregate** command is an unmapped parameter.

The following certified commands are examples for this category:

- Create Volume
- Create LUN

Commands that update objects

This category of commands is used to find an object and update the attributes.

You should use the following parameter mapping rules for this category of commands:

- Map the objects that are updated to the dictionary entry.
- Do not map the parameters that are updated for the object.

For example, in the **Set Volume State** command, the Volume parameter is mapped but the new State is

unmapped.

Commands that remove objects

This category of commands is used to find an object and delete it.

You should map the object that is deleted by the command to its dictionary entry. For example, in the **Remove Volume** command, the Volume to be deleted is mapped to the relevant attributes and references of the Volume dictionary entry.

Commands that deal with optional parent and child objects

You should use the following parameter mapping rules for this category of commands:

- Do not map any mandatory parameter of a command as a reference from an optional parameter of the command.

This rule is more relevant when a command deals with optional child objects of a specific parent object. In this case, the child and parent object should be mapped explicitly. For example, in the **Stop Deduplication Jobs** command, the command stops a running deduplication job on a specific volume when specified along with Array or on all volumes of the given Array. In this case, the array parameter should be mapped directly to the array dictionary entry and not to Volume.Array because Volume is an optional parameter in this command.

- If a parent and child relationship exists between dictionary entries at the logical level but not between the actual instances in a specific command, then those objects should be mapped separately.

For example, in the **Move Volume** command, Volume is moved from its current parent aggregate to a new destination aggregate. Therefore, Volume parameters are mapped to a Volume dictionary entry and the destination aggregate parameters are mapped separately to the Aggregate dictionary entry but not as volume.aggregate.name.

Commands that update associations between objects

For this category of commands, you should map both the association and the objects to relevant dictionary entries. For example, in the Add Volume to vFiler command, the Volume and vFiler parameters are mapped to the relevant attributes of the Volume and vFiler dictionary entries.

How you define constants

You can create and use constants to define a value, which can be used across a single workflow. Constants are defined at a workflow level.

The constants used in the workflow and their value are displayed in the monitoring window of the workflow during planning and execution. You must use unique names for constants.

You can use the following naming conventions to define constants:

- Uppercase for the first letter of each word, without underscores or spaces between words

All terms and abbreviations should use upper case—for example, ActualVolumeSizeInMB.

- Uppercase for all letters

You can use underscores to separate words—for example, AGGREGATE_USED_SPACE_THRESHOLD.

You can include the following as values for workflow constants:

- Numbers
- Strings
- MVEL expressions

Expressions are evaluated during the planning and execution phases of the workflows. In the expressions, you must not reference variables that are defined in a loop.

- User inputs
- Variables

How repeat row works

A workflow contains commands and command details arranged in rows. You can specify the commands in a row to be repeated for a fixed number of iterations or dynamic number of iterations based on the results of search criteria.

The command details in a row can be specified to repeat a certain number of times or when the workflow is designed. The workflow can also be designed such that the number of times the row must repeat can be specified when the workflow is executed or scheduled for an execution. You can specify search criteria for an object and the commands in a row can be set to repeat as many times as the objects are returned by the search criteria. Rows can also be set to repeat when certain conditions are met.

Row repetition variables

You can specify variables in the variable list that can be manipulated during the row iterations. For the variables, you can specify a name, a value with which the variables are initialized, and an MVFLEX Expression Language (MVEL) expression that is evaluated after every iteration of the row repetition.

The following illustration shows the repeat row options and an example of a row repetition variable:

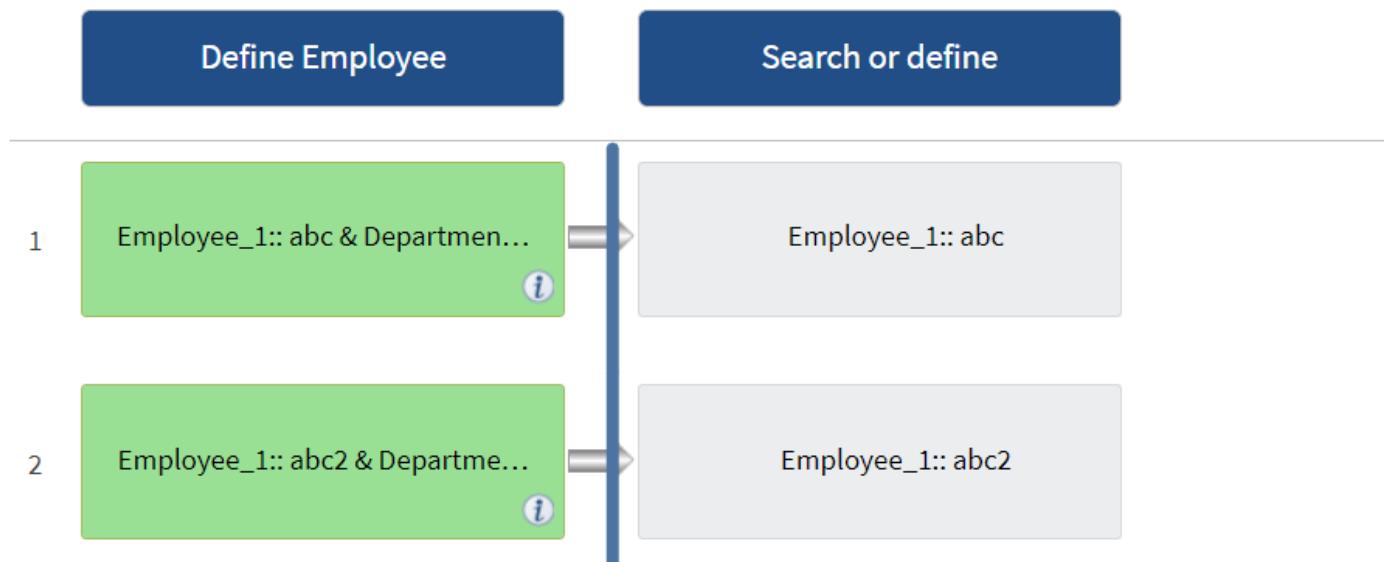
Row Repetition Details [?](#) X

Repeats*	Number of times						
Number of Times*	Number of times For every resource in a group						
Index Variable*	Index1						
Variables	<table border="1"> <thead> <tr> <th>Name</th> <th>Initial Value</th> <th>Expression</th> </tr> </thead> <tbody> <tr> <td>size_to_alloc</td> <td>SIZE_MB</td> <td>(int)size_to_allocated - getData()</td> </tr> </tbody> </table>	Name	Initial Value	Expression	size_to_alloc	SIZE_MB	(int)size_to_allocated - getData()
Name	Initial Value	Expression					
size_to_alloc	SIZE_MB	(int)size_to_allocated - getData()					
<input type="button" value="Add"/> <input type="button" value="Remove"/>							
<input type="button" value="Cancel"/> <input type="button" value="OK"/>							

Row repetition with approval points

When you have specified iterations of repeat rows for commands and included approval points, all the iterations of the commands before an approval point are executed. After you approve the approval point, the execution of all iterations of the successive commands continues until the next approval point.

The following illustration shows how the iterations of repeat rows are executed when an approval point is included in a workflow:



Repeat row examples in predefined workflows

You can open the following predefined workflows in the Designer to understand how repeat rows are used:

- Create a Clustered Data ONTAP NFS Volume
- Create VMware NFS Datastore on Clustered Data ONTAP Storage
- Establish Cluster Peering
- Remove a Clustered Data ONTAP Volume

How resource selection works

OnCommand Workflow Automation (WFA) uses search algorithms to select storage resources for workflow execution. You should understand how resource selection works in order to design workflows efficiently.

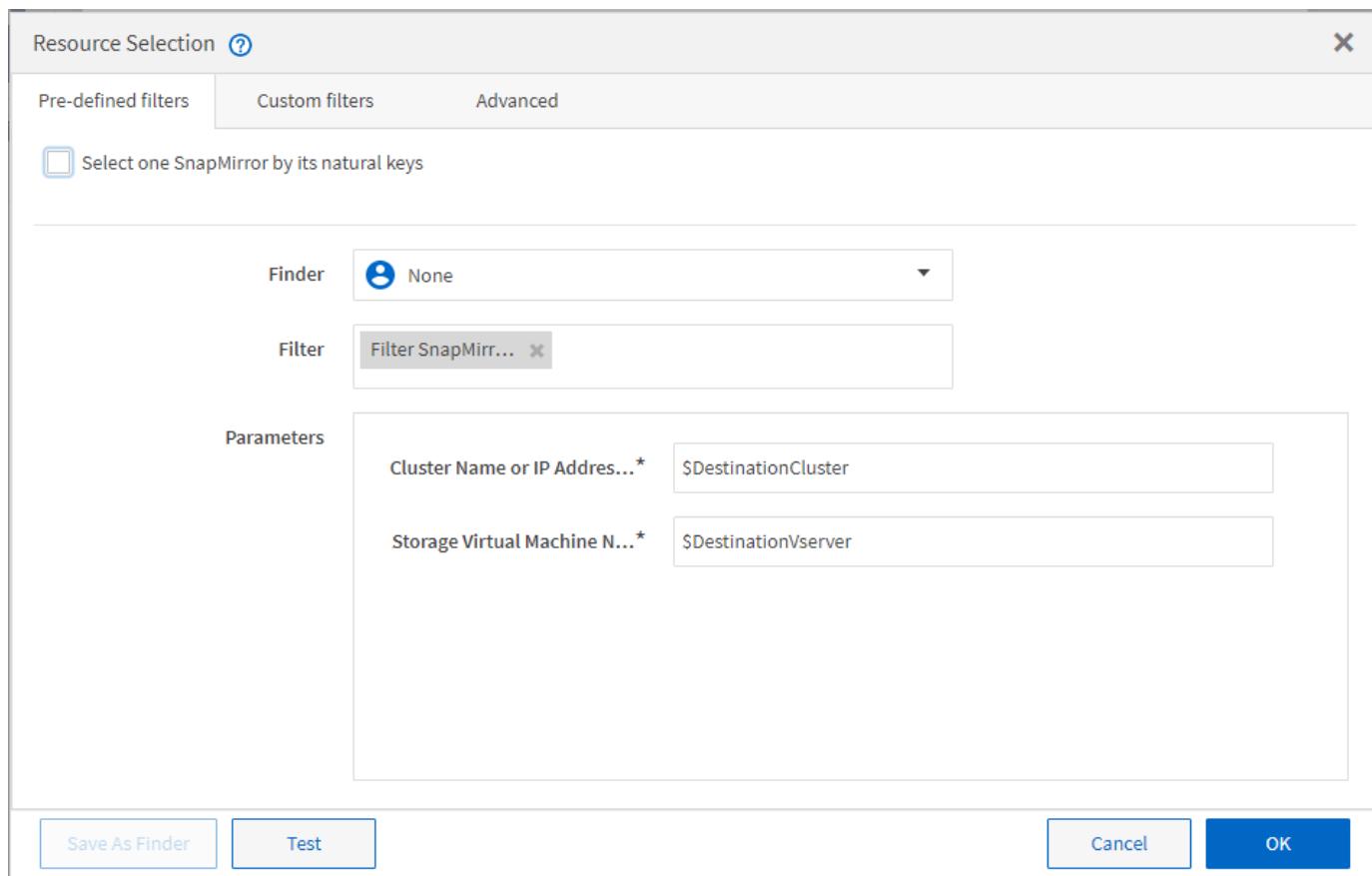
WFA selects dictionary entry resources—such as vFiler units, aggregates, and virtual machines—using search algorithms. The selected resources are then used for executing the workflow. The WFA search algorithms are part of the WFA building blocks, and include finders and filters. To locate and select the required resources, the search algorithms search through the data that is cached from different repositories, such as Active IQ Unified Manager, VMware vCenter Server, and a database. By default, a filter is available for every dictionary entry for searching a resource based on its natural keys.

You should define the resource selection criteria for each command in your workflow. In addition, you can use a finder to define the resource selection criteria in each row of your workflow. For example, when you are creating a volume that requires a specific amount of storage space, you can use the “Find aggregate by available capacity” finder in the “Create Volume” command, which selects an aggregate with a specific amount of available space and creates the volume on it.

You can define a set of filter rules for dictionary entry resources, such as vFiler units, aggregates, and virtual machines. Filter rules can contain one or more groups of rules. A rule consists of a dictionary entry attribute, an operator, and a value. The attribute can also include attributes of its references. For example, you can specify a rule for aggregates as follows: List all aggregates that have names starting with the string “aggr” and have more than 5 GB of available space. The first rule in the group is the attribute “name”, with the operator “starts-with”, and the value “aggr”. The second rule for the same group is the attribute “available_size_mb”, with the operator “>”, and the value “5000”. You can define a set of filter rules along with public filters. The Define filter rules option is disabled if you have selected a finder. The Save as Finder option is disabled if you have selected the Define filter rules check box.

In addition to the filters and finders, you can use a search or define command to search for available resources. The search or define command is the preferred option over the No-op commands. The search and define command can be used to define resources of both the certified dictionary entry type and the custom dictionary entry type. The search or define command searches for resources but does not perform any action on the resource. However, when a finder is used to search for resources, it is used in the context of a command, and the actions defined by the command are executed on the resources. The resources returned by a search or define command are used as variables for the other commands in the workflow.

The following illustration shows that a filter is used for resource selection:



Resource selection examples in predefined workflows

You can open the command details of the following predefined workflows in the Designer to understand how resource selection options are used:

- Create a Clustered Data ONTAP NFS Volume
- Establish Cluster Peering
- Remove a Clustered Data ONTAP Volume

How reservation works

OnCommand Workflow Automation resource reservation capability reserves the required resources to ensure that the resources are available for successful execution of workflows.

WFA commands can reserve the required resources and remove the reservation after the resource is available in the WFA cache database, typically after a cache acquisition. The reservation capability ensures that the reserved resources are available for the workflow until the reservation expiration period that you have configured in the WFA configuration settings.

You can use the reservation capability to exclude resources reserved by other workflows during resource selection. For example, if a workflow that has reserved 100 GB of space on an aggregate is scheduled for execution after a week, and you are executing another workflow that uses the **Create Volume** command, the workflow that is executing does not consume the space reserved by the scheduled workflow to create a new volume. In addition, the reservation capability enables workflows to be executed in parallel.

When previewing a workflow for execution, the WFA planner considers all the reserved objects, including the

existing objects in the cache database. If you have enabled reservation, the effects of the scheduled workflows and the workflows that are executing in parallel, and the existence of storage elements are considered when planning the workflow.

The arrow in the following illustration shows that reservation is enabled for the workflow:

Workflow 'Abort SnapMirror relationship'

Details	Define Workflow	User Inputs	Constants	Return Parameters	Help Content	Advanced
Workflow Name*	Abort SnapMirror relationship					
Entity Version*	1.0.0					
Categories	Data Protection					
Workflow Description	The 'Abort SnapMirror' workflow stops ongoing transfers for a					
Ready For Production	<input checked="" type="checkbox"/>					
Consider Reserved Elements	<input checked="" type="checkbox"/> 					
Enable Element Existence Validation	<input checked="" type="checkbox"/>					
Minimum Software Versions	Clustered Data ONTAP 8.2.0					

Reservation examples in predefined workflows

You can open the following predefined workflows in the Designer to understand how reservation is used:

- Clone Environment
- Create a Clustered Data ONTAP Volume
- Establish Cluster Peering
- Remove a Clustered Data ONTAP Volume

What incremental naming is

Incremental naming is an algorithm that enables you to name the attributes in a workflow based on the search results for a parameter. You can name the attributes based on an incremental value or a custom expression. The incremental naming functionality helps you implement a naming convention based on your requirement.

You can use the incremental naming functionality when designing workflows to dynamically name the objects created by the workflow. The functionality enables you to specify search criteria for an object using the resource selection feature and the value returned by the search criteria is used for the object's attribute. In addition, you can specify a value for the attribute if no object was found with the specified search criteria.

You can use one of the following options for naming the attributes:

- Providing an increment value and suffix

You can provide a value that should be used along with the value of the object found by the search criteria and increment with the number you specify. For example, if you want to create volumes with the naming convention of `filer_name_unique_number_environment`, you can use a finder to find the last volume by its name prefix and increment the unique number by 1, as well as add the suffix name to the volume name. If the last volume name prefix found was `vf_023_prod` and you are creating three volumes, the names for the volumes created are `vf_024_prod`, `vf_025_prod`, and `vf_026_prod`.

- Providing a custom expression

You can provide a value that should be used along with the value of the object found by the search criteria and add additional values based on the expression you enter. For example, if you want to create a volume with the naming convention of `last_volume.name` padded with 1, you can enter the expression `last_volume.name + ' ' + nextName("lab1")`. If the last volume name found was `vf_023`, the name for the volume created is `vf_023_lab2`.

The following illustration shows how a custom expression can be provided to specify a naming convention:

Incremental Naming Wizard for Volume : name X

*The Incremental Naming wizard allows you to define the value of **name** based on a search for an existing **Volume***

Search criteria for existing Volume **Volume Name : \$VolumeName, Cluster Name or IP Address : \$...**

Enter a value for **name** if no **Volume** matches the above search criteria

PRE_8_2_CLUSTER

if **Volume** was found using above search criteria, set value for **name** by

Custom expression

providing a custom expression

▼

Custom expression

last_volume.name

Cancel
Save

What conditional execution is

Conditional execution helps you to design workflows that can execute commands when specified conditions are met.

Execution of commands in a workflow can be dynamic. You can specify a condition for the execution of each command or a row of commands in your workflow. For example, you might want the “Add volume to dataset” command to be executed only when a specific dataset is found and you do not want the workflow to fail if the dataset is not found. In this case, you can enable the “Add volume to dataset” command to search for a specific dataset and if it is not found, you can disable the command in the workflow.

Options for conditional execution of commands are available in the `Dictionary` object tab and the Advanced tab of the `Parameters for commands` dialog box.

You can abort a workflow or disable a specific command in the workflow. In addition, you can set a command to be executed using one of the following options:

- Without any condition
- When the variables you have specified are found
- When the variables you have specified are not found
- When the expression you have specified is true

You can also set a command to wait for a specific time interval.

Conditional execution examples in predefined workflows

You can open the command details of the following predefined workflows in the Designer to understand how conditional execution of commands are used:

- Create a basic Clustered Data ONTAP Volume
- Create a Clustered Data ONTAP NFS Volume

How return parameters work

Return parameters are parameters that are available after the planning phase of a workflow. The values returned by these parameters are useful in debugging a workflow. You should understand how return parameters work and what parameters can be used as return parameters to debug workflows.

You can designate a set of parameters, such as variable attributes, expressions, and user input values, in a workflow as return parameters. During workflow execution, the values of the designated parameters are populated in the planning phase and execution of the workflow starts. The values of these parameters are then returned the way they were calculated in that specific execution of the workflow. If you want to debug the workflow, you can refer to the values that were returned by the parameters.

You can specify the required return parameters in a workflow when you want to see what are the calculated or selected values for those parameters. For example, when using resource selection logic to select an aggregate in a workflow, you can specify aggregate as the return parameter so that you can see which aggregate was selected during the planning of the workflow.

Before referring to the values of the return parameters for debugging your workflow, you should confirm that the execution of the workflow is complete. The return parameter values are set for each workflow execution. If

you have added a return parameter after several executions of a workflow, the value of that parameter is available only for executions after the addition of the parameter.

Parameters that can be used as return parameters

Return parameters	Example
Variable attributes that are scalar	volume1.name, which is an attribute of the "volume name" variable
Constants	MAX_VOLUME_SIZE
User inputs	\$clusterName
MVEL expressions that involve variable attributes, constants, and user inputs	volume1.name+'-'+\$clusterName
The return parameter that a command adds during execution	The \$volumeUUID parameter is added as a return parameter when you use the following line in a PowerShell command: Add-WfaWorkflowParameter -Name "VolumeUUID" -Value "12345" -AddAsReturnParameter \$true.

Examples of return parameters in predefined workflows

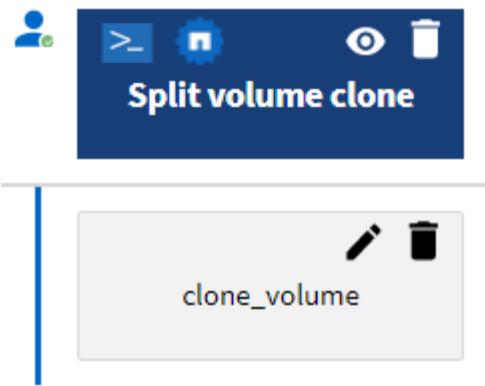
If you want to understand how return parameters are specified, you can open the following predefined workflows in the Designer and review the specified return parameters:

- Create an NFS Volume in a vFiler
- Create a Qtree CIFS Share in a vFiler
- Create a Clustered Data ONTAP Volume CIFS Share

What approval points are

Approval points are check points used in a workflow to pause the workflow execution and resume it based on a user approval.

The blue vertical bar shown in the following illustration is an approval point:



You can use approval points for incremental execution of a workflow, where sections of the workflow should be executed only after a certain condition is met. For example, when the next section has to be approved or when successful execution of the first section is validated. Approval points do not handle any process between pausing and resuming of a workflow. Email and SNMP notifications are sent, as specified in the WFA configuration, and the storage operator can be asked to perform certain actions upon receiving the workflow pause notification. For example, the storage operator can send planning details to admin, approver, or operator for approval and resume the workflow when the approval is received.

Approvals might not be required at all times. In some scenarios, the approval might be required only if a particular condition is met and the conditions can be configured when an approval point is added. For example, consider a workflow that is designed to increase the size of a volume. You can add an approval point at the beginning of the workflow for the storage operator to obtain approval from the managers when the increase in the volume size results in an 85% usage of the space in the aggregate that contains the volume. During the workflow execution and on selecting a volume that results in this condition, the execution is stopped until it is approved.

The condition that is set up for the approval point can have one of the following options:

- Without any condition
- When the variable you have specified is found
- When the variable you have specified is not found
- When the expression you have specified evaluates to true

There is no limitation on the number of approval points in a workflow. You can insert approval points before commands in a workflow and set the commands after the approval point to wait for approval before execution. Approval points provide information, such as time of change, user, and comments, allowing you to see when and why the workflow execution was paused or resumed. The approval point comments can include MVEL expressions.

Approval point examples in predefined workflows

You can open the following predefined workflows in the Designer to understand how approval points are used:

- Remove a Clustered Data ONTAP Volume
- Controller and shelf upgrade of an HA pair
- Migrate Volumes

How you execute custom REST end points

OnCommand Workflow Automation (WFA) provides a mechanism to configure the custom REST end points to execute the workflows. Custom REST end points help an architect to configure easy-to-understand, intuitive, and uniform resource identifiers (URIs) to execute workflows, which follow the REST conventions of POST, PUT, or DELETE based on the workflow semantics. These URIs ease the client code development for client developers.

WFA enables you to configure a custom URI path for workflow execution through the API calls. Each segment in the URI path can be a string or a valid name of the user input of the workflow in brackets, for example, `/devops/{ProjectName}/clone`. The workflow can be invoked as a call to `https://WFAServer:HTTPS_PORT/rest/devops/Project1/clone/jobs`.

Validation for the URI path is as follows:

- The REST path must start with “/”.
- The characters allowed are alphabets, digits, and underscore.
- The user input name must be surrounded by “{}”.



You must check that the value surrounded by “{}” is a valid user input name.

- There should be no empty path segments, for example, //, /{}, and so on.
- The HTTP method configuration and custom URI path configuration should either both be configured or neither configured.

How continue on failure works

The continue on failure feature helps you to configure a step in a workflow so that the workflow execution can continue even if the step fails. You can address the failed steps and resolve the issue that caused the step to fail by accessing the `wfa.log` file or by clicking the icon.

A workflow that has one or more such failed steps is in the Partially Successful state after the execution is complete. You can configure a step so that the workflow execution continues even if the step fails by selecting the required option in the Advanced tab of the Parameters for <command_name> dialog box.

If a step is not configured to continue on failure, the workflow execution is aborted if the step fails.

If a step that is configured to continue on failure fails, you can set the workflow to be executed by using one of the following options:

- Abort workflow execution (default option)
- Continue execution from the next step
- Continue execution from the next row

Sample workflow requirements checklist

A workflow requirements checklist includes detailed requirements—such as commands,

user input, and resources—for a planned workflow. You can use the checklist to plan your workflows and identify the gaps in the requirements.

Requirements checklist example

The following sample workflow requirements checklist lists the requirements for the “Create a Clustered Data ONTAP Volume” workflow. You can use this sample checklist as a template to list your workflow requirements.

Requirement	Description
Workflow name	Create a Clustered Data ONTAP Volume
Category	Storage provisioning
Description	The workflow creates a new volume in a specific SVM. This workflow is meant for a scenario where a volume is provisioned and delegated for later usage.
High-level description of how the workflow works	<ul style="list-style-type: none">• The SVM that contains the volume is specified by the user (cluster, SVM names).• A volume is created based on the specified size.• The configuration of the volume is described in a template.

Requirement	Description
Details	<ul style="list-style-type: none"> • Use the Create CM Volume command • Command details for Create CM Volume: <ul style="list-style-type: none"> ◦ Execution is set as always ◦ Volume details are specified by filling in attributes for the volume ◦ Use the Space Guaranteed Settings template for configuring the volume ◦ Volume name and size are provided by user. • The volume will be mounted in the SVM namespace as <code>/volname</code> (under the root namespace). • Use the actualVolumeSize function because the snap reserve will be 5%. • SVM reference is defined with the following resource selection logic: <ul style="list-style-type: none"> ▪ CM SVM by key — searches for SVM by name and the cluster, which is provided by the user ▪ CM SVM by type — only data SVMs (type = cluster) ▪ SVM by state — (state = running) • Aggregate reference is defined with the resource selection logic as a predefined finder (CM Aggregate by space thresholds and RAID Type): <ul style="list-style-type: none"> ▪ CM Aggregate by available capacity (capacity = size of volume to be provisioned, cluster given by user) ▪ CM Aggregate by delegation to SVM ▪ CM Aggregate by RAID Type (RAID-DP) ▪ CM Aggregate not aggr0 ▪ CM Aggregate by used size % (threshold = 90, spaceToBeProvisioned = size provided, since guarantee is volume) ▪ CM Aggregate by over commitment (threshold = 300, spaceToBeAllocated = Size of volume being provisioned) ▪ Select the aggregate with maximum free space

Name	Type	Description (data values, validation, and so on)
Cluster	Locked query (tabular)	<ul style="list-style-type: none"> • Cluster hosting the SVM • Query can be tabular display with primary address and name of the cluster • Sort alphabetically by name
SVM	Locked query	<ul style="list-style-type: none"> • SVM in which the volume is provisioned • Query should only display SVM names belonging to the cluster chosen in the previous input <p>Show only cluster type SVMs, not admin or node (type column of cm_storage.vserver)</p> <ul style="list-style-type: none"> • Sort alphabetically
Volume	String	<ul style="list-style-type: none"> • Name of the volume to be created
Size in GB	Integer	<ul style="list-style-type: none"> • Size of the volume to be provisioned • Data size (snap reserve should be considered)

Commands

Name	Description	Status
Create CM Volume	Creates a volume in the SVM	Existing

Return Parameters

Name	Value
Volume name	Name of the provisioned volume
Aggregate name	Name of the selected aggregate
Node name	Name of the node
Cluster name	Name of the cluster

Gaps and issues

1.	
2.	
3.	
4.	
5.	

Create a workflow

You can use Workflow Automation (WFA) to create workflows for tasks such as provisioning, migrating, and decommissioning storage for databases or file systems. You should create workflows when the predefined WFA workflows do not match your requirements.

What you'll need

- You must have understood the concepts for WFA building blocks.
- You must have understood capabilities such as repeat row, approval points, and resource selection that are required for your workflow.
- You must have completed the planning required for your workflow, including the workflow requirement checklist.
- You should have created the help content, which provides information about the workflow to storage operators.

About this task

The construction of each workflow might vary based on the goal and requirement of the workflow. This task does not provide instructions for a specific workflow, but provides general instructions for creating a workflow.

Steps

1. Click **Workflow Design > Workflows**.
2. Click  on the toolbar.
3. In the **Workflow** tab, perform the following steps:
 - a. Expand the required schema, and then double-click the required  (command) or  (workflow) from the **Available Steps** list.
 - b. **Optional:** Click  to add the required number of rows, which are used to specify details for execution of steps.

You can repeat this step as required. You can drag-and-drop steps to rearrange the steps in the workflow editor.

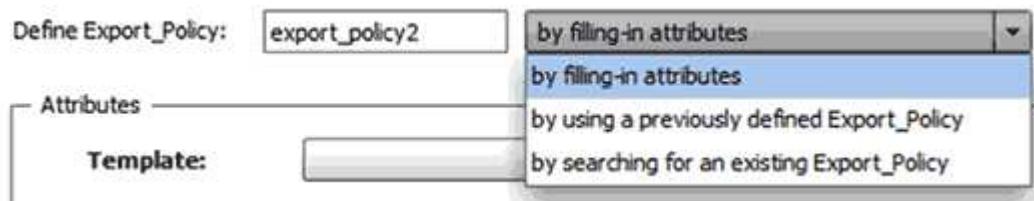
 - c. Position your cursor below the step you have added and click  to add step details for the step

Each step is executed based on the specified step details at the specified row and column. The steps are executed from left to right and in the top to bottom order.

execution, at the required row.

For this step...	Do this...
Workflow	Enter the required user inputs in the Workflow tab and the required condition in the Advanced tab.
Command	In the Parameters for <command> tab, click each object tab, select the required option to define the object attributes, and then enter the required details in the Advanced tab and the Other Parameter tab.
Search or define	Select the dictionary entry object that should be searched for or defined.

The following illustration shows the available options for defining the object attributes:



Choose the appropriate action:

For...	Do this...
by filling-in attributes	<p>Enter the value for attributes using the following options:</p> <ul style="list-style-type: none">• Expressions• Variables• User inputs• Resource selection• Incremental naming <p>You must position your cursor over the attribute fields and click to use the resource selection or incremental naming capabilities.</p>
by using a previously defined <i>object</i>	Select the previously defined object in the box before the option list.

For...	Do this...
by searching for an existing <i>object</i>	<ul style="list-style-type: none"> <li data-bbox="887 164 1486 234">i. Click Enter search criteria to search for the object using the resource selection capability. <li data-bbox="887 255 1486 325">ii. Select one of the required options for execution if the required object is not found: <ul style="list-style-type: none"> <li data-bbox="936 346 1155 375">◦ Abort workflow <p data-bbox="959 403 1486 473">This option aborts the workflow execution if the specific object is not found.</p> <ul style="list-style-type: none"> <li data-bbox="936 502 1246 532">◦ Disable this command <div data-bbox="985 593 1442 709" style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 10px auto;"> <p data-bbox="985 593 1442 709">This option disables only the current step and executes the workflow.</p> </div> <ul style="list-style-type: none"> <li data-bbox="936 783 1475 853">◦ Fill-in attributes for <i>object</i> and execute the command <div data-bbox="985 914 1462 1064" style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 10px auto;"> <p data-bbox="985 914 1462 1064">This option enables you to enter the required attributes and execute the workflow.</p> </div>

4. If you want to insert an approval point, click  and enter the required comment for the approval point.

Approval point comments can include MVEL expressions.

5. Click  that is next to the row numbers to perform the following:

- Insert a row.
- Copy the row.
- Repeat the row.

You can use one of the following options to specify repetition of the command parameters:

- Number of times

You can use this option to repeat the command execution for the number of repetitions you specify. For example, you can specify that the “Create qtree” command should be repeated three times to create three qtrees.

You can also use this option for a dynamic number of command executions. For example, you can create a user input variable for the number of LUNs to be created and use the number specified by the storage operator when the workflow is executed or scheduled.

- For every resource in a group

You can use this option and then specify a search criteria for an object. The command is repeated as many times as the object is returned by the search criteria. For example, you can search for the nodes in a cluster and repeat the “Create iSCSI Logical Interface” command for each node.

- Add a condition for execution of the row.
- Remove the row.

6. In the **Details** tab, perform the following steps:

- Specify the required information in the **Workflow name** and **Workflow Description** fields.

The workflow name and description must be unique for each workflow.

- Optional:** Specify the entity version.
- Optional:** Clear the **Consider Reserved Elements** check box if you do not want to use the reservation capability.
- Optional:** Clear the **Enable element existence validation** check box if you do not want to enable validation for elements that exist with the same name.

7. If you want to edit the user inputs, perform the following steps:

- Click the **User Inputs** tab.
- Double-click the user input you want to edit.
- In the **Edit Variable: <user input>** dialog box, edit the user input.

8. If you want to add constants, perform the following steps

- Click the **Constants** tab, and then add the required constants for your workflow by using the **Add** button.

You can define constants when you are using a common value for defining the parameters for multiple commands. For example, see the AGGREGATE_OVERCOMMITMENT_THRESHOLD constant used in the “Create, map and protect LUNs with SnapVault” workflow.

- Enter the name, description, and value for each constant.

9. Click the **Return Parameters** tab, and then add the required parameters for your workflow by using the **Add** button.

You can use return parameters when the workflow planning and execution must return some calculated or selected values during planning. You can view the calculated or selected values in the Return Parameters tab of the monitoring window in the workflow preview or after the workflow execution is complete.

Aggregate: You can specify aggregate as a return parameter to see which aggregate was selected using the resource selection logic.

If you have included a child workflow in your workflow and if the child workflow return parameter names contain a space, dollar sign (\$), or a function, you should specify the return parameter name within square brackets in the parent workflow to view the child workflow return parameter value in your parent workflow.

If the parameter name is...	Specify as...
ChildWorkflow1.abc\$value	ChildWorkflow1["abc\$"+ "value"]
ChildWorkflow1.\$value	ChildWorkflow1["\$"+ "value"]

If the parameter name is...	Specify as...
ChildWorkflow1.value\$	ChildWorkflow1.value\$
ChildWorkflow1.P_N	ChildWorkflow1["P_N"]
ChildWorkflow1.return_string("HW")	ChildWorkflow1["return_string(\"HW\")"]

10. **Optional:** Click the **Help Content** tab to add the help content file you have created for the workflow.
11. Click **Preview** and ensure that the planning of the workflow is completed successfully.
12. Click **OK** to close the preview window.
13. Click **Save**.

After you finish

Test the workflow in your test environment, and then mark the workflow as ready for production in **WorkflowName > Details**.

Create workflow help content

OnCommand Workflow Automation (WFA) admins and architects who design workflows can create help content for the workflows and include it in the workflow.

What you'll need

You must be aware of how to create web pages using HTML.

About this task

The help should provide information about the workflow and the user inputs for the workflow to the storage operator who executes the workflow.

Steps

1. Create a folder with the following name: workflow-help.
2. Author the help content using an HTML editor or a text editor and save it as an `index.htm` file in the `workflow-help` folder.

You must not include JavaScript files as part of the help content. The following are the supported file extensions:

- .jpg
- .jpeg
- .gif
- .png
- .xml
- .thmx
- .htm

- .html
- .css

You can also include the `Thumbs.db` file, which is created by Windows.

3. Verify that the `index.htm` file and other files associated with the help content, such as images, are available in the `workflow-help` folder.
4. Create a `.zip` file of the folder and ensure that the size of the `.zip` file is not more than 2 MB.

Create an NFS `volume-help.zip`

5. Edit the workflow for which you have created the help content, and then click **Setup > Help Content > Browse** to upload the `.zip` file.

Create WFA workflow packs

You can create workflow packs on OnCommand Workflow Automation (WFA) for your storage automation and integration requirements.

Steps

1. Log in to the WFA window through a web browser.
2. Click **Content Management > Packs**.
3. Click the **New Pack** icon.
4. In the **New Pack** dialog box, enter values for the **Name**, **Author**, **Version**, and **Description** fields.
5. Click **Save**.
6. Verify that the new pack is created in the **Packs** window.

Add entities to WFA workflow packs

You can add one or more entities to a workflow pack in OnCommand Workflow Automation (WFA) for your storage automation and integration requirements.

About this task

You can remove a pack from the following entities:

- Workflow
- Finders
- Filters
- Commands
- Functions
- Templates
- Schemes
- Dictionary
- Data Sources Types
- Remote System Types

- Cache Queries
- Categories

Steps

1. Log in to the **WFA** window through a web browser.
2. In the **Workflow design portal**, navigate to the entity you want to add and click **<Entity>**.
3. In the **Entity** window, select the entity you want to add to the pack.
4. Click the **Add To Pack** icon.

“Add To Pack” is enabled only for entities for which the certification is set to **None**.

5. In the **Add To Pack <Entity>** dialog box, from the **Available Packs** drop-down list, select the pack to which you want to add the entity.
6. Click **OK**.

Delete OnCommand Workflow Automation packs

You can delete a pack from OnCommand Workflow Automation (WFA) if you no longer require it. If you delete a pack, all the entities associated with the pack are deleted.

About this task

- You cannot delete a pack if there are any dependencies on the entities that are part of the pack.

For example, if you attempt to delete a pack that includes a command that is part of a custom workflow, the delete operation will fail because the custom workflow is dependent on the pack. You can delete the pack only after you delete the custom workflow.

- Entities that are part of a pack cannot be deleted individually.

To delete an entity that is part of a pack, you must delete the pack containing that entity. If an entity is part of multiple packs, the entity will be deleted from the WFA server only when all the packs containing that entity are deleted.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Content Management > Packs**.
3. Select the pack you want to delete and click .
4. In the **Delete Pack** confirmation dialog box, click **OK**.

Export OnCommand Workflow Automation content

You can save user-created OnCommand Workflow Automation (WFA) content as a **.dar** file and share the content with other users. The WFA content can include the entire user-created content or specific items such as workflows, finders, commands, and dictionary terms.

What you'll need

- You must have access to the WFA content that you want to export.

- If content that is to be exported contains references to certified content, the corresponding certified content packs must be available on the system when the content is imported.

These packs can be downloaded from the Storage Automation Store.

About this task

- You cannot export the following types of certified content:
 -  - NetApp-certified content
 -  - content developed by Professional Services (PS), which is available only on custom installations made by PS
 -  - packs developed by users
- All of the objects that are dependent on the exported object are also exported.

For example, exporting a workflow also exports the dependent commands, filters, and finders for the workflow.

- You can export locked objects.

The objects remain in the locked state when they are imported by other users.

Steps

1. Log in to WFA through a web browser.
2. Export the necessary content:

If you want to...	Do this...
Export all user-created content as a single .dar file	<ol style="list-style-type: none"> a. Click Settings, and under Maintenance click Export All Workflows. b. Specify a file name for the .dar file, and then click Export.
Export specific content	<ol style="list-style-type: none"> a. Navigate to the window from which you want to export content. b. Select one or more items from the window, and then click . c. In the Export As dialog box, specify a file name for the .dar file, and then click Export.

3. In the **Save As** dialog box, specify the location where you want to save the .dar file, and then click **Save**.

Import OnCommand Workflow Automation content

You can import user-created OnCommand Workflow Automation (WFA) content such as workflows, finders, and commands. You can also import content that is exported from another WFA installation, content that is downloaded from the Storage Automation Store or the WFA community, as well as packs, including Data ONTAP PowerShell toolkits and Perl NMSDK toolkits.

What you'll need

- You must have access to the WFA content that you want to import.
- The content that you want to import must have been created on a system that is running the same version or an earlier version of WFA.

For example, if you are running WFA 2.2, you cannot import content that was created using WFA 3.0.

- You can import content developed on N-2 versions of WFA only into WFA 5.1.
- If the .dar file references NetApp-certified content, the NetApp-certified content packs must be imported.

The NetApp-certified content packs can be downloaded from the Storage Automation Store. You must refer to the documentation of the pack to verify that all requirements are met.

Steps

1. Log in to WFA through a web browser.
2. Click **Settings**, and under **Maintenance** click **Import Workflows**.
3. Click **Choose File** to select the .dar file that you want to import, and then click **Import**.
4. In the **Import Success** dialog box, click **OK**.

Import WFA workflow packs

You can import workflow packs from the server to OnCommand Workflow Automation (WFA) for your storage automation and integration requirements.

What you'll need

You must have access to the WFA content in the server that you want to import.

Steps

1. Log in to the **WFA** window through a web browser.
2. Click **Content Management > Packs**.
3. Click the **Import From Server** icon.
4. In the **Import From Server Folder** dialog box, in the **Folder location at server system** field, enter the location of the pack in the server in a string format, for example, C:\work\packs\test.
5. Click **OK**.
6. Verify that the pack is imported in the **Packs** window.

Considerations while importing OnCommand Workflow Automation content

You must be aware of certain considerations when you import user-created content, content that is exported from another OnCommand Workflow Automation (WFA) installation, or content that is downloaded from the Storage Automation Store or the WFA community.

- WFA content is saved as a .dar file and can include the entire user-created content from another system or specific items such as workflows, finders, commands, and dictionary terms.
- When an existing category is imported from a .dar file, the imported content is merged with the existing

content in the category.

For example, consider there are two workflows WF1 and WF2 in category A in the WFA server. If workflows WF3 and WF4 in category A are imported to the WFA server, category A will contain workflows WF1, WF2, WF3, and WF4 after the import.

- If the .dar file contains dictionary entries, then the cache tables corresponding to the dictionary entries are automatically updated.

If the cache tables are not updated automatically, an error message is logged in the wfa.log file.

- When importing a .dar file that has a dependency on a pack that is not present in the WFA server, WFA tries to identify whether all the dependencies on the entities are met.
 - If one or more entities are missing or if a lower version of an entity is found, the import fails and an error message is displayed.

The error message provides details of the packs that should be installed in order to meet the dependencies.

- If a higher version of an entity is found or if the certification has changed, a generic dialog box about the version mismatch is displayed, and the import is completed.

The version mismatch details are logged in a wfa.log file.

- Questions and support requests for the following must be directed to the WFA community:
 - Any content downloaded from the WFA community
 - Custom WFA content that you have created
 - WFA content that you have modified

Pack identification during upgrade

During the upgrade process, OnCommand Workflow Automation (WFA) identifies and classifies the entities into a pack. If you had deleted any entity of a pack before the upgrade, the pack will not be identified during the upgrade.

During the upgrade process, WFA compares the packs in the database with the list of packs that were released in the Storage Automation Store to identify the packs that were installed before the upgrade. Pack identification thus classifies existing packs in the database.

WFA performs the following processes to identify and classify packs:

- Maintains a list of packs released in the Storage Automation Store to compare and identify the packs that were installed before the upgrade.
- Classifies the entities in a pack as part of the Storage Automation Store synchronization, if Storage Automation Store is enabled.
- Classifies the entities into packs using the updated list.

Pack identification is applicable only to NetApp-certified packs that were downloaded from the Storage Automation Store.

If a pack is not identified during upgrade, you can re-import the pack to get it identified in WFA. The wfa.log

files provide details about the entities that were not identified as a pack during the upgrade.

Integrating WFA workflow packs with the SCM repository

You can integrate the OnCommand Workflow Automation (WFA) pack with the Source Control Management (SCM) repository.

You must have admin or architect credentials.

SCM tools such as GitHub, Perforce, and SVN require you to map a local directory to check out the code from the SCM repository server. This local directory mapping is called the *SCM client location*. You must set up the SCM client with a file system location as a client area.

You can set up the SCM client on a WFA server system. You must have access to the WFA server system for SCM operations

Check in a new workflow pack to SCM

You can create a new workflow pack with OnCommand Workflow Automation (WFA) and check it in to Source Control Management (SCM).

What you'll need

SCM must be set up and you must have admin or architect credentials.

Steps

1. Log in to the **WFA** window through a web browser.
2. Create a new workflow pack.

[Create a workflow automation pack](#)

3. Add entities to the pack you created.

[Add entities to an OnCommand Workflow Automation pack](#)

4. Click the **Export To Server** icon.
5. In the Export To Server Folder dialog box, in the **Folder location at server system** field, enter the file system location where the pack is to be saved in the server containing the SCM client.

To edit or re-export the pack or the contents, click the **Unlock** icon.

6. In the SCM client location, check in the pack content to the SCM server.

Check in a new version of a WFA workflow pack

You can update the version of a pack in OnCommand Workflow Automation (WFA), and then check in the updated pack to a new location in the Source Control Management (SCM) server.

What you'll need

The SCM must be set up and you must have admin or architect credentials.

Steps

1. Log in to the **WFA** window through a web browser.
2. Click **Content Management > Packs**.
3. Click the **Edit Pack** icon.
4. In the **Pack <pack name>** dialog box, in the **Version** field, update the version of the pack.
5. Click **Save**.
6. Click the **Export To Server** icon at the pack level.
7. In the **Export To Server Folder** dialog box, in the **Folder location at server system** field, enter a new file system location.

If the pack was previously saved in the `C:\p4\cdot\1.0.0` file system location, now save it in the `C:\p4\cdot\2.0.0` location.

8. In the SCM client location, check in the pack content to a new location in the SCM server.

If the pack was previously saved in the `//depot/wfa/packs/cdot/1.0.0` path in the SCM server, you can save it in another location, such as `//depot/wfa/packs/cdot/2.0.0`.

Update WFA workflow packs from the SCM server

You can update a pack in the Source Control Management (SCM) server, and then import the updated pack to OnCommand Workflow Automation (WFA).

What you'll need

The SCM must be set up and you must have admin or architect credentials

About this task

If you make any changes or updates to a pack in the SCM server, the admin or architect needs to resolve the conflicts, if any, using the SCM-provided diff tools. WFA tailors the XML diff file to show only the relevant changes.

Before importing the pack, you are notified about the changes going in to the WFA pack content.

Steps

1. Log in to the WFA window through a web browser.
2. Import the updated pack to WFA.

[Import WFA workflow packs](#)



If the WFA database already contains the same pack, then the pack content will be overwritten.

Check in existing WFA workflow packs to the SCM server

You can check in already existing packs to the Source Control Management (SCM) server from OnCommand Workflow Automation (WFA).

What you'll need

The SCM must be set up and you must have admin or architect credentials.

Steps

1. Log in to the **WFA** window through a web browser.
2. Click **Content Management > Packs**.
3. Click the **Export To Server** icon.
4. In the **Export To Server Folder** dialog box, in the **Folder location at server system** field, enter the server folder location where the pack is saved in the server.

This exports the pack in an exploded form in the file system where the SCM client is created.

5. In the SCM client location, check in the pack content to the SCM server.
6. Use the SCM-provided diff tools to verify the changes against the SCM version of the pack.

Remove WFA workflow packs from entities

You can remove a pack from the entities in OnCommand Workflow Automation (WFA) and check in the updated pack to the Source Control Management (SCM) server.

What you'll need

The SCM must be set up and you must have admin or architect credentials.

About this task

You can remove a pack from the following entities:

- Workflow
- Finders
- Filters
- Commands
- Functions
- Templates
- Schemes
- Dictionary
- Data Sources Types
- Remote System Types
- Cache Queries
- Categories

Steps

1. Log in to the **WFA** window through a web browser.
2. In the **Workflow design portal**, navigate to the entity you want to remove and click **<Entity>**.
3. Click the **Remove From Pack** icon.
4. In the **Remove From Pack <Entity>** dialog box, select the pack you want to delete from that entity.
5. Click **OK**.
6. Click the **Packs** tab.

7. Click the **Export To Server** icon.
8. In the **Export To Server Folder** dialog box, in the **Folder location at server system** field, enter the server folder location where the pack is saved in the server.

This exports the pack in exploded form in the file system where the SCM client is created.
9. In the SCM client location, check in the pack content to the SCM server.
10. Use the SCM-provided diff tools to verify the changes against the SCM version of the pack.

Roll back a WFA workflow pack to its previous version in SCM

You can roll back a pack to the previous version in Source Control Management (SCM) and import it to OnCommand Workflow Automation (WFA).

What you'll need

The SCM must be set up and you must have admin or architect credentials.

Steps

1. In the SCM client location, roll back the pack to a previous version in the file system location using SCM tools.

The SCM client gets synced to the exact change number that you are interested in.
2. Log in to the WFA window through a web browser.
3. Import the updated pack to WFA.

[Import WFA workflow packs](#)



This rolls back the WFA database to the previous version.

Creating building blocks for workflows

Workflow Automation (WFA) includes several building blocks, which are used to construct workflows. You can create the WFA building blocks that are required for your workflows.

Create a data source type

You must create a data source type to enable data acquisition from a data source, which is not predefined in OnCommand Workflow Automation (WFA).

What you'll need

- You must have created the required dictionary entry and scheme if you are creating a custom data source type that is not predefined in WFA.
- You must be aware of PowerShell scripting to create a data source type that uses the script method.

Steps

1. Click **Data Source Design > Data Source Types**.
2. Click on the toolbar.

3. In the **New Data Source Type** dialog box, enter or select the required details in the **Data source**, **Data source version**, and **Scheme** fields.
4. In the **Default port** field, enter the port number.

2638

The port number you have entered is populated when you add this data source type for data acquisition. By default, the port is used by WFA to communicate with the data source and the port should be open on the data source server.

5. From the **Method** list, select the method that WFA must use to acquire data:

If you have selected...	Then...
SQL	<p>From the Driver type list, select one of the following drivers that is appropriate for the data source:</p> <ul style="list-style-type: none"> • MySQL Connector/J • MySQL Server JDBC Driver 3.0 • Oracle JDBC Driver 11.2.0.3
SCRIPT	<p>In the Script field, enter the PowerShell script that is used to connect and retrieve data from the data source.</p> <p> The data in the dictionary entry equivalent CSV files should include tabs as field separator. For example, see the PowerShell script for the VMware vCenter data source type.</p>

6. Click **Save**.

Create a command

You can create a WFA command to complete a specific task in your workflow if there is no predefined WFA command that is suitable for the task.

What you'll need

You must know how to write the required code for the command using either PowerShell or Perl.

Steps

1. Click **Workflow Design > Commands**.
2. Click  on the toolbar.
3. In the **Properties** tab of the **New Command Definition** dialog box, enter or select the required details in the **Name**, **Description**, and **Timeout** fields.
 - a. In the **String Representation** field, enter a string representation for the command using MVEL syntax.

```
VolumeName + "=>" + SnapshotName
```

The string representation for a command is used to display the information you want to see in the workflow design during planning and execution. You must use only the parameters of the command in string representation for a command.

- b. **Optional:** If you are creating a wait command, select **Wait for condition** in the **Command type** section, and set the required value in the **Waiting interval (s)** field.
4. In the **Code** tab, perform the following steps:

- a. Select the required scripting language for the command from the **Script Language** list.

You can click + and select an additional language for the command.

- b. Enter the appropriate code for the command in the selected language tab.

If you want to use password type for user inputs in the PowerShell script, you must create an alias for the parameter and include `_Password` in the attribute. For Perl script, you can specify the type as **Password** in the **Parameters Definition** tab.



The following special characters are not supported in the command editor: "<", "&", and XML tags "< >".

```
param (
    [parameter(Mandatory=$false, HelpMessage="Specify an AD
administrator password.")]
    [Alias("ADAdminPassword_Password")] [string]$ADAdminPassword
)
```

5. In the **Parameters Definition** tab, perform the following steps:

- a. Click **Discover Parameters** to populate the parameters definition table.

The parameters and their attributes are extracted from the code and displayed in the table. For example, the `Array` and `VolumeName` parameters are extracted from the following code:

```
param (
    [parameter(Mandatory=$true, HelpMessage="Array name or IP
address")]
    [string]$Array,
    [parameter(Mandatory=$true, HelpMessage="Volume name")]
    [string]$VolumeName,
)
```

- b. Click the description column of the parameters to edit the description.

You cannot edit any other field in this tab.

6. In the **Parameters Mapping** tab, perform the following steps for each parameter:

- a. From the **Type** column, select the appropriate dictionary object.
- b. In the **Attribute** column, enter or select the appropriate attributes for the dictionary object from the list.

After entering an attribute, you can enter a period (.) and include another attribute of that object.

Enter `cm_storage.volume` as type and `aggregate.name` as the attribute for the `AggregateName` parameter.

- c. In the **Object Name** column, enter an object name.

The object name is used for grouping the parameters under a tab in the Parameters for <command> dialog box when you are specifying the command details in a workflow.

The unmapped parameters are displayed in the **Other parameters** tab of the Parameters for <command> dialog box when you are specifying the command details in a workflow.

7. In the **Reservation** tab, enter a reservation script using SQL queries to reserve the resources that are required by the command during a scheduled workflow execution:

- a. **Optional:** In the **Reservation Representation** field, enter a string representation for the reservation using MVEL syntax.

```
``Add rule for SnapMirror label ``  
" + SnapMirrorLabel + "[.code] to the SnapMirror policy`  
" + PolicyName + "``
```

The string representation is used to display the details of the resources reserved in the Reservations window.



The reservation script must not perform any operation on databases except the `cm_storage`, `cm_performance`, `storage`, `performance`, `vc`, and `custom` schemes.

8. **Optional:** In the **Verification** tab, enter an SQL query to verify whether the command has affected the data sources and the WFA cache as expected so that the reservation can be removed.

The SQL query that you enter can only consist of SQL SELECT statements.

- a. Click **Test Verification** to test the verification script.
 - b. In the **Verification** dialog box, enter the required test parameters.
 - c. If you do not want to use the reservation data to test the verification script, clear the **Use *reservation data in test** field.
 - d. Click **Test**.
 - e. After reviewing the test result, close the dialog box.

9. Click **Test** to test the command.

10. In the **Testing Command <command name>** dialog box, click **Test**.

The result of the test is displayed in the Log messages section of the dialog box.

11. Click **Save**.

Test the reservation script for commands

You can test the reservation scripts you have written for OnCommand Workflow Automation (WFA) commands on the playground database to ensure that the scripts are working fine and not affecting the WFA database tables.

About this task

The default WFA installation path is used in this procedure. If you changed the default location during installation, you must use the changed WFA installation path.

Steps

1. Open a command prompt on the WFA server and change directories to the following location:
`c:\Program Files\NetApp\WFA\mysql\bin`
2. Create a dump of the WFA database using the following command:
`mysqldump -u wfa -pWfa123 --single-transaction --skip-add-drop-table database_tables> dump_location`
Command to create a dump of the cm_storage database tables:
`mysqldump -u wfa -pWfa123 --single-transaction --skip-add-drop-table cm_storage> c:\tmp\cmSt2.sql`
3. Restore the dump you have created on to the WFA playground database using the following command:
`mysql -u wfa -pWfa123 playground < dump_location`
`mysql -u wfa -pWfa123 playground < c:\tmp\cmSt2.sql`
4. Create or edit a WFA command and write the reservation script in the **Reservation** tab.

You must ensure that the reservation and verification scripts use only the playground database.

5. Create or edit a workflow, include the command in the workflow, and then execute the workflow.
6. Verify that the reservation and verification scripts are working as expected.

The WFA data source acquisition process does not update the playground database. You must remove the reservations created by the command manually.

Create a finder

You can create a WFA finder that can search for resources if there is no predefined WFA finder that is suitable for searching the required resources.

What you'll need

You must have created the required filters that are used in the finder.

Steps

1. Click **Workflow Design > Finders**.
2. Click  on the toolbar.
3. In the **Properties** tab of the **New Finder** dialog box, enter or select the required details in the **Name**, **Type**, and **Description** fields.
4. In the **Filters** tab, select the required filters from the **Available Filters** list and click .

You can add or remove filters based on your requirement.

5. In the Returned Attributes tab, select the required attributes for the filter from the **Available** list and click .
6. **Optional:** Click **Test** to test the finder.
 - a. In the **Test Finder <FinderName>** dialog box, enter the required test parameters.
 - b. Clear the **Use reservation data in test** check box if you do not want to use the reservation data for testing the finder.
 - c. Click **Test**.
- The result of the test is displayed.
- d. Close the dialog box.

7. Click **Save**.

Create a filter

You can create a WFA filter that can search for resources if there is no predefined WFA filter that is suitable for the task.

What you'll need

You must know the appropriate SQL syntaxes to create the filter.

Steps

1. Click **Workflow Design > Filters**.
2. Click  on the toolbar.
3. In the **Properties** tab of the **New Filter** dialog box, enter or select the required details in the **Name**, **Dictionary type**, and **Description** fields.
4. In the **Query** tab, enter the appropriate SQL query for the filter.

You must enter a single SQL query and optionally use input parameters. You should use the following syntax to use an input parameter: \${ParameterName}.

```
SELECT
    array.ip
FROM
    storage.array
WHERE
    array.name = '${ArrayName}'
```

5. Click **Refresh** to populate the **Input Parameters** table and the **Returned Attributes** list.

This information is obtained from the SQL query that you have entered. For example, if you use the SQL query example from the previous step, ip is displayed in Returned Attributes and ArrayName is displayed in Input Parameters. You can edit the entries in the **Label** and **Description** columns.

6. **Optional:** Click **Test** to test the filter.
 - a. In the **Test Filter <FilterName>** dialog box, enter the required test parameters.
 - b. Clear the **Use reservation data in test** check box if you do not want to use the reservation data for testing the filter.
 - c. Click **Test**.

The test result is displayed.

 - d. Close the dialog boxes.
7. Click **Save**.

Create a dictionary entry

You can create a WFA dictionary entry when you want to define a new object type and its relationship in your storage environment.

Steps

1. Click **Data Source Design > Dictionary**.
2. Click  on the toolbar.
3. In the **New Dictionary Entry** dialog box, enter the required details in the **Name of object type** and the **Description** fields.
4. For the **Scheme** field, perform one of the following actions:
 - Select one of the available scheme from the list.
 - Click **Add New Scheme**, enter the required **Scheme Name** in the New Scheme dialog box, and then click **Add**.
5. Click **Add row**, and perform the following steps to describe the attribute:
 - a. Click the **Name** column and enter the name of the attribute.
 - b. From the **Type** column, select the required type.

The **String Length** column is populated and is editable if you selected string as the type. Also, the **Values** column is editable if you selected **enum** as the type.

- c. Select the appropriate check boxes for the attribute from the **Natural Key**, **To be Cached**, and **Can be Null** columns.

If you have selected the **Natural Key** check box, you cannot select the **Can be Null** check box.

- d. Add the required attributes for the dictionary object.
- e. **Optional:** Select the **Values in natural key columns are case sensitive** check box if you want the natural keys to be case-sensitive.

6. Click **Save**.

Create a function

You can create a WFA function that can be used as a utility, if there is no predefined WFA function that is suitable for your task.

What you'll need

You must know MVFLEX Expression Language (MVEL) syntaxes to create a function.

About this task

You must include the following for the function definitions:

- Name: name of the function

You must not use a reserved word in MVEL syntax. Each function must have a unique name.

- MVEL definition: a string specifying the MVEL syntax of the function definition

Steps

1. Click **Workflow Design > Functions**.
2. Click  on the toolbar.
3. In the **New Function** dialog box, enter or select the required details in the **Function description** and **Function definition** fields.

```
def actualVolumeSize(data_size, snap_pct)
{
    if (snap_pct < 0) {
        snap_pct = 0;
    } else if (snap_pct > 99) {
        snap_pct = 99;
    }

    div = 1 - (snap_pct/100);
    return (int)(data_size/div);
}
```

The **Function name** field is populated from the data that is used in the MVEL syntax.

4. **Optional:** Click **Test** to test the function:

- a. In the **Expression** section of the **Test** dialog box, enter the required expression of function.

```
actualVolumeSize(600, 1)
```

- b. Click **Test**.

The test result is displayed.

- c. Close the dialog box.

5. Click **Save**.

Create a template

You can create a template that can be used as a blueprint for filling up attributes in command details.

Steps

1. Click **Workflow Design > Templates**.
2. Click  on the toolbar.
3. In the **New Template** dialog box, enter or select the required details in the **Name**, **Type**, and **Description** fields.

The Attributes table is populated based on the dictionary object you have selected in the **Type** field.

4. Click the value column of each attribute and perform one of the following:
 - Enter or select the required value from the list.
 - Enter a user input entry—for example, `$size` for size user input.
5. Click **Save**.

Create a cache query

You can define a cache query when you want to cache information about a dictionary object in the WFA database from a data source type. You can create a cache query and associate it with a dictionary entry and one or more data source types, such as Active IQ Unified Manager 6.1.

What you'll need

You must know the appropriate SQL syntaxes to create a cache query.

Steps

1. Click **Data Source Design > Cache Queries**.
2. Click  on the toolbar.
3. In the **Add Cache Query** dialog box, select the required dictionary entry and data source type.
4. In the “SQL select query” section, enter the appropriate SQL query.

The following SQL query caches information about the disk dictionary object from the Active IQ Unified Manager 6.1 data source type:

```

SELECT
    disk.objId AS id,
    disk.name AS NAME,
    disk.uid AS uid,
    disk.effectiveInterfaceType AS TYPE,
    disk.rpm AS rpm,
    disk.homeNodeId AS home_node_id,
    disk.ownerNodeId AS owner_node_id,
    disk.model AS model,
    disk.serialNumber AS serial_number,
    disk.totalBytes/1024/1024 AS size_mb,
    disk.shelf AS shelf,
    disk.shelfBay AS shelf_bay,
    disk.pool AS pool,
    disk.vendor AS vendor,
    LOWER(disk.raidPosition) AS raid_position,
    disk.containerTypeRaw AS container_type,
    disk.clusterId AS cluster_id
FROM
    netapp_model_view.disk disk

```

5. If you want to test the SQL query, click **Test**.

If you have selected more than one data source type, the Test Cache Query dialog box opens and enables you to select the required data source type.

The test result is displayed.

6. Close the dialog box.
7. Click **Save**.

Create recurring schedules

OnCommand Workflow Automation (WFA) provides two scheduling options for workflows. You can either schedule a workflow to execute once at a specific time or you can create recurring schedules and associate the schedules to workflows so that the workflows are executed routinely.

About this task

A schedule that you have created can be reused and associated with several workflows.

Steps

1. Click **Execution > Schedules**.
2. Click  on the toolbar.
3. In the **New Schedule** dialog box, enter or select a name, description, and frequency for the schedule.

For frequency, you must enter time in 24-hour format. The WFA server time is applied to the schedules.

4. Click **OK**.

After you finish

- You can associate the schedule to a workflow when you execute the workflow by using the **Execute recurrently** option.
- You can view the details of a workflow and its association with a schedule by clicking **Execution > Recurring Schedules**.

The resource and execution planning for the workflows that are scheduled to execute once are done immediately when the workflows are scheduled. However, the resource and execution planning for the workflows with recurring schedule occur at the scheduled time and not when the schedule is associated with a workflow.

Define filter rules

You can define a set of rules for filtering dictionary entry resources such as vFiler units, aggregates, and virtual machines. You can customize the rules for existing workflows and for new workflows while you are creating them.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Workflow Design > Workflows**.
3. In the **Workflows** window, double-click the workflow that you want to modify.

The Workflow <workflow name> window is displayed.

4. Define a set of rules by choosing one of the following options:

If you want to...	Then do this...
Search for resources when the commands in a row are repeated	<ol style="list-style-type: none">a. Click a row number and select Repeat row.b. In the Row Repetitions dialog box, select the For every resource in the group option from the Repeats drop-down list.c. Select a resource type.d. Click the Enter search criteria link.
Search for resources required in command inputs	<ol style="list-style-type: none">a. Click .b. In the Parameters for <command_name> dialog box, select the by searching for an existing <dictionary object> option from the Define <dictionary object> drop-down list.c. Click the Enter search criteria link.

If you want to...	Then do this...
Search for resources referenced by variables in command inputs	<ol style="list-style-type: none"> Click . In the Parameters for <command_name> dialog box, select the by filling in attributes option from the Define <dictionary object> drop-down list. Click  for a field marked with .
Name command inputs of String type	<ol style="list-style-type: none"> Click . In the Parameters for <command_name> dialog box, select the by filling in attributes option from the Define <dictionary object> drop-down list. Click  for a string field.

5. In the **Resource Selection** dialog box, select the **Define filter rules** check box.

If you have selected one of the options from the Finder drop-down of Resource Selection dialog box, the Define filter rules check box is disabled. The value for the finder must be set to “None” for the Define filter rules to be enabled.

6. Enter the attribute, operator, and value for the rule.

The value must be provided within single quotation marks. The filter rules can contain one or more groups.

7. Click **OK**.

Add approval points

You can add an approval point as a checkpoint in a workflow to pause the workflow execution and resume it based on your approval. You can use approval points for incremental execution of a workflow, where sections of the workflow are executed only after a certain condition is met—for example, when the next section has to be approved or when successful execution of the first section is validated.

Steps

1. Log in to WFA through a web browser as an architect or an admin.
2. Click **Workflow Design > Workflows**.
3. In the **Workflows** window, double-click the workflow that you want to modify.
4. In the **Workflow <workflow name>** window, click the  icon to the left of the step for which you want to add the approval point.

You can add approval points for one or more steps.

5. In the **New Approval Point** dialog box, provide the comment and condition details.
6. Click **OK**.

Coding guidelines for WFA

You should understand the general OnCommand Workflow Automation (WFA) coding guidelines, naming conventions, and recommendations on creating various building blocks such as filters, functions, commands, and workflows.

Guidelines for variables

You must be aware of the guidelines for PowerShell and Perl variables in OnCommand Workflow Automation (WFA) when you create a command or a data source type.

PowerShell variables

Guidelines	Example
For script input parameters: <ul style="list-style-type: none">• Use Pascal case.• Do not use underscores.• Do not use abbreviations.	\$VolumeName \$AutoDeleteOptions \$Size
For script internal variables: <ul style="list-style-type: none">• Use Camel case.• Do not use underscores.• Do not use abbreviations.	\$newVolume \$qtreeName \$time
For functions: <ul style="list-style-type: none">• Use Pascal case.• Do not use underscores.• Do not use abbreviations.	GetVolumeSize
Variable names are not case-sensitive. However, to improve readability, you should not use different capitalization for the same name.	\$variable is the same as \$Variable.
Variable names should be in plain English and should be related to the functionality of the script.	Use \$name and not \$a.
Declare the data type for each variable, explicitly.	[string]name [int]size
Do not use special characters (! @ # & % , .) and spaces.	None

Guidelines	Example
Do not use PowerShell reserved keywords.	None
Group the input parameters by placing the mandatory parameters first followed by the optional parameters.	<pre data-bbox="848 276 1354 656"> param([parameter(Mandatory=\$true)] [string]\$Type, [parameter(Mandatory=\$true)] [string]\$Ip, [parameter(Mandatory=\$false)] [string]\$VolumeName) </pre>
Comment all input variables using <i>HelpMessage</i> annotation with a meaningful help message.	<pre data-bbox="848 783 1452 889"> [parameter(Mandatory=\$false,HelpMessage="LUN to map")] [string]\$LUNName </pre>
Do not use “Filer” as a variable name; use “Array” instead.	None
Use <i>ValidateSet</i> annotation in cases where the argument gets enumerated values. This automatically translates to Enum data type for the parameter.	<pre data-bbox="848 1136 1452 1311"> [parameter(Mandatory=\$false,HelpMessage="Volume state")] [ValidateSet("online","offline","restricted")] [string]\$State </pre>
Add an alias to a parameter that ends with “_Capacity” to indicate that the parameter is of capacity type.	<p data-bbox="824 1400 1416 1463">The “Create Volume” command uses aliases as follows:</p> <pre data-bbox="848 1526 1452 1776"> [parameter(Mandatory=\$false,HelpMessage="Volume increment size in MB")] [Alias("AutosizeIncrementSize_Capacity")] [int]\$AutosizeIncrementSize </pre>

Guidelines	Example
Add an alias to a parameter that ends with “_Password” to indicate that the parameter is of password type.	<pre data-bbox="850 198 1459 671"> param ([parameter(Mandatory=\$false, HelpMessage="In order to create an Active Directory machine account for the CIFS server or setup CIFS service for Storage Virtual Machine, you must supply the password of a Windows account with sufficient privileges")] [Alias("Pwd_Password")] [string]\$ADAdminPassword) </pre>

Perl variables

Guidelines	Example
For script input parameters: <ul style="list-style-type: none"> • Use Pascal case. • Do not use underscores. • Do not use abbreviations. 	\$VolumeName \$AutoDeleteOptions \$Size
Do not use abbreviations for script internal variables.	\$new_volume \$qtree_name \$time
Do not use abbreviations for functions.	get_volume_size
Variable names are case-sensitive. To improve readability, you should not use different capitalization for the same name.	\$variable is not the same as \$Variable.
Variable names should be in plain English and should be related to the functionality of the script.	Use \$name and not \$a.
Group the input parameters by placing the mandatory parameters first, followed by the optional parameters.	None

Guidelines	Example
In GetOptions function, explicitly declare the data type of each variable for input parameters.	<pre data-bbox="848 200 1224 348">GetOptions ("Name=s"=>\\$Name, "Size=i"=>\\$Size)</pre>
Do not use "Filer" as a variable name; use "Array" instead.	None
Perl does not include the <i>ValidateSet</i> annotation for enumerated values. Use explicit "if" statements for cases where argument gets enumerated values.	<pre data-bbox="848 580 1460 982">if (defined\$SpaceGuarantee&&! (\$SpaceGuaranteeeq 'none' \$SpaceGuaranteeeq 'volume' \$SpaceGuaranteeeq 'file')) { die'Illegal SpaceGuarantee argument: \''.\$SpaceGuarantee.'\''; }</pre>
All Perl WFA commands must use the "strict" pragma to discourage the use of unsafe constructs for variables, references, and subroutines.	<pre data-bbox="848 1098 1346 1288">use strict; # the above is equivalent to use strictvars; use strictsubs; use strictrefs;</pre>
<p>All Perl WFA commands must use the following Perl modules:</p> <ul data-bbox="155 1467 277 1499" style="list-style-type: none"> • Getopt <p>This is used for specifying input parameters.</p> <ul data-bbox="155 1594 285 1626" style="list-style-type: none"> • WFAUtil <p>This is used for utility functions that are provided for command logging, reporting command progress, connecting to array controllers, and so on.</p>	<pre data-bbox="848 1404 1150 1510">use Getopt:::Long; use NaServer; use WFAUtil;</pre>

Guidelines for indentation

You must be aware of the guidelines for indentation when writing a PowerShell or Perl script for OnCommand Workflow Automation (WFA).

Guidelines	Example
A tab is equal to four empty spaces.	
Use tabs and braces to show the beginning and end of a block.	<p>PowerShell script</p> <pre>if (\$pair.length-ne 2) { throw "Got wrong input data" }</pre>
Add blank lines between sets of operations or chunks of code.	<p>Perl script</p> <pre>if defined \$MaxDirectorySize { # convert from MBytes to Bytes my \$MaxDirectorySizeBytes = \$MaxDirectorySize * 1024 * 1024; }</pre>
	<pre>\$options=\$option.trim(); \$pair=\$option.split(" "); Get-WFALogger -Info -messages \$("split options: "+ \$Pair)</pre>

Guidelines for comments

You must be aware of the guidelines for PowerShell and Perl comments in your scripts for OnCommand Workflow Automation (WFA).

PowerShell comments

Guidelines	Example
Use the # character for a single line comment.	<pre># Single line comment \$options=\$option.trim();</pre>
Use the # character for an end of line comment.	<pre>\$options=\$option.trim(); # End of line comment</pre>
Use the <# and #> characters for a block comment.	<pre><# This is a block comment #> \$options=\$option.trim();</pre>

Perl comments

Guidelines	Example
Use the # character for single line comment.	<pre># convert from MBytes to Bytes my \$MaxDirectorySizeBytes = \$MaxDirectorySize * 1024 * 1024;</pre>
Use the # character for end of line comment.	<pre>my \$MaxDirectorySizeBytes = \$MaxDirect orySize * 1024 * 1024; # convert to Bytes</pre>

Guidelines	Example
<p>Use the # character in every line with an empty # at the beginning and end to create a comment border for multi-line comments.</p>	<pre># # This is a multi-line comment. Perl 5, unlike # Powershell, does not have direct support for # multi-line comments. Please use a '#' in every line # with an empty '#' at the beginning and end to create # a comment border #</pre>
<p>Do not include commented and dead code in WFA commands. However, for testing purposes, you can use the Plain Old Documentation (POD) mechanism to comment out the code.</p>	<pre>=begin comment # Set deduplication if(defined \$Deduplication && \$Deduplication eq "enabled") { \$wfaUtil->sendLog ("Enabling Deduplication"); } =end comment =cut</pre>

Guidelines for logging

You must be aware of the guidelines for logging when writing a PowerShell or Perl script for OnCommand Workflow Automation (WFA).

PowerShell logging

Guidelines	Example
<p>Use the Get-WFALogger cmdlet for logging.</p>	<pre>Get-WFALogger -Info -message "Creating volume"</pre>

Guidelines	Example
Log every action that requires interaction with internal packages such as Data ONTAP, VMware, and PowerCLI. All the log messages are available in Execution Logs in the execution status history of workflows.	None
Log every relevant argument that is passed to internal packages.	None
Use appropriate log levels when using the Get-WFALogger cmdlet, depending on the usage context. -Info, -Error, -Warn, and -Debug are the various available log levels. If a log level is not specified, then the default log level is Debug.	None

Perl logging

Guidelines	Example
Use the WFAUtil sendLog for logging.	<pre data-bbox="848 908 1387 1140"> my wfa_util = WFAUtil->new(); eval { \$wfa_util->sendLog('INFO', "Connecting to the cluster: \$DestinationCluster"); }</pre>
Log every action that requires interaction with anything external to the command such as Data ONTAP, VMware, and WFA. All the log messages that you create using the WFAUtil sendLog routine are stored in the WFA database. These log messages are available for the executed workflow and command.	None
Log every relevant argument passed to the routine that was called.	None
Use appropriate log levels. -Info, -Error, -Warn, and -Debug are the various available log levels.	None

Guidelines	Example
<p>When logging at the -Info level, be precise and concise. Do not specify implementation details such as class name and function name in log messages. Describe the exact step or the exact error in plain English.</p>	<p>The following code snippet shows an example of a good message and a bad message:</p> <pre data-bbox="856 297 1313 445"> \$wfa_util->sendLog('WARN', "Removing volume: '.'.\$VolumeName); # Good Message </pre> <pre data-bbox="856 551 1362 699"> \$wfa_util->sendLog('WARN', 'Invoking volume- destroy ZAPI: '.\$VolumeName); # Bad message </pre>

Guidelines for error handling

You must be aware of the guidelines for error handling when writing a PowerShell or Perl script for OnCommand Workflow Automation (WFA).

PowerShell error handling

Guidelines	Example
<p>Common parameters added to cmdlets by PowerShell runtime include error handling parameters such as ErrorAction and WarningAction:</p> <ul data-bbox="155 1269 796 1607" style="list-style-type: none"> • The ErrorAction parameter determines how a cmdlet should react to a non-terminating error from the command. • The WarningAction parameter determines how a cmdlet should react to a warning from the command. • Stop, SilentlyContinue, Inquire, and Continue are the valid values for the ErrorAction and WarningAction parameters. <p>For more information, you can use the <code>Get-Help about_CommonParameters</code> command in PowerShell CLI.</p>	<p>ErrorAction: the following example shows how to handle a non-terminating error as a terminating error:</p> <pre data-bbox="856 1269 1395 1374"> New-NcIgroup-Name \$IgroupName- Protocol \$Protocol-Type\$OSType- ErrorActionstop </pre> <p>WarningAction</p> <pre data-bbox="856 1543 1428 1691"> New-VM-Name \$VMName-VM \$SourceVM- DataStore\$DataStoreName- VMHost\$VMHost- WarningActionSilentlyContinue </pre>

Guidelines	Example
<p>Use the general “try/catch” statement if the type of the incoming exception is unknown.</p>	<pre>try { "In Try/catch block" } catch { "Got exception" }</pre>
<p>Use the specific “try/catch” statement if the type of the incoming exception is known.</p>	<pre>try { "In Try/catch block" } catch[System.Net.WebException], [System.IO. IOException] { "Got exception" }</pre>
<p>Use the “finally” statement to release resources.</p>	<pre>try { "In Try/catch block" } catch { "Got exception" } finally { "Release resources" }</pre>

Guidelines	Example
<p>Use PowerShell automatic variables to access information about exceptions.</p>	<pre data-bbox="845 192 1416 713"> try { Get-WFALogger -Info -message \$("Creating Ipspace: " + \$Ipspace) New-NaNetIpspace-Name \$Ipspace } catch { Throw "Failed to create Ipspace. Message: " + \$_.Exception.Message; } </pre>

Perl error handling

Guidelines	Example
<p>Perl does not include native language support for try/catch blocks. Use eval blocks for checking and handling errors. Keep eval blocks as small as possible.</p>	<pre data-bbox="845 196 1457 1262"> eval { \$wfa_util->sendLog('INFO', "Quiescing the relationship : \$DestinationCluster://\$Destination Vserver /\$DestinationVolume"); \$server->snapmirror_quiesce('destination-vserver' => \$DestinationVserver, 'destination-volume' => \$DestinationVolume); \$wfa_util->sendLog('INFO', 'Quiesce operation started successfully.'); }; \$wfa_util->checkEvalFailure("Failed to quiesce the SnapMirror relationship \$DestinationCluster://\$Destination Vserver /\$DestinationVolume", \$@); </pre>

General PowerShell and Perl conventions for WFA

You must understand certain PowerShell and Perl conventions that are used in WFA to create scripts that are consistent with existing scripts.

- Use variables that help to clarify what you want the script to do.
- Write readable code that can be understood without comments.
- Keep the scripts and commands as simple as possible.
- For PowerShell scripts:
 - Use cmdlets whenever possible.
 - Invoke .NET code when there is no cmdlet available.
- For Perl scripts:
 - Always end “die” statements with newline characters.

In the absence of a newline character, the script line number is printed, which is not useful for debugging Perl commands executed by WFA.

- In the “GetOpt” module, make the string arguments to a command mandatory.

Perl modules bundled with Windows

Some Perl modules are bundled with the Windows Active state Perl distribution for OnCommand Workflow Automation (WFA). You can use these Perl modules in your Perl code for writing commands, only if they are bundled with Windows.

The following table lists the Perl database modules that are bundled with Windows for WFA.

Database module	Description
DBD::mysql	Perl5 database interface driver that enables you to connect to the MySQL database.
Try::Tiny	Minimizes common mistakes with evaluation blocks.
XML::LibXML	Interface to libxml2 that provides XML and HTML parsers with DOM, SAX, and XMLReader interfaces.
DBD::Cassandra	Perl5 database interface driver for Cassandra that uses the CQL3 query language.

Considerations for adding custom PowerShell and Perl modules

You must be aware of certain considerations before adding custom PowerShell and Perl modules to OnCommand Workflow Automation (WFA). Custom PowerShell and Perl modules enable you to use custom commands for creating workflows.

- During the execution of WFA commands, all custom PowerShell modules are added to the WFA install directory `/Posh/modules` are automatically imported.
- All custom Perl modules added to the `WFA/perl` directory are included in the `@INC` library.
- Custom PowerShell and Perl modules are not backed up as part of the WFA backup operation.
- Custom PowerShell and Perl modules are not restored as part of the WFA restore operation.

You must manually back up custom PowerShell and Perl modules in order to copy them to a new WFA installation.

The folder name in modules' directory must be same as that of the module name.

WFA cmdlets and functions

OnCommand Workflow Automation (WFA) provides several PowerShell cmdlets as well as PowerShell and Perl functions that you can use in your WFA commands.

You can view all the PowerShell cmdlets and functions provided by the WFA server using the following

PowerShell commands:

- `Get-Command -Module WFAWrapper`
- `Get-Command -Module WFA`

You can view all the Perl functions provided by the WFA server in the `WFAUtil.pm` module. The help sections, WFA PowerShell cmdlets help and WFA Perl methods help, of the WFA Help module Support Links enables access to all the PowerShell cmdlets and functions and the Perl functions.

PowerShell and Perl WFA modules

You must be aware of the PowerShell or Perl modules for OnCommand Workflow Automation (WFA) to write scripts for your workflows.

PowerShell modules

Guidelines	Example
Use the Data ONTAP PS Toolkit to invoke APIs whenever the toolkit is available.	The <code>Add_VLAN</code> command uses the toolkit as follows: <code>Add-NaNetVlan-Interface \$Interface-Vlans\$VlanID</code>
If there are no cmdlets available in the Data ONTAP PS Toolkit, use the <code>Invoke-SSH</code> command to invoke the CLI on Data ONTAP.	<code>Invoke-NaSsh-Name \$ArrayName-Command "ifconfig -a"-Credential \$Credentials</code>

Perl modules

The `NaServer` module is used in WFA commands. The `NaServer` module allows the invocation of Data ONTAP APIs, which are used in active management of Data ONTAP systems.

Guidelines	Example
<p>Use the NaServer module to invoke APIs whenever the NetApp Manageability SDK is available.</p>	<p>The following example shows how the NaServer module is used for a resume SnapMirror operation:</p> <pre> eval { \$wfa_util->sendLog('INFO', "Connecting to the cluster: \$DestinationCluster"); my \$server = \$wfa_util- >connect(\$DestinationClusterIp, \$DestinationVserver); my \$sm_info = \$server- >snapmirror_get('destination-vserver' => \$DestinationVserver, 'destination-volume' => \$DestinationVolume); my \$sm_state = \$sm_info- >{ 'attributes' }->{ 'snapmirror- info' }->{ 'mirror-state' }; my \$sm_status = \$sm_info- >{ 'attributes' }->{ 'snapmirror- info' }->{ 'relationship-status' }; \$wfa_util->sendLog('INFO', "SnapMirror relationship is \$sm_state (\$sm_status)"); if (\$sm_status ne 'quiesced') { \$wfa_util->sendLog('INFO', 'The status needs to be quiesced to resume transfer.'); } else { my \$result = \$server- >snapmirror_resume('destination-vserver' => \$DestinationVserver, 'destination-volume' => \$DestinationVolume); \$wfa_util->sendLog('INFO', 177 "Result of resume: \$result"); } } </pre>

Guidelines	Example
<p>If a Data ONTAP API is not available, invoke the Data ONTAP CLI using the executeSystemCli utility method.</p> <p> executeSystemCli is not supported and is currently available only for Data ONTAP operating in 7-Mode.</p>	None

Considerations while converting PowerShell commands to Perl

You must be aware of certain important considerations when you convert PowerShell commands to Perl because PowerShell and Perl have different capabilities.

Command input types

OnCommand Workflow Automation (WFA) allows workflow designers to use arrays and hash as input for the command when defining a command. These input types cannot be used when the command is defined using Perl. If you want a Perl command to accept array and hash inputs, you can define the input as a string in the designer. The command definition can then parse the input, which is passed to create an array or hash as required. The description for the input describes the format in which the input is expected.

```
my @input_as_array = split ',', $InputString; #Parse the input string of
format val1,val2 into an array

my %input_as_hash = split /[:=]/, $InputString; #Parse the input string of
format key1=val1;key2=val2 into a hash.
```

PowerShell statement

The following examples show how an array input can be passed into PowerShell and Perl. The examples describe the input CronMonth, which specifies the month when the cron job is scheduled to run. The valid values are integers -1 to 11. A value of -1 indicates that the schedule executes every month. Any other value denotes a specific month, with 0 being January and 11 being December.

```
[parameter(Mandatory=$false, HelpMessage="Months in which the schedule
executes. This is a comma separated list of values from 0 through 11.
Value -1 means all months.")]
[ValidateRange(-1, 11)]
[array]$CronMonths,
```

Perl statement

```

GetOptions(
    "Cluster=s"          => \$Cluster,
    "ScheduleName=s"     => \$ScheduleName,
    "Type=s"              => \$Type,
    "CronMonths=s"        => \$CronMonths,
) or die 'Illegal command parameters\n';

sub get_cron_months {
    return get_cron_input_hash('CronMonths', $CronMonths, 'cron-month',
-1,
    11);
}

sub get_cron_input_hash {
    my $input_name    = shift;
    my $input_value   = shift;
    my $zapi_element = shift;
    my $low           = shift;
    my $high          = shift;
    my $exclude       = shift;

    if (!defined $input_value) {
        return undef;
    }

    my @values = split ',', $input_value;

    foreach my $val (@values) {
        if ($val !~ /^[+-]?\d+$/) {
            die
                "Invalid value '$input_value' for $input_name: $val must
be an integer.\n";
        }
        if ($val < $low || $val > $high) {
            die
                "Invalid value '$input_value' for $input_name: $val must
be from $low to $high.\n";
        }
        if (defined $exclude && $val == $exclude) {
            die
                "Invalid value '$input_value' for $input_name: $val is not
valid.\n";
        }
    }
    # do something
}

```

Command definition

A one-line expression in PowerShell using a pipe operator might have to be expanded into multiple blocks of statements in Perl in order to achieve the same functionality. An example from one of the wait commands is shown in the following table.

PowerShell statement	Perl statement
<pre># Get the latest job which moves # the specified volume to the # specified # aggregate. \$job = Get-NcJob -Query \$query where {\$_.JobDescription -eq "Split" + \$VolumeCloneName} Select-Object -First 1</pre>	<pre>my \$result = \$server- >job_get_iter('query' => { 'job-type' => 'VOL_CLONE_SPLIT' }, 'desired-attributes' => { 'job-type' => '', 'job-description' => '', 'job-progress' => '', 'job-state' => '' }); my @jobarray; for my \$job (@{ \$result- >{ 'attributes-list' } }) { my \$description = \$job->{ 'job- description' }; if(\$description =~ /\$VolumeCloneName/) { push(@jobarray, \$job) } }</pre>

Guidelines for WFA building blocks

You must be aware of the guidelines for using Workflow Automation building blocks.

Guidelines for SQL in WFA

You must be aware of the guidelines for using SQL in OnCommand Workflow Automation (WFA) to write SQL queries for WFA.

SQL is used in the following places in WFA:

- SQL queries to populate user inputs for selection
- SQL queries for creating filters to filter objects of a specific dictionary entry type
- Static data in tables in the playground database

- A custom data source type of SQL type where the data has to be extracted from an external data source such as a custom configuration management database (CMDB).
- SQL queries for reservation and verification scripts

Guidelines	Example
SQL reserved keywords must be in uppercase characters.	<pre data-bbox="850 354 1383 502"> SELECT vserver.name FROM cm_storage.vserver vserver </pre>
Table and column names must be in lowercase characters.	Table: aggregate Column: used_space_mb
Separate words with an underscore (_) character. Spaces are not allowed.	array_performance
Table name is defined in singular. A table is a collection of one or more entries.	“function”, not “functions”
Use table aliases with meaningful names in SELECT queries.	<pre data-bbox="850 1009 1416 1516"> SELECT vserver.name FROM cm_storage.cluster cluster, cm_storage.vserver vserver WHERE vserver.cluster_id = cluster.id AND cluster.name = '\${ClusterName}' AND vserver.type = 'cluster' ORDER BY vserver.name ASC </pre>

Guidelines	Example
<p>If you have to refer to a filter input parameter or user input parameter in a filter query or user query, use the syntax as <code>`\${inputVariableName}`</code>. You can also use the syntax to refer to a command definition parameter in reservation scripts and verification scripts.</p>	<pre> SELECT volume.name AS Name, aggregate.name as Aggregate, volume.size_mb AS 'Total Size (MB)', volume.used_size_mb AS 'Used Size (MB)', volume.space_guarantee AS 'Space Guarantee' FROM cm_storage.cluster, cm_storage.aggregate, cm_storage.vserver, cm_storage.volume WHERE cluster.id = vserver.cluster_id AND aggregate.id = volume.aggregate_id AND vserver.id = voulme.vserver_id AND vserver.name = `\${VserverName}` AND cluster.name = `\${ClusterName}` ORDER BY volume.name ASC </pre>
<p>Use comments for complex queries. Some of the supported comment styles in queries are as follows:</p> <ul style="list-style-type: none"> “<code>--</code> until the end of the line <p>A space is mandatory after the second hyphen in this comment style.</p> <ul style="list-style-type: none"> From a <code>#</code> character until the end of the line From a “/” to the following “/” sequence 	<pre> /* multi-line comment */ --line comment SELECT ip as ip, # comment till end of this line NAME as name FROM --end of line comment storage.array </pre>

Guidelines for WFA functions

You can create functions to encapsulate commonly used and more complex logic in a named function, and then reuse the function as command parameter values or filter parameters values in OnCommand Workflow Automation (WFA).

Guidelines	Example
Use Camel case for a function name.	calculateVolumeSize
Variable names should be in plain English and related to the functionality of the function.	splitByDelimiter
Do not use abbreviations.	calculateVolumeSize, <i>not</i> calcVolSize
Functions are defined using MVFLEX Expression Language (MVEL).	None
The function definition should be specified according to the official Java Programming Language guidelines.	None

Guidelines for WFA dictionary entries

You must be aware of the guidelines for creating dictionary entries in OnCommand Workflow Automation (WFA).

Guidelines	Example
Dictionary entry names must contain only alphanumeric characters and underscores.	Cluster_License Switch_23
Dictionary entry names must start with an uppercase character. Begin every word in the name with an uppercase character and separate each word with an underscore (_).	Volume Array_License
Dictionary entry attribute names should not include the name of the dictionary entry.	None
Attributes and references in a dictionary entry must be in lowercase characters.	aggregate, size_mb
Separate words with an underscore. Spaces are not allowed.	resource_pool

Guidelines	Example
Dictionary entries cannot include references that are from a different scheme. When a dictionary entry requires cross-reference to an object in a different scheme, ensure that all the natural keys of the object being referred to are present in the dictionary entry.	Array_Performance dictionary entry requires all the natural keys of the Array dictionary entry as direct attributes in it.
Use appropriate data types for attributes.	None
Use Long data type for size or space-related attributes.	size_mb and available_size_mb in storage.Volume dictionary entry
Use Enum when an attribute has a fixed set of values.	raid_type in storage.Volume dictionary entry
Set “To be Cached” as true for an attribute or reference when a data source provides value for that attribute or reference. For Active IQ Unified Manager data source, add cacheable attributes if the data source can provide the value to it.	None
Set “Can be Null” as true if the data source providing the value for this attribute or reference can return NULL.	None
Provide a meaningful description to each attribute and reference. The description is displayed in command details when designing a workflow.	None
Do not use “id” as the name of an attribute in dictionary entries. It is reserved for internal WFA usage.	None

Related information

[References to learning material](#)

Guidelines for commands

You must be aware of the guidelines for creating commands in OnCommand Workflow Automation (WFA).

Guidelines	Example
Use an easily identifiable name for commands.	Create_Qtree
Use spaces to delimit words and each word must start with an uppercase character.	Create_Volume

Guidelines	Example
Provide a description to explain the functionality of the command, including the expected outcome of the optional parameters.	None
By default, the timeout for standard commands is 600 seconds. The default timeout is set while creating the command. Change the default value only if the command might take a longer time to complete.	Create Volume command
In case of long-running operations, create two commands—one to invoke the long-running operation and another to report the progress of the operation periodically. The first command should be a Standard Execution command type and the second should be Wait for Condition command type.	Create VSM and Wait for VSM commands
Prefix the Wait for condition command names with "Wait" for easy identification.	Wait for CM Volume Move
Use an appropriate waiting interval for the "Wait for condition" commands. The specified value governs the interval at which the polling command gets executed to check if the long-running operation is complete.	60s sampling interval for the Wait for VSM command
For the Wait for condition commands, use an appropriate timeout based on the expected time for the long-running operation to complete. The expected time might be considerably longer if the operation involves data transfer over a network.	A VSM baseline transfer can take many days to complete. Therefore, the specified timeout is 6 days.

String representation

The string representation for a command displays the details of a command in a workflow design during planning and execution. Only the command parameters can be used in the string representation for a command.

Guidelines	Example
Avoid using attributes that do not have any value. An attribute without a value is displayed as NA.	VolName 10.68.66.212[NA]aggr1/testVol7
Separate different entries in the string representation using the following delimiters: [] , / :	ArrayName [ArrayIp]
Provide meaningful labels to every value in string representation.	Volume name=VolumeName

Command definition language

Commands can be written using the following supported scripting languages:

- PowerShell
- Perl

Command parameter definition

The command parameters are described by Name, Description, Type, a default value for the parameter, and whether the parameter is mandatory. The parameter type can be String, Boolean, Integer, Long, Double, Enum, DateTime, Capacity, Array, Hashtable, Password, or an XmlDocument. While the values for most of the types are intuitive, the values for Array and Hashtable should be in a particular format as described in the following table:

Guidelines	Example
Ensure that the value for an Array input type is a list of values, separated by comma.	<pre>[parameter (Mandatory=\$false, HelpMessage="Months in which the schedule executes.")] [array]\$CronMonths</pre> <p>Input is passed as following: 0,3,6,9</p>
Ensure that the value for a Hashtable input type is a list of key=value pairs, separated by semicolon.	<pre>[parameter (Mandatory=\$false, HelpMessage="Volume names and size (in MB)")] [hashtable]\$VolumeNamesAndSize</pre> <p>Input is passed as following: Volume1=100;Volume2=250;Volume3=50</p>

Guidelines for workflows

You must be aware of the guidelines for creating or modifying a predefined workflow for OnCommand Workflow Automation (WFA).

General guidelines

Guidelines	Example
Name the workflow such that it reflects the operation that is executed by the storage operator.	Create a CIFS Share

Guidelines	Example
For workflow names, capitalize the initial letter of the first word and every word that is an object. Capitalize letters for abbreviations and acronyms.	Volume Qtree Create a Clustered Data ONTAP Qtree CIFS Share
For workflow descriptions, include all of the important steps of the workflow, including any prerequisites, result of the workflow, or conditional aspects of execution.	See the description of the sample workflow Create VMware NFS Datastore on Clustered Data ONTAP Storage , which includes the prerequisites.
Set “Ready For Production” to <code>true</code> only when the workflow is ready for production and can be displayed in the portal page.	None
By default, set “Consider reserved elements” to <code>true</code> . When previewing a workflow for execution, the WFA planner considers all of the objects that are reserved along with the existing objects in the cache database. Effects of other scheduled workflows or workflows executing in parallel are considered when planning a specific workflow if this option is set to <code>true</code> .	<ul style="list-style-type: none"> • Scenario 1 <p>Workflow 1 creates a volume, and is scheduled to execute one week later. Workflow 2 creates qtrees or LUNs in volumes that are searched for, and if workflow 2 is executed within a day or so, you should turn off “Consider reserved elements” for workflow 2 to prevent it from considering the volume that is to be created in a week.</p> <ul style="list-style-type: none"> • Scenario 2 <p>Workflow 1 uses the <code>Create Volume</code> command. If there is a scheduled workflow 2 that consumes 100 GB from an aggregate, then workflow 1 must consider the requirements for workflow 2 during planning.</p>

Guidelines	Example
<p>By default, “Enable element existence validation” is set to true.</p>	<ul style="list-style-type: none"> Scenario 1 <p>If you create a workflow that first removes a volume by name using the command <code>Remove Volume</code> only if the volume exists, and the re-creates it using another command such as <code>Create Volume</code> or <code>Clone Volume</code>, then the workflow should not use this flag. The effect of removing the volume will not be available to the <code>Create volume</code> command, thereby causing the workflow to fail.</p> Scenario 2 <p>The <code>Create Volume</code> command is used in a workflow with a specific name as “vol198”.</p> <p>If this option is set to true, WFA planner checks during planning to see if a volume by that name exists in the given array. If the volume exists, the workflow fails during planning.</p>
<p>When the same command is selected more than once in a workflow, provide appropriate display names for the command instances.</p>	<p>The “Create, map, and protect LUNs with SnapVault” sample workflow uses the <code>Create Volume</code> command twice. However, it uses the display names <code>as Create Primary Volume</code> and <code>Create Secondary Volume</code> appropriately for the primary volume and the mirrored destination volume.</p>

User inputs

Guidelines	Example
<p>Names:</p> <ul style="list-style-type: none"> Start the name with the “\$” character. Use an uppercase letter at the beginning of each word. Use uppercase letters for all terms and abbreviations. Do not use underscores. 	<p>\$Array \$VolumeName</p>

Guidelines	Example
<p>Display names:</p> <ul style="list-style-type: none"> • Use an uppercase letter at the beginning of each word. • Separate words with spaces. • If inputs have specific units, specify the unit in brackets in the display name directly. 	<p>Volume Name</p> <p>Volume Size (MB)</p>
<p>Descriptions:</p> <ul style="list-style-type: none"> • Provide a meaningful description for each user input. • Provide examples when required. <p>You should do this especially when the user input is expected to be in a specific format.</p> <p>The user input descriptions are displayed as tooltips for the user inputs during workflow execution.</p>	<p>Initiators to be added to an “iGroup”. For example, IQN or WWPN of the initiator.</p>
<p>Type: Select Enum as the type if you want to restrict the input to a specific set of values.</p>	<p>Protocol: “iscsi”, “fcp”, “mixed”</p>
<p>Type: Select Query as the type when the user can select from values available in the WFA cache.</p>	<p>\$Array: QUERY type with query as follows:</p> <pre data-bbox="840 1129 1493 1341"> SELECT ip, name FROM storage.array </pre>
<p>Type: Mark the user input as locked when the user input should be restricted to the values that are obtained from a query or should be restricted to only the supported Enum types.</p>	<p>\$Array: Locked Query type: Only arrays in the cache can be selected.\$Protocol: Locked Enum type with valid values as iscsi, fcp, mixed. No other value than the valid value is supported.</p>
<p>Type: Query TypeAdd additional columns as return values in the query when it helps the storage operator to make the right choice of user input.</p>	<p>\$Aggregate: Provide name, total size, available size so that the operator knows the attributes before selecting the aggregate.</p>

Guidelines	Example
<p>Type: Query TypeSQL query for user inputs can refer to any other user inputs preceding it. This can be used to limit the results from a query based on other user inputs such as vFiler units of an array, volumes of an aggregate, LUNs in a storage virtual machine (SVM).</p>	<p>In the sample workflow <i>Create a Clustered Data ONTAP Volume</i>, the query for <i>VserverName</i> is as follows:</p> <pre data-bbox="850 340 1416 846"> SELECT vserver.name FROM cm_storage.cluster cluster, cm_storage.vserver vserver WHERE vserver.cluster_id = cluster.id AND cluster.name = '\${ClusterName}' AND vserver.type = 'cluster' ORDER BY vserver.name ASC </pre> <p>The query refers to \${ClusterName}, where \$ClusterName is the name of the user input preceding the \$VserverName user input.</p>
<p>Type: Use Boolean type with values as “true, false” for user inputs that are Boolean in nature. This helps in writing internal expressions in the workflow design using the user input directly. For example, \$UserInputName rather than \$UserInputName == “Yes”.</p>	<p>\$CreateCIFSShare: Boolean type with valid values as “true” or “false”</p>
<p>Type: For string and number type, use regular expressions in the values column when you want to validate the value with specific formats.</p> <p>Use regular expressions for IP address and network mask inputs.</p>	<p>Location-specific user input can be expressed as “[A-Z][A-Z]\-0[1-9]”. This user input accepts values such as “US-01”, “NB-02”, but not “nb-00”.</p>
<p>Type: For number type, a range-based validation can be specified in the values column.</p>	<p>For Number of LUNs to be created, the entry in the Values column is 1-20.</p>
<p>Group: Group related user inputs into appropriate buckets and name the group.</p>	<p>“Storage Details” for all storage-related user inputs. “Datastore Details” for all VMware-related user inputs.</p>

Guidelines	Example
Mandatory: If the value of any user input is necessary for the workflow to execute, mark the user input as mandatory. This ensures that the user input screen mandatorily accepts that input from the user.	"\$VolumeName" in the "Create NFS Volume" workflow.
Default value: If a user input has a default value that can work for most of the workflow executions, provide the values. This helps in allowing the user to provide fewer inputs during execution, if the default serves the purpose.	None

Constants, variables, and returns parameters

Guidelines	Example
Constants: Define constants when using a common value for defining parameters to multiple commands.	<code>AGGREGATE_OVERCOMMITMENT_THRESHOLD</code> in the Create, map, and protect LUNs with SnapVault sample workflow.
Constants:Names <ul style="list-style-type: none"> Use an uppercase letter at the beginning of each word. Use uppercase letters for all terms and abbreviations. Do not use underscores. Use uppercase letters for all letters of constant names. 	<code>AGGREGATE_USED_SPACE_THRESHOLD</code> <code>ActualVolumeSizeInMB</code>
Variables: Provide a name to an object defined in one of the command parameter boxes. Variables are automatically generated names and can be changed.	None
Variables: Names Use lowercase characters for variable names.	volume1 cifs_share
Return parameters: Use return parameters when the workflow planning and execution should return some calculated or selected values during planning. The values are made available in the preview mode when the workflow is executed from a web service as well.	Aggregate: If the aggregate is selected using the resource selection logic, then the actual selected aggregate can be defined as a return parameter.

Guidelines for creating validation scripts for remote system types

You must be aware of the guidelines for creating validation scripts that are used to test the remote system types that you define in OnCommand Workflow Automation (WFA).

- The Perl script that you create must be similar to the sample script provided in the Validation Script window.
- The output of your validation script must be similar to that of the sample script.

Sample validation script

```

# Check connectivity.
# Return 1 on success.
# Return 0 on failure and set $message
sub checkCredentials {
my ($host, $user, $passwd, $protocol, $port, $timeout) = @_;
#
# Please add the code to check connectivity to $host using $protocol here.
#
return 1;
}

```

Guidelines for creating data source types

You must be aware of the guidelines for creating data source types that are used to define custom data sources for OnCommand Workflow Automation (WFA).

You can define a data source type by using one of the following methods:

- SQL: You can use the WFA SQL guidelines to define select queries from data sources based on an external database.
- SCRIPT: You can write a PowerShell script that provides the data for a specific scheme of dictionary entries.

The guidelines for creating data source types are as follows:

- You should use PowerShell language must be used to create script.
- The PowerShell script should provide the output for each dictionary entry in its current working directory.
- The data files should be named `dictionary_entry.csv`, where the name of the dictionary entry should be in lower-case characters.

The predefined data source type that collects information from Performance Advisor uses a SCRIPT-based data source type. The output files are named `array_performance.csv` and `aggregate_performance.csv`.

- The `.csv` file should include the content in the exact order as that of the dictionary entry attributes.

A dictionary entry includes attributes in the following order: `array_ip`, `date`, `day`, `hour`, `cpu_busy`, `total_ops_per_sec`, `disk_throughput_per_sec`.

The PowerShell script adds data to the `.csv` file in the same order.

```
$values = get-Array-CounterValueString ([REF]$data)
Add-Content $arrayFile ([byte[]][char[]] "\n"
t$arrayIP't$date't$day't$hour't$values'n")
```

- You should use Encoding to ensure that the data output from the script is loaded into the WFA cache accurately.
- You should use \N while entering a Null value in the .csv file.

Reserved words

OnCommand Workflow Automation (WFA) includes some reserved words. You must not use the reserved words in workflows for any attribute or parameters such as variable names, user input, constants, and return parameters.

The following is a list of the reserved words in WFA:

• and	• Float	• proto
• Array	• float	• return
• assert	• for	• Runtime
• Boolean	• foreach	• SecurityManager
• boolean	• function	• Short
• Byte	• if	• short
• byte	• import	• soundslike
• char	• import_static	• StrictMath
• Character	• in	• String
• CharSequence	• instanceof	• StringBuffer
• Class	• int	• StringBuilder
• ClassLoader	• Integer	• strsim
• Compiler	• is	• switch
• contains	• isdef	• System
• convertable_to	• Long	• Thread
• def	• long	• ThreadLocal
• do	• Math	• true
• Double	• new	• until
• double	• null	• var
• else	• Number	• Void
• empty	• Object	• while
• false	• or	• with

How you use REST APIs

You can use REST APIs provided by Workflow Automation (WFA) to invoke workflows from external portals and the data center orchestration software. WFA supports XML and JSON content types for all REST APIs.

WFA allows external services to access various resource collections, such as workflows, users, filters, and finders, through URI paths. The external services can use HTTP methods, such as GET, PUT, POST, and DELETE, on these URIs to perform CRUD operations on the resources.

You can perform several actions through the WFA REST APIs, including the following:

- Access workflow definitions and metadata.
- Execute workflows and monitor their execution.
- View users and roles, and change passwords.
- Execute and test resource selection filters.
- Execute and test resource finders.
- Manage credentials of storage or other data center objects.
- View data sources and data source types.

REST documentation has more information about REST APIs:

https://wfa_server_ip:port/rest/docs

wfa_server_ip is the IP address of your WFA server and port is the TCP port number you have used for the WFA server during installation.



WFA checks Cross-Site Request Forgery (CSRF) tokens for requests coming from the web UI. However, WFA does not check CSRF tokens for incoming REST requests coming from REST clients or orchestration software.

References to learning material

You should be aware of certain scripting and programming practices to create advanced Workflow Automation (WFA) workflows. You can use reference material to learn about the required options before creating WFA building blocks or workflows.

Windows PowerShell

WFA uses PowerShell scripts for workflow operations. The following table includes references to learning material for PowerShell:

Getting Started with Windows PowerShell	http://msdn.microsoft.com/en-us/library/windows/desktop/aa973757(v=vs.85).aspx
---	---

PowerShell Development—Integrated Scripting Environment (ISE)	https://docs.microsoft.com/en-us/powershell/scripting/windows-powershell/ise/introducing-the-windows-powershell-ise?view=powershell-7.2
.NET Framework Naming Guidelines	http://msdn.microsoft.com/en-us/library/xzf533w0%28v=vs.71%29.aspx
PowerShell code style	http://get-powershell.com/post/2011/04/13/Extra-Points-for-Style-when-writing-PowerShell-Code.aspx
PowerShell Try/Catch Finally	http://technet.microsoft.com/en-us/library/dd315350.aspx
PowerShell Automatic Variables	http://technet.microsoft.com/en-us/library/dd347675.aspx
PowerShell Error Reporting	https://docs.microsoft.com/en-us/powershell/scripting/developer/cmdlet/error-reporting-concepts?view=powershell-7.2
PowerShell Common Parameters	https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_commonparameters?view=powershell-7.2

Data ONTAP PowerShell toolkit

The Data ONTAP PowerShell toolkit is bundled along with WFA. You can use the PowerShell toolkit cmdlets to invoke Data ONTAP commands from a PowerShell script. For more information, see the *Data ONTAP PowerShell Toolkit Help*, which you can access from the following location:

`WFA_install_location\WFA\PosH\Modules\DataONTAP\webhelp\index.html`.

`WFA_install_location` is the WFA installation directory, and `C:\Program Files\NetApp` is the default installation directory.

The following table includes references to information about the Data ONTAP PowerShell toolkit:

ONTAP PowerShell Toolkit Articles	https://community.netapp.com/t5/Tech-OnTap-Articles/The-Data-ONTAP-PowerShell-Toolkit/ta-p/85933
ONTAP PowerShell Toolkit NetApp Community	https://community.netapp.com/t5/forums/filteredbylabelpage/board-id/microsoft-cloud-and-virtualization-discussions/label-name/powershell%20toolkit

Perl

WFA supports Perl commands for workflow operations. When you install WFA, the required Perl and Perl modules are installed on the WFA server.

ActivePerl User Guide

You can also access the *ActivePerl User Guide* from the following location:

`WFA_install_location\WFA\Perl64\html\index.html`.

`WFA_install_location` is the WFA installation directory, and `C:\Program Files\NetApp` is the default installation directory.

WFA uses Perl scripts for workflow operations. The following table includes references to learning material for Perl:

Modern Perl: 2014	http://modernperlbooks.com/books/modern_perl_2014/index.html
Perl programming documentation	http://perldoc.perl.org/
Perl programming language	http://www.perl.org/

NetApp Manageability SDK

The required Perl modules of the NetApp Manageability SDK are bundled along with WFA. These Perl modules are required for using the Perl commands in WFA. For more information, see the NetApp Manageability SDK documentation, which you can access from the following location:

`WFA_install_location\WFA\perl\NMSDK\html`.

`WFA_install_location` is the WFA installation directory, and `C:\Program Files\NetApp` is the default installation directory.

Structured Query Language (SQL)

The SQL SELECT syntax is used in filters and to populate user inputs.

[MySQL SELECT syntax](#)

MVFLEX Expression Language (MVEL)

You can use MVEL expression syntaxes in WFA workflows—for example, in functions and variables.

For more information, see the *MVEL Language Guide*.

Regular expressions

You can use regular expression (regex) in WFA.

[ActionScript 3.0 Using regular expressions](#)

Related documentation for OnCommand Workflow Automation

There are additional documents and tools to help you learn to perform more advanced

configuration of your OnCommand Workflow Automation (WFA) server.

Other references

The Workflow Automation space within the NetApp community provides additional learning resources, including the following:

- **NetApp community**

[NetApp community: Workflow Automation \(WFA\)](#)

Tool references

- **Interoperability Matrix**

Lists supported combinations of hardware components and software versions.

[Interoperability Matrix](#)

Manage and configure

Overview of OnCommand Workflow Automation

OnCommand Workflow Automation (WFA) is a software solution that helps to automate storage management tasks, such as provisioning, migration, decommissioning, data protection configurations, and cloning storage. You can use WFA to build workflows to complete tasks that are specified by your processes.

A workflow is a repetitive and procedural task that consists of sequential steps, including the following types of tasks:

- Provisioning, migrating, or decommissioning storage for databases or file systems
- Setting up a new virtualization environment, including storage switches and datastores
- Setting up storage for an application as part of an end-to-end orchestration process

Storage architects can define workflows to follow best practices and meet organizational requirements, such as the following:

- Using required naming conventions
- Setting unique options for storage objects
- Selecting resources
- Integrating internal configuration management database (CMDB) and ticketing applications

WFA features

- Workflow design portal to build workflows

The workflow design portal includes several building blocks, such as commands, templates, finders, filters, and functions, that are used to create workflows. The designer enables you to include advanced capabilities to workflows such as automated resource selection, row repetition (looping), and approval points.

The workflow design portal also includes building blocks, such as dictionary entries, cache queries, and data source types, for caching data from external systems.

- Execution portal to execute workflows, verify status of workflow execution, and access logs
- Administration/Settings option for tasks such as setting up WFA, connecting to data sources, and configuring user credentials
- Web service interfaces to invoke workflows from external portals and data center orchestration software
- Storage Automation Store to download WFA packs

WFA license information

No license is required for using the OnCommand Workflow Automation server.

Create local users

OnCommand Workflow Automation (WFA) enables you to create and manage local WFA users with specific permissions for various roles, such as guest, operator, approver, architect, admin, and backup.

What you'll need

You must have installed WFA and logged in as an admin.

About this task

WFA enables you to create users for the following roles:

- **Guest**

This user can view the portal and the status of a workflow execution, and can be notified of a change in the status of a workflow execution.

- **Operator**

This user is allowed to preview and execute workflows for which the user is given access.

- **Approver**

This user is allowed to preview, execute, approve, and reject workflows for which the user is given access.



It is recommended to provide the email ID of the approver. If there are multiple approvers, you can provide a group email ID in the **E-mail** field.

- **Architect**

This user has full access to create workflows, but is restricted from modifying global WFA server settings.

- **Admin**

This user has complete access to the WFA server.

- **Backup**

This is the only user who can remotely generate backups of the WFA server. However, the user is restricted from all other access.

Steps

1. Click **Settings**, and under **Management** click **Users**.
2. Create a new user by clicking  on the toolbar.
3. Enter the required information in the **New User** dialog box.
4. Click **Save**.

Configuring OnCommand Workflow Automation

OnCommand Workflow Automation (WFA) enables you to configure various settings - for example, AutoSupport and notifications.

When configuring WFA, you can set up one or more of the following, as required:

- AutoSupport for sending AutoSupport messages to technical support
- Microsoft Active Directory Lightweight Directory Access Protocol (LDAP) server for LDAP authentication and authorization for WFA users
- Mail for email notifications about workflow operations and sending AutoSupport messages
- Simple Network Management Protocol (SNMP) for notifications about workflow operations
- Syslog for remote data logging

Configure authentication settings

You can configure OnCommand Workflow Automation (WFA) to use a Microsoft Active Directory (AD) Lightweight Directory Access Protocol (LDAP) server for authentication and authorization.

What you'll need

You must have configured a Microsoft AD LDAP server in your environment.

About this task

Only Microsoft AD LDAP authentication is supported for WFA. You cannot use any other LDAP authentication methods, including Microsoft AD Lightweight Directory Services (AD LDS) or Microsoft Global Catalog.



During communication, LDAP sends the user name and password in plain text. However, LDAPS (LDAP secure) communication is encrypted and secure.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Setup** click **Authentication**.
3. Select the **Enable Active Directory** check box.
4. Enter the required information in the fields:
 - a. **Optional:** If you want to use the *user@domain* format for domain users, replace `sAMAccountName` with `userPrincipalName` in the **User name attribute** field.
 - b. **Optional:** If unique values are required for your environment, edit the required fields.
 - c. Enter the AD server URI as follows:
Example: `ldap://active_directory_server_address[:port\]`

`ldap://NB-T01.example.com[:389]`

If you have enabled LDAP over SSL, you can use the following URI format:

`ldaps://active_directory_server_address[:port\]`

- d. Add a list of AD group names the required roles.



You can add a list of AD group names to the required roles in the Active Directory Groups Window.

5. Click **Save**.

Configure email notifications

You can configure OnCommand Workflow Automation (WFA) to send you email notifications about workflow operations—for example, workflow started or workflow failed.

What you'll need

You must have configured a mail host in your environment.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Setup** click **Mail**.
3. Enter the required information in the fields.
4. **Optional:** Test the mail settings by performing the following steps:
 - a. Click **Send test mail**.
 - b. In the **Test Connection** dialog box, enter the email address to which you want to send the email.
 - c. Click **Test**.
5. Click **Save**.

Configure SNMP

You can configure OnCommand Workflow Automation (WFA) to send Simple Network Management Protocol (SNMP) traps about the status of workflow operations.

About this task

WFA now supports SNMP v1 and SNMP v3 protocols. SNMP v3 provides additional security features.

The WFA .mib file provides information about the traps that are sent by the WFA server. The .mib file is located in the <WFA_install_location>\wfa\bin\wfa.mib directory on the WFA server.



The WFA server sends all the trap notifications with a generic object identifier (1.3.6.1.4.1.789.1.1.12.0).

You cannot use SNMP community strings such as `community_string@SNMP_host` for SNMP configuration.

Configure SNMP Version 1

Steps

1. Log in to WFA through a web browser as an admin user, and then access the WFA server.
2. Click **Settings**, and under **Setup** click **SNMP**.

3. Select the **Enable SNMP** check box.
4. In the **Version** drop-down list, select **Version 1**.
5. Enter an IPv4 or IPv6 address or the host name, and the port number of the management host.

WFA sends SNMP traps to the specified port number. The default port number is 162.

6. In the **Notify On** section, select one or more of the following check boxes:

- Workflow execution started
- Workflow execution completed successfully
- Workflow execution failed/partially successful
- Workflow execution waiting for approval
- Acquisition failure

7. Click **Send Test Notification** to verify the settings.

8. Click **Save**.

Configure SNMP Version 3

You can also configure OnCommand Workflow Automation (WFA) to send Simple Network Management Protocol (SNMP) Version 3 traps about the status of workflow operations.

About this task

Version 3 offers two additional security options:

- Version 3 with Authentication

Traps are sent unencrypted over the network. SNMP management applications, which are configured by the same authentication parameters as SNMP trap messages, can receive traps.

- Version 3 with Authentication and Encryption

Traps are sent encrypted over the network. To receive and decrypt these traps, you must configure SNMP management applications with the same authentication parameters and encryption key as the SNMP traps.

Steps

1. Log in to WFA through a web browser as an admin user, and then access the WFA server.
2. Click **Settings**, and under **Setup** click **SNMP**.
3. Select the **Enable SNMP** check box.
4. In the **Version** drop-down list, select one of the following options:
 - Version 3
 - Version 3 with Authentication
 - Version 3 with Authentication and Encryption
5. Select the SNMP configuration options that correspond to the specific SNMP Version 3 option you chose in Step 4.

6. Enter an IPv4 or IPv6 address or the host name, and the port number of the management host. WFA sends SNMP traps to the specified port number. The default port number is 162.
7. In the **Notify On** section, select one or more of the following check boxes:
 - Workflow planning started/failed/completed
 - Workflow execution started
 - Workflow execution completed successfully
 - Workflow execution failed/ partially successful
 - Workflow execution waiting for approval
 - Acquisition failure
8. Click **Send Test Notification** to verify the settings.
9. Click **Save**.

Configure Syslog

You can configure OnCommand Workflow Automation (WFA) to send log data to a specific Syslog server for purposes such as event logging and log information analysis.

What you'll need

You must have configured the Syslog server to accept data from the WFA server.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Maintenance** click **Syslog**.
3. Select the **Enable Syslog** check box.
4. Enter the Syslog host name and select the Syslog log level.
5. Click **Save**.

Configure AutoSupport

You can configure several AutoSupport settings such as the schedule, content of the AutoSupport messages, and the proxy server. AutoSupport sends weekly logs of the content that you selected to technical support for archiving and issue analysis.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Settings**, and under **Setup** click **AutoSupport**.
3. Ensure that the **Enable AutoSupport** box is selected.
4. Enter the required information.
5. Select one of the following from the **Content** list:

If you want to include...	Then choose this option...
Only configuration details such as users, workflows, and commands of your WFA installation	send only configuration data
WFA configuration details and data in WFA cache tables such as the scheme	send configuration and cache data (default)
WFA configuration details, data in WFA cache tables, and data in the installation directory	send configuration and cache extended data



The password of any WFA user is *not* included in the AutoSupport data.

6. **Optional:** Test that you can download an AutoSupport message:
 - a. Click **Download**.
 - b. In the dialog box that opens, select the location to save the **.7z** file.
7. **Optional:** Test the sending of an AutoSupport message to the specified destination by clicking **Send Now**.
8. Click **Save**.

Configure email notifications for data source acquisition failures

You can control the notifications that are generated in OnCommand Workflow Automation (WFA) when a data source acquisition fails. You can configure the threshold and interval factor for the notifications.

What you'll need

You must have configured email notifications in OnCommand Workflow Automation (WFA).

Steps

1. Log in to WFA through a web browser as an administrator.
2. Configure the notification threshold:
 - a. Click **Settings**, and under **Setup** click **Workflow Settings**.
 - b. Scroll down to the **Advanced** section.
 - c. In the **Acquisition notification threshold** field, enter the required value.

The default value is 2.

If the value specified is 2, then WFA waits for two consecutive data source acquisition failures before sending an email notification.

3. Click **Save**.
4. Configure the notification interval factor:
 - a. Click **Settings**, and under **Setup** click **Data Sources**.
 - b. Note the interval specified for the required data source.

The default value is 30. You can edit the data source to modify the interval time.

- c. Click **Settings**, and under **Setup** click **Workflow Settings**.
- d. Scroll down to the **Advanced** section.
- e. In the **Acquisition notification interval factor** field, enter the required value.

The default value is 6.

If the interval specified for a data source is 30 minutes and the multiplication factor is 6, then an email notification is sent when data source acquisition for the data source does not occur for 180 minutes.



If you specify 12 hours or more as the acquisition interval for a data source, WFA considers the interval factor as 1 and sends a notification after the specified acquisition interval for the data source.

5. Click **Save**.

Configure resource reservation for workflows

You can configure the resource reservation capability for OnCommand Workflow Automation (WFA) workflows to ensure that the required resources are available for successful execution of workflows.

Steps

1. Log in to WFA through a web browser as an administrator.
2. Click **Settings**, and under **Setup** click **Workflow Settings**.
3. Ensure that the **Enable reservation for commands** check box is selected.
4. In the **Reservation expiration (H)** field, enter the required value.

The default is 4.

5. Click **Save**.
6. Enable resource reservation in each required workflow:
 - a. Click **Workflow > Details** of the required workflow.

Workflow is the workflow that requires resource reservation.

- b. Select **Consider Reserved Elements**.

Configure the credentials of a target system

You can configure the credentials of a target system in OnCommand Workflow Automation (WFA) and use the credentials to connect to that specific system and execute commands.

About this task

After initial data acquisition, you must configure the credentials for the arrays on which the commands are run. PowerShell WFA controller connection works in two modes:

- With credentials

WFA tries to establish a connection using HTTPS first, and then tries using HTTP. You can also use Microsoft Active Directory LDAP authentication to connect to arrays without defining credentials in WFA. To use Active Directory LDAP, you must configure the array to perform authentication with the same Active Directory LDAP server.

- Without credentials (for storage systems operating in 7-Mode)

WFA tries to establish a connection using domain authentication. This mode uses the remote procedure call protocol, which is secured using the NTLM protocol.

- WFA checks the Secure Sockets Layer (SSL) certificate for ONTAP systems. Users might be prompted to review and accept/deny the connection to ONTAP systems if the SSL certificate is not trusted.
- You must reenter the credentials for ONTAP, NetApp Active IQ and Lightweight Directory Access Protocol (LDAP) after you restore a backup or complete an in-place upgrade.

Steps

- Log in to WFA through a web browser as an admin.
- Click **Settings**, and under **Setup** click **Credentials**.
- Click  on the toolbar.
- In the **New Credentials** dialog box, select one of the following options from the **Match** list:
 - Exact**
Credentials for a specific IP address or host name
 - Pattern**
Credentials for the entire subnet or IP range
You can use regular expression syntax for this option.
- Select the remote system type from the **Type** list.
- Enter either the host name or the IPv4 or IPv6 address of the resource, the user name, and the password.
- Test the connectivity by performing the following action:

If you selected the following match type...	Then...
Exact	Click Test .
Pattern	Save the credentials and choose one of the following: <ul style="list-style-type: none"> Select the credential and click  on the toolbar. Right-click and select Test Connectivity.

- Click **Save**.

Configure protocols for connecting to remote systems

You can configure the protocol used by OnCommand Workflow Automation (WFA) to connect to remote systems. You can configure the protocol based on your organization's security requirements and the protocol supported by the remote system.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Data Source Design > Remote System Types**.
3. Perform one of the following actions:

If you want to...	Do this...
Configure a protocol for a new remote system	<ol style="list-style-type: none">a. Click .b. In the New Remote System Type dialog box, specify the details such as name, description, and version.
Modify the protocol configuration of an existing remote system	<ol style="list-style-type: none">a. Select and double-click the remote system that you want to modify.b. Click .

4. From the **Connection Protocol** list, select one of the following:
 - HTTPS with fallback to HTTP (default)
 - HTTPS only
 - HTTP only
 - Custom
5. Specify the details for the protocol, default port, and default timeout.
6. Click **Save**.

OnCommand Workflow Automation Designer features

OnCommand Workflow Automation includes various features to help you design storage workflows.

For more information about the features, see the next topics.

How repeat row works

A workflow contains commands and command details arranged in rows. You can specify the commands in a row to be repeated for a fixed number of iterations or dynamic number of iterations based on the results of search criteria.

The command details in a row can be specified to repeat a certain number of times or when the workflow is designed. The workflow can also be designed such that the number of times the row must repeat can be

specified when the workflow is executed or scheduled for an execution. You can specify search criteria for an object and the commands in a row can be set to repeat as many times as the objects are returned by the search criteria. Rows can also be set to repeat when certain conditions are met.

Row repetition variables

You can specify variables in the variable list that can be manipulated during the row iterations. For the variables, you can specify a name, a value with which the variables are initialized, and an MVFLEX Expression Language (MVEL) expression that is evaluated after every iteration of the row repetition.

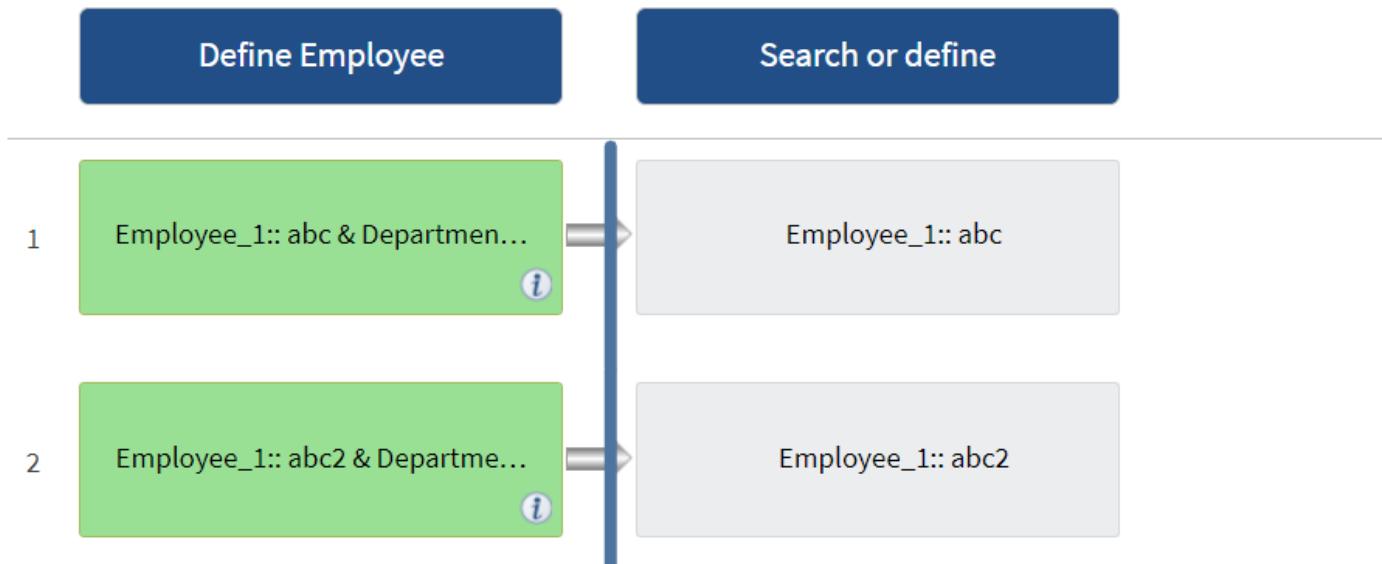
The following illustration shows the repeat row options and an example of a row repetition variable:

Variables	Name	Initial Value	Expression
	size_to_alloc	SIZE_MB	(int)size_to_allocated - getData()

Row repetition with approval points

When you have specified iterations of repeat rows for commands and included approval points, all the iterations of the commands before an approval point are executed. After you approve the approval point, the execution of all iterations of the successive commands continues until the next approval point.

The following illustration shows how the iterations of repeat rows are executed when an approval point is included in a workflow:



Repeat row examples in predefined workflows

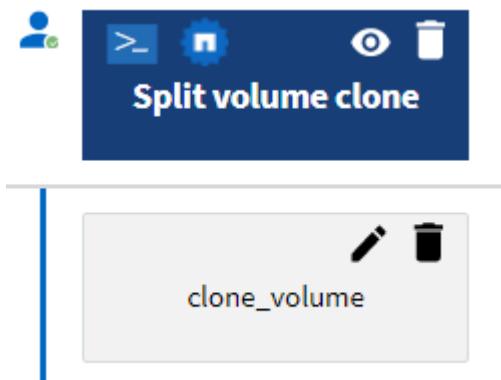
You can open the following predefined workflows in the Designer to understand how repeat rows are used:

- Create a Clustered Data ONTAP NFS Volume
- Create VMware NFS Datastore on Clustered Data ONTAP Storage
- Establish Cluster Peering
- Remove a Clustered Data ONTAP Volume

What approval points are

Approval points are check points used in a workflow to pause the workflow execution and resume it based on a user approval.

The blue vertical bar shown in the following illustration is an approval point:



You can use approval points for incremental execution of a workflow, where sections of the workflow should be executed only after a certain condition is met. For example, when the next section has to be approved or when successful execution of the first section is validated. Approval points do not handle any process between pausing and resuming of a workflow. Email and SNMP notifications are sent, as specified in the WFA configuration, and the storage operator can be asked to perform certain actions upon receiving the workflow pause notification. For example, the storage operator can send planning details to admin, approver, or operator

for approval and resume the workflow when the approval is received.

Approvals might not be required at all times. In some scenarios, the approval might be required only if a particular condition is met and the conditions can be configured when an approval point is added. For example, consider a workflow that is designed to increase the size of a volume. You can add an approval point at the beginning of the workflow for the storage operator to obtain approval from the managers when the increase in the volume size results in an 85% usage of the space in the aggregate that contains the volume. During the workflow execution and on selecting a volume that results in this condition, the execution is stopped until it is approved.

The condition that is set up for the approval point can have one of the following options:

- Without any condition
- When the variable you have specified is found
- When the variable you have specified is not found
- When the expression you have specified evaluates to true

There is no limitation on the number of approval points in a workflow. You can insert approval points before commands in a workflow and set the commands after the approval point to wait for approval before execution. Approval points provide information, such as time of change, user, and comments, allowing you to see when and why the workflow execution was paused or resumed. The approval point comments can include MVEL expressions.

Approval point examples in predefined workflows

You can open the following predefined workflows in the Designer to understand how approval points are used:

- Remove a Clustered Data ONTAP Volume
- Controller and shelf upgrade of an HA pair
- Migrate Volumes

How continue on failure works

The continue on failure feature helps you to configure a step in a workflow so that the workflow execution can continue even if the step fails. You can address the failed steps and resolve the issue that caused the step to fail by accessing the `wfa.log` file or by clicking the  icon.

A workflow that has one or more such failed steps is in the Partially Successful state after the execution is complete. You can configure a step so that the workflow execution continues even if the step fails by selecting the required option in the Advanced tab of the Parameters for <command_name> dialog box.

If a step is not configured to continue on failure, the workflow execution is aborted if the step fails.

If a step that is configured to continue on failure fails, you can set the workflow to be executed by using one of the following options:

- Abort workflow execution (default option)
- Continue execution from the next step
- Continue execution from the next row

How resource selection works

OnCommand Workflow Automation (WFA) uses search algorithms to select storage resources for workflow execution. You should understand how resource selection works in order to design workflows efficiently.

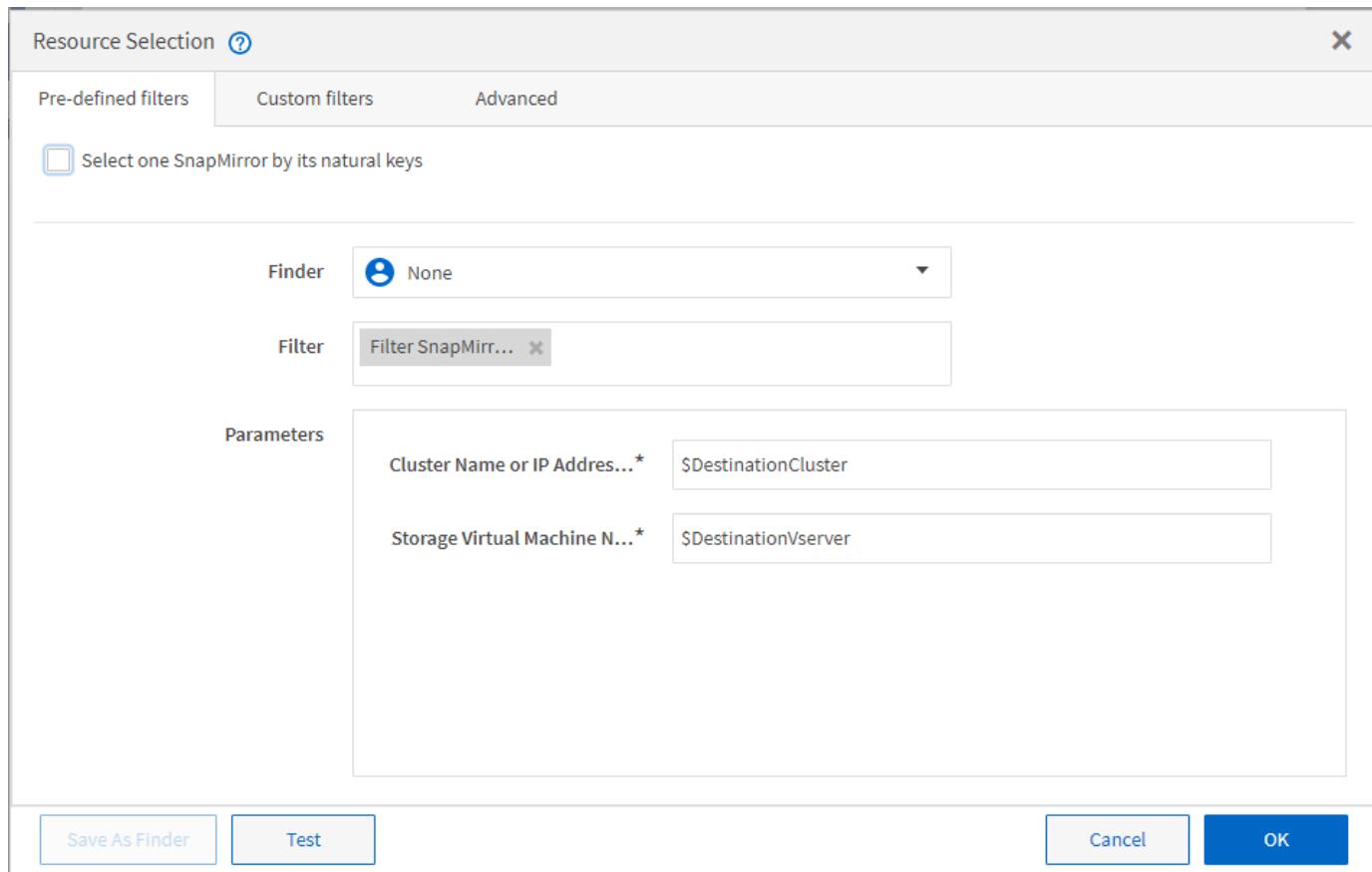
WFA selects dictionary entry resources—such as vFiler units, aggregates, and virtual machines—using search algorithms. The selected resources are then used for executing the workflow. The WFA search algorithms are part of the WFA building blocks, and include finders and filters. To locate and select the required resources, the search algorithms search through the data that is cached from different repositories, such as Active IQ Unified Manager, VMware vCenter Server, and a database. By default, a filter is available for every dictionary entry for searching a resource based on its natural keys.

You should define the resource selection criteria for each command in your workflow. In addition, you can use a finder to define the resource selection criteria in each row of your workflow. For example, when you are creating a volume that requires a specific amount of storage space, you can use the “Find aggregate by available capacity” finder in the “Create Volume” command, which selects an aggregate with a specific amount of available space and creates the volume on it.

You can define a set of filter rules for dictionary entry resources, such as vFiler units, aggregates, and virtual machines. Filter rules can contain one or more groups of rules. A rule consists of a dictionary entry attribute, an operator, and a value. The attribute can also include attributes of its references. For example, you can specify a rule for aggregates as follows: List all aggregates that have names starting with the string “aggr” and have more than 5 GB of available space. The first rule in the group is the attribute “name”, with the operator “starts-with”, and the value “aggr”. The second rule for the same group is the attribute “available_size_mb”, with the operator “>”, and the value “5000”. You can define a set of filter rules along with public filters. The Define filter rules option is disabled if you have selected a finder. The Save as Finder option is disabled if you have selected the Define filter rules check box.

In addition to the filters and finders, you can use a search or define command to search for available resources. The search or define command is the preferred option over the No-op commands. The search and define command can be used to define resources of both the certified dictionary entry type and the custom dictionary entry type. The search or define command searches for resources but does not perform any action on the resource. However, when a finder is used to search for resources, it is used in the context of a command, and the actions defined by the command are executed on the resources. The resources returned by a search or define command are used as variables for the other commands in the workflow.

The following illustration shows that a filter is used for resource selection:



Resource selection examples in predefined workflows

You can open the command details of the following predefined workflows in the Designer to understand how resource selection options are used:

- Create a Clustered Data ONTAP NFS Volume
- Establish Cluster Peering
- Remove a Clustered Data ONTAP Volume

How reservation works

OnCommand Workflow Automation resource reservation capability reserves the required resources to ensure that the resources are available for successful execution of workflows.

WFA commands can reserve the required resources and remove the reservation after the resource is available in the WFA cache database, typically after a cache acquisition. The reservation capability ensures that the reserved resources are available for the workflow until the reservation expiration period that you have configured in the WFA configuration settings.

You can use the reservation capability to exclude resources reserved by other workflows during resource selection. For example, if a workflow that has reserved 100 GB of space on an aggregate is scheduled for execution after a week, and you are executing another workflow that uses the **Create Volume** command, the workflow that is executing does not consume the space reserved by the scheduled workflow to create a new volume. In addition, the reservation capability enables workflows to be executed in parallel.

When previewing a workflow for execution, the WFA planner considers all the reserved objects, including the existing objects in the cache database. If you have enabled reservation, the effects of the scheduled workflows and the workflows that are executing in parallel, and the existence of storage elements are considered when planning the workflow.

The arrow in the following illustration shows that reservation is enabled for the workflow:

Workflow 'Abort SnapMirror relationship' 

Details	Define Workflow	User Inputs	Constants	Return Parameters	Help Content	Advanced
Workflow Name*	Abort SnapMirror relationship					
Entity Version*	1.0.0					
Categories	Data Protection					
Workflow Description	The 'Abort SnapMirror' workflow stops ongoing transfers for a					
Ready For Production	<input checked="" type="checkbox"/>					
Consider Reserved Elements	<input checked="" type="checkbox"/> 					
Enable Element Existence Validation	<input checked="" type="checkbox"/>					
Minimum Software Versions	Clustered Data ONTAP 8.2.0					

Reservation examples in predefined workflows

You can open the following predefined workflows in the Designer to understand how reservation is used:

- Clone Environment
- Create a Clustered Data ONTAP Volume
- Establish Cluster Peering
- Remove a Clustered Data ONTAP Volume

What incremental naming is

Incremental naming is an algorithm that enables you to name the attributes in a workflow based on the search results for a parameter. You can name the attributes based on an incremental value or a custom expression. The incremental naming functionality helps you implement a naming convention based on your requirement.

You can use the incremental naming functionality when designing workflows to dynamically name the objects created by the workflow. The functionality enables you to specify search criteria for an object using the resource selection feature and the value returned by the search criteria is used for the object's attribute. In addition, you can specify a value for the attribute if no object was found with the specified search criteria.

You can use one of the following options for naming the attributes:

- Providing an increment value and suffix

You can provide a value that should be used along with the value of the object found by the search criteria and increment with the number you specify. For example, if you want to create volumes with the naming convention of *filer name_unique number_environment*, you can use a finder to find the last volume by its name prefix and increment the unique number by 1, as well as add the suffix name to the volume name. If the last volume name prefix found was *vf_023_prod* and you are creating three volumes, the names for the volumes created are *vf_024_prod*, *vf_025_prod*, and *vf_026_prod*.

- Providing a custom expression

You can provide a value that should be used along with the value of the object found by the search criteria and add additional values based on the expression you enter. For example, if you want to create a volume with the naming convention of *last volume name_environment* name padded with 1, you can enter the expression *last_volume.name + ' ' + nextName("lab1")*. If the last volume name found was *_vf_023*, the name for the volume created is *vf_023_lab2*.

The following illustration shows how a custom expression can be provided to specify a naming convention:

Incremental Naming Wizard for Volume : name [?](#) X

*The Incremental Naming wizard allows you to define the value of **name** based on a search for an existing **Volume***

Search criteria for existing Volume **Volume Name : \$VolumeName, Cluster Name or IP Address : \$...**

Enter a value for **name** if no **Volume** matches the above search criteria

if **Volume** was found using above search criteria, set value for **name** by

providing a custom expression ▼

Custom expression

Cancel **Save**

What conditional execution is

Conditional execution helps you to design workflows that can execute commands when specified conditions are met.

Execution of commands in a workflow can be dynamic. You can specify a condition for the execution of each command or a row of commands in your workflow. For example, you might want the “Add volume to dataset” command to be executed only when a specific dataset is found and you do not want the workflow to fail if the dataset is not found. In this case, you can enable the “Add volume to dataset” command to search for a specific dataset and if it is not found, you can disable the command in the workflow.

Options for conditional execution of commands are available in the `Dictionary` object tab and the Advanced tab of the Parameters for commands dialog box.

You can abort a workflow or disable a specific command in the workflow. In addition, you can set a command to be executed using one of the following options:

- Without any condition
- When the variables you have specified are found
- When the variables you have specified are not found
- When the expression you have specified is true

You can also set a command to wait for a specific time interval.

Conditional execution examples in predefined workflows

You can open the command details of the following predefined workflows in the Designer to understand how conditional execution of commands are used:

- Create a basic Clustered Data ONTAP Volume
- Create a Clustered Data ONTAP NFS Volume

How return parameters work

Return parameters are parameters that are available after the planning phase of a workflow. The values returned by these parameters are useful in debugging a workflow. You should understand how return parameters work and what parameters can be used as return parameters to debug workflows.

You can designate a set of parameters, such as variable attributes, expressions, and user input values, in a workflow as return parameters. During workflow execution, the values of the designated parameters are populated in the planning phase and execution of the workflow starts. The values of these parameters are then returned the way they were calculated in that specific execution of the workflow. If you want to debug the workflow, you can refer to the values that were returned by the parameters.

You can specify the required return parameters in a workflow when you want to see what are the calculated or selected values for those parameters. For example, when using resource selection logic to select an aggregate in a workflow, you can specify `aggregate` as the return parameter so that you can see which aggregate was selected during the planning of the workflow.

Before referring to the values of the return parameters for debugging your workflow, you should confirm that

the execution of the workflow is complete. The return parameter values are set for each workflow execution. If you have added a return parameter after several executions of a workflow, the value of that parameter is available only for executions after the addition of the parameter.

Parameters that can be used as return parameters

Return parameters	Example
Variable attributes that are scalar	volume1.name, which is an attribute of the “volume name” variable
Constants	MAX_VOLUME_SIZE
User inputs	\$clusterName
MVEL expressions that involve variable attributes, constants, and user inputs	volume1.name+'-'+\$clusterName
The return parameter that a command adds during execution	The \$volumeUUID parameter is added as a return parameter when you use the following line in a PowerShell command: Add-WfaWorkflowParameter -Name "VolumeUUID" -Value "12345" -AddAsReturnParameter \$true.

Examples of return parameters in predefined workflows

If you want to understand how return parameters are specified, you can open the following predefined workflows in the Designer and review the specified return parameters:

- Create an NFS Volume in a vFiler
- Create a Qtree CIFS Share in a vFiler
- Create a Clustered Data ONTAP Volume CIFS Share

What schemes are

A scheme represents the data model for a system. A data model is a collection of dictionary entries. You can define a scheme and then define a data source type. The data source defines how the data is acquired and the scheme is populated. For example, a vc scheme acquires data about your virtual environment, such as virtual machines, hosts, and datastores.

Schemes can also be populated directly with data through workflows that are customized to solve specific problems.

Dictionary entries are associated with an existing scheme when the dictionary entries are created. Dictionary entries are also associated with cache queries, and cache queries include SQL queries.

Schemes can acquire data using either script based data source type or SQL data source type. The scripts are

defined while creating the data source type and SQL queries are defined in the cache queries.

The following schemes are included in WFA:

- **7-Mode (storage)**

Scheme to acquire data through Active IQ Unified Manager from Data ONTAP operating in 7-Mode.

- **Clustered Data ONTAP (cm_storage)**

Scheme to acquire data through Active IQ Unified Manager from clustered Data ONTAP.

- **7-Mode Performance (performance)**

Scheme to acquire performance data of Data ONTAP operating in 7-Mode through Performance Advisor.

- **Clustered Data ONTAP Performance (cm_performance)**

Scheme to acquire performance data of clustered Data ONTAP through Performance Advisor.

- **VMware vCenter (vc)**

Scheme to acquire data from VMware vCenter.

- **Playground (playground)**

Scheme that can directly populate with data.

What remote system types are

OnCommand Workflow Automation (WFA) communicates with remote system types. A remote system type specifies the type of remote systems with which WFA can communicate. You can configure remote system types in WFA. For example, Data ONTAP system can be configured as a remote system type.

A remote system type has the following attributes:

- Name
- Description
- Version
- Protocol
- Port
- Timeout

You can have a Perl script for each remote system type to validate the credentials of the remote system. You can store the credentials for the remote systems configured on WFA. You can add or edit a new custom remote system type. You can also clone an existing remote system type. You can delete a remote system type only if no systems are associated with it.

How entity versioning works

The OnCommand Workflow Automation (WFA) entities, such as commands and workflows, are versioned. You can use the version numbers to easily manage changes to the WFA entities.

Each WFA entity includes a version number in the `major.minor.revision` format—for example, 1.1.20. You can include up to three digits in each part of the version number.

Before modifying the version number of a WFA entity, you must be aware of the following rules:

- Version numbers cannot be changed from the current version to an earlier version.
- Each part of the version must be a number from 0 through 999.
- New WFA entities are versioned as 1.0.0, by default.
- An entity's version number is retained when cloning or using **Save As** to save a copy of the entity.
- Multiple versions of an entity cannot exist in a WFA installation.

When you update the version of a WFA entity, the version of its immediate parent entity is updated automatically. For example, updating the version of the **Create Volume** command updates the **Create an NFS Volume** workflow, because the **Create an NFS Volume** workflow is an immediate parent entity of the **Create Volume** command. The automatic update to versions is applied as follows:

- Modifying the major version of an entity updates the minor version of its immediate parent entities.
- Modifying the minor version of an entity updates the revision version of its immediate parent entities.
- Modifying the revision version of an entity does not update any part of the version of its immediate parent entities.

The following table lists the WFA entities and their immediate parent entities:

Entity	Immediate parent entity
Cache query	<ul style="list-style-type: none">• Data source type
Template	<ul style="list-style-type: none">• Workflow
Function	<ul style="list-style-type: none">• Workflow• Template <p> If a function contains special or mixed case characters, the version of its immediate parent entities might not be updated.</p>

Entity	Immediate parent entity
Dictionary	<ul style="list-style-type: none"> Template Filter Cache query Command Data source types which are using script method
Command	<ul style="list-style-type: none"> Workflow
Filter	<ul style="list-style-type: none"> Finder Workflow
Finder	<ul style="list-style-type: none"> Workflow
Data source type	None
Workflow	None

You can search for an entity in WFA either using the parts of the version number or the complete version number.

If you delete a parent entity, the child entities are retained and their version is not updated for the deletion.

How versioning works when importing entities

If you import entities from versions earlier than Workflow Automation 2.2, the entities are versioned as 1.0.0, by default. If the imported entity is already present in the WFA server, the existing entity is overwritten with the imported entity.

The following are the potential changes to WFA entities during an import:

- Upgrade of entities

The entities are replaced with a later version.

- Rollback of entities

The entities are replaced with an earlier version.



When you perform a rollback of an entity, the version of its immediate parent entities are updated.

- Import of new entities



You cannot selectively import entities from a .dar file.

If a later version of an entity is imported, the version of its immediate parent entities is updated.

If there are multiple child entities to the imported parent entity, only the highest degree of change (major, minor, or revision) to the child entities is applied to the parent entity. The following examples explain how this rule works:

- For an imported parent entity, if there is one child entity with a minor change and another child entity with a revision change, the minor change is applied to the parent entity.

The revision part of the parent's version is incremented.

- For an imported parent entity, if there is one child entity with a major change and another child entity with a minor change, the major change is applied to the parent entity.

The minor part of the parent's version is incremented.

Example of how the versions of imported child entities affect the parent's version

Consider the following workflow in WFA: "Create Volume and export using NFS - Custom" 1.0.0.

The existing commands included in the workflow are as follows:

- "Create Export Policy - Custom" 1.0.0
- "Create Volume - Custom" 1.0.0

The commands included in the .dar file, which is to be imported, are as follows:

- "Create Export Policy - Custom" 1.1.0
- "Create Volume - Custom" 2.0.0

When you import this .dar file, the minor version of the "Create Volume and export using NFS - Custom" workflow is incremented to 1.1.0.

How you define workflows

You must break down the goal of a workflow into the steps that should be executed by the workflow. You can then arrange the steps to complete your workflow.

A workflow is an algorithm that includes a series of steps that are required to complete an end-to-end process. The scope of the process might vary, depending on the goal of the workflow. The goal of a workflow might be defined to handle only storage operations or more complex processes such as handling networking, virtualization, IT systems, and other applications as part of a single process. OnCommand Workflow Automation (WFA) workflows are designed by storage architects and are executed by storage operators.

Defining your workflow includes breaking down the goal of your workflow into a series of steps—for example, creating an NFS volume includes the following steps:

1. Creating a volume object
2. Creating a new export policy and associating the policy with the volume

You can use a WFA command or a workflow for each step in your workflow. WFA includes predefined commands and workflows, which are based on common storage use cases. If you do not find a predefined command or workflow that can be used for a particular step, you can do one of the following:

- Choose a predefined command or workflow that closely matches the step, and then clone and modify the predefined command or workflow according to your requirements.
- Create a new command or workflow.

You can then arrange the commands or workflows in a new workflow to create the workflow that accomplishes your goal.

At the beginning of the workflow execution, WFA plans the execution and verifies that the workflow can be executed using the input to the workflow and the commands. When planning the workflow, all resource selection and user input are resolved to create an execution plan. After planning is completed, WFA executes the execution plan, which consists of a set of WFA commands with applicable parameters.

How you map command parameters

The parameters in Workflow Automation (WFA) commands are mapped to specific attributes and dictionary entry references based on certain rules. You must be aware of the rules to map command parameters when you create or edit a WFA command.

Command parameter mapping defines how command details are defined in the workflows. Mapped command parameters of a command are displayed in tabs when you are specifying the command details for commands in workflows. The tabs are named based on the group name specified in the Object Name column of the Parameters Mapping tab. The parameters that are not mapped are displayed in the Other Parameters tab when you are specifying the command details in workflows.

The rules for command parameter mapping are applicable based on the command category and how the commands are represented in the workflow editor.

The following are the command categories:

- Commands that create objects
- Commands that update objects
- Commands that remove objects
- Commands that deal with optional parent and child objects
- Commands that update associations between objects

The rules are listed below for each category:

All command categories

When mapping a command parameter, you should use the natural path based on how the command is used in workflows.

The following examples show how you can define a natural path:

- For the `ArrayIP` parameter, depending on the command, you should use the `aggregate.array.ip` attribute of the `Volume` dictionary entry and not the `array.ip` attribute.

This is important when a workflow creates a volume and then performs an additional step with the created volume by referring to it. The following are similar examples:

- `volume.aggregate.array.ip` of the `Qtree` dictionary entry

- `volume.aggregate.array.ip` of the LUN dictionary entry
- For Cluster used in commands, you should use one of the following:
 - `vserver.cluster.primary_address` of the Volume dictionary entry
 - `volume.vserver.cluster.primary_address` of the Qtree dictionary entry

Commands that create objects

This category of commands is used for one of the following:

- Finding a parent object and defining new objects
- Searching for an object and creating the object if the object does not exist

You should use the following parameter mapping rules for this category of commands:

- Map the relevant parameters of the object that is created to the object's dictionary entry.
- Map the parent object through the references of the dictionary entry that is created.
- Ensure that the relevant attribute is present in the dictionary entry when adding a new parameter.

The following are the exception scenarios for this rule:

- Some objects that are created do not have a corresponding dictionary entry and only the parent object is mapped to the relevant parent dictionary entry—for example, the **Create VIF** command—in which only an array can be mapped to array dictionary entry.
- Parameter mapping is not required

For example, the `ExecutionTimeout` parameter in the **Create or resize aggregate** command is an unmapped parameter.

The following certified commands are examples for this category:

- Create Volume
- Create LUN

Commands that update objects

This category of commands is used to find an object and update the attributes.

You should use the following parameter mapping rules for this category of commands:

- Map the objects that are updated to the dictionary entry.
- Do not map the parameters that are updated for the object.

For example, in the **Set Volume State** command, the `Volume` parameter is mapped but the new `State` is unmapped.

Commands that remove objects

This category of commands is used to find an object and delete it.

You should map the object that is deleted by the command to its dictionary entry. For example, in the **Remove Volume** command, the `Volume` to be deleted is mapped to the relevant attributes and references of the `Volume` dictionary entry.

Commands that deal with optional parent and child objects

You should use the following parameter mapping rules for this category of commands:

- Do not map any mandatory parameter of a command as a reference from an optional parameter of the command.

This rule is more relevant when a command deals with optional child objects of a specific parent object. In this case, the child and parent object should be mapped explicitly. For example, in the **Stop Deduplication Jobs** command, the command stops a running deduplication job on a specific volume when specified along with `Array` or on all volumes of the given `Array`. In this case, the `array` parameter should be mapped directly to the `array` dictionary entry and not to `Volume.Array` because `Volume` is an optional parameter in this command.

- If a parent and child relationship exists between dictionary entries at the logical level but not between the actual instances in a specific command, then those objects should be mapped separately.

For example, in the **Move Volume** command, `Volume` is moved from its current parent aggregate to a new destination aggregate. Therefore, `Volume` parameters are mapped to a `Volume` dictionary entry and the destination aggregate parameters are mapped separately to the `Aggregate` dictionary entry but not as `volume.aggregate.name`.

Commands that update associations between objects

For this category of commands, you should map both the association and the objects to relevant dictionary entries. For example, in the **Add Volume to vFiler** command, the `Volume` and `vFiler` parameters are mapped to the relevant attributes of the `Volume` and `vFiler` dictionary entries.

How user inputs are defined

The OnCommand Workflow Automation (WFA) user inputs are data input options that are available during the execution of workflows. You must define the user input parameters for your workflows to enhance the flexibility and usability of your workflows.

User inputs are shown as input fields, which can be filled out with relevant data when previewing or executing workflows. You can create a user input field when specifying the command details in a workflow by prefixing a label or variable with the dollar sign (\$). For example, `$VolumeName` creates a `Volume Name` user input field. WFA automatically populates the User Inputs tab in the Workflow <workflow name> window with the user input labels that you have created. You can also define the type of the user input and customize the input fields by modifying the user input attributes, such as type, display name, default values, and validation values.

User input type options

- **String**

You can use a regular expression for valid values—for example, `a*`.

Strings, such as `0d` and `0f`, are evaluated as numbers similar to `0d` evaluated as 0 of type double.

- **Number**

You can define a numerical range that can be selected—for example, 1 through 15.

- **Enum**

You can create enumeration values that can be selected when filling the user input field using the enum type. You can optionally lock the enum values that you have created to ensure that only the values you have created are selected for the user input.

- **Query**

You can select the query type when you want the user input to be selected from the values available in the WFA cache. For example, you can use the following query to automatically populate the user input fields with the IP address and name values from the WFA cache: `SELECT ip, name FROM storage.array`. You can optionally lock the values retrieved by a query so that only the results retrieved by the query are selected.

- **Query (multi-select)**

The query (multi-select) type, which is similar to the query type, enables the selection of multiple values during the execution of the workflow. For example, users can select multiple volumes or a volume together with its shares and exports. You can allow the users to select multiple rows, or restrict the selection to a single row. Selecting a row selects the values from all the columns of the selected row.

You can use the following functions when using the query (multi-select) type of user input:

- `getSize`
- `getValueAt`
- `getValueAt2D`
- `getValueFrom2DByRowKey`

- **Boolean**

You can use the Boolean type to display a check box in the user input dialog box. You must use the Boolean type for user inputs that have “true” and “false” as the possible values.

- **Table**

You can use the table type of user input to specify the column headers of a table that can be used to enter multiple values during the execution of the workflow. For example, a table that can be used to specify a list of node names and port names. You can also specify one of the following user input types for the column headers to validate the values that are entered during run time:

- `String`
- `Number`
- `Enum`
- `Boolean`
- `Query`

`String` is the default user input type for the column headers. You must double-click the Type column to specify a different user input type.

You can open the Create SnapMirror policy and rules workflow in the Designer to see how the user input types are used in the “SnapMirrorPolicyRule” user input.

You can use the following functions when using the table type of user input:

- getSize
- getValueAt
- getValueAt2D
- getValueFrom2DByRowKey

You can open the **Create and configure a Storage Virtual Machine with Infinite Volume** workflow in the Designer to see how the table type is used.

- **Password**

You can use the password type for user inputs that are meant for entering passwords. The password entered by the user is encrypted and displayed as a sequence of asterisk characters across the WFA application and in the log files. You can use the following functions to decrypt the password, which can then be used by the command:

- For Perl commands: `WFAUtil::getWfaInputPassword ($password)`
- For PowerShell commands: `Get-WfaInputPassword -EncryptedPassword $password`

Here, `$password` is the encrypted password that is passed by WFA to the command.

- **Dictionary**

You can add the table data for the selected dictionary entry. The dictionary entry attribute selects the attribute that is to be returned. You can select a single value or multiple values while executing the workflow. For example, you can select a single volume or multiple volumes. By default, single values are selected. You can also select Rules for filtering. A rule consists of a dictionary entry attribute, an operator, and a value. The attribute can also include attributes of its references.

For example, you can specify a rule for aggregates by listing all aggregates with name starting with the string “aggr” and have an available size greater than 5 GB. The first rule in the group is the attribute `name`, with the operator `starts-with`, and the value `aggr`. The second rule for the same group is the attribute `available_size_mb`, with the operator `>` and the value `5000`.

The following table lists the options that you can apply to the user input types:

Option	Description
Validating	<p>You can validate the user inputs type so that only valid values are entered by users:</p> <ul style="list-style-type: none"> • The string and number types of user input can be validated with the values entered during run time of the workflow. • The string type can also be validated with a regular expression. • The number type is a numeric floating-point field and can be validated using a specified numeric range.
Locking values	<p>You can lock the values of the query and enum types to prevent the user from overwriting the drop-down values and to enable the selection of only the displayed values.</p>
Marking as mandatory	<p>You can mark user inputs as mandatory so that the users must enter certain user inputs in order to continue with the execution of the workflow.</p>
Grouping	<p>You can group related user inputs and provide a name for the user input group. The groups can be expanded and collapsed in the user input dialog box. You can select a group that should be expanded by default.</p>
Applying conditions	<p>With the conditional user input capability, you can set the value of a user input based on the value that is entered for another user input. For example, in a workflow that configures the NAS protocol, you can specify the required user input for protocol as NFS to enable the “Read/Write host lists” user input.</p>

How you define constants

You can create and use constants to define a value, which can be used across a single workflow. Constants are defined at a workflow level.

The constants used in the workflow and their value are displayed in the monitoring window of the workflow during planning and execution. You must use unique names for constants.

You can use the following naming conventions to define constants:

- Uppercase for the first letter of each word, without underscores or spaces between words

All terms and abbreviations should use upper case—for example, ActualVolumeSizeInMB.

- Uppercase for all letters

You can use underscores to separate words—for example, AGGREGATE_USED_SPACE_THRESHOLD.

You can include the following as values for workflow constants:

- Numbers
- Strings
- MVEL expressions

Expressions are evaluated during the planning and execution phases of the workflows. In the expressions, you must not reference variables that are defined in a loop.

- User inputs
- Variables

How you use REST APIs

You can use REST APIs provided by Workflow Automation (WFA) to invoke workflows from external portals and the data center orchestration software. WFA supports XML and JSON content types for all REST APIs.

WFA allows external services to access various resource collections, such as workflows, users, filters, and finders, through URI paths. The external services can use HTTP methods, such as GET, PUT, POST, and DELETE, on these URIs to perform CRUD operations on the resources.

You can perform several actions through the WFA REST APIs, including the following:

- Access workflow definitions and metadata.
- Execute workflows and monitor their execution.
- View users and roles, and change passwords.
- Execute and test resource selection filters.
- Execute and test resource finders.
- Manage credentials of storage or other data center objects.
- View data sources and data source types.

REST documentation has more information about REST APIs:

https://wfa_server_ip:port/rest/docs

wfa_server_ip is the IP address of your WFA server and port is the TCP port number you have used for the WFA server during installation.

Set up a data source

You must set up a connection with a data source in OnCommand Workflow Automation (WFA) to acquire data from the data source.

What you'll need

- For Active IQ Unified Manager 6.0 and later, you must have created a database user account on the Unified Manager server.

See the *OnCommand Unified Manager Online Help* for details.

- The TCP port for incoming connections on the Unified Manager server must be open.

See the documentation on your firewall for details.

The following are the default TCP port numbers:

TCP port number	Unified Manager server version	Description
3306	6.x	MySQL database server

- For Performance Advisor, you must have created an Active IQ Unified Manager user account with a minimum role of GlobalRead.

See the *OnCommand Unified Manager Online Help* for details.

- The TCP port for incoming connections on the VMware vCenter Server must be open.

The default TCP port number is 443. See the documentation on your firewall for details.

About this task

You can add multiple Unified Manager server data sources to WFA using this procedure. However, you must not use this procedure if you want to pair Unified Manager server 6.3 and later with WFA and use the protection functionality in Unified Manager server.

For more information about pairing WFA with Unified Manager server 6.x, see the *OnCommand Unified Manager Online Help*.



While setting up a data source with WFA, you must be aware that Active IQ Unified Manager 6.0, 6.1, and 6.2 data source types are deprecated in the WFA 4.0 release, and these data source types will not be supported in future releases.

Steps

- Access WFA using a web browser.
- Click **Settings**, and under **Setup** click **Data Sources**.
- Choose the appropriate action:

To...	Do this...
Create a new data source	Click on the toolbar.
Edit a restored data source if you have upgraded WFA	Select the existing data source entry, and click on the toolbar.

If you have added a Unified Manager server data source to WFA and then upgraded the version of the Unified Manager server, WFA will not recognize the upgraded version of the Unified Manager server. You

must delete the previous version of the Unified Manager server and then add the upgraded version of the Unified Manager server to WFA.

4. In the **New Data Source** dialog box, select the required data source type, and enter a name for the data source and the host name.

Based on the selected data source type, the port, user name, password, and timeout fields might be automatically populated with the default data, if available. You can edit these entries as required.

5. Choose an appropriate action:

For...	Do this...
Active IQ Unified Manager 6.3 and later	<p>Enter the credentials of the Database User account that you created on the Unified Manager server. See <i>OnCommand Unified Manager Online Help</i> for details on creating a database user account.</p> <p> You must not provide the credentials of an Active IQ Unified Manager Database User account that was created using the command-line interface or the ocsetup tool.</p>

6. Click **Save**.
7. **Optional:** In the Data Sources table, select the data source, and click  on the toolbar.
8. Verify the status of the data acquisition process.

Configure a database user by running ocsetup on Windows

You can run the `ocsetup` file on the DataFabric Manager 5.x server to configure read-only access of the DataFabric Manager 5.x database to OnCommand Workflow Automation.

1. Download the `wfa_ocsetup.exe` file to a directory in the DataFabric Manager 5.x server from the following location:

https://WFA_Server_IP/download/wfa_ocsetup.exe.

`WFA_Server_IP` is the IP address (IPv4 or IPv6 address) of your WFA server.

If you have specified a non-default port for WFA, you must include the port number as follows:

https://wfa_server_ip:port/download/wfa_ocsetup.exe.

`port` is the TCP port number that you have used for the WFA server during installation.

If you are specifying an IPv6 address, you must enclose it with square brackets.

2. Double-click the `wfa_ocsetup.exe` file.
3. Read the information in the setup wizard and click **Next**.

4. Browse or type the OpenJDK location and click **Next**.
5. Enter a user name and password to override the default credentials.

A new database user account is created with access to the DataFabric Manager 5.x database.



If you do not create a user account, the default credentials are used. You must create a user account for security purposes.

6. Click **Next** and review the results.
7. Click **Next**, and then click **Finish** to complete the wizard.

Configure a database user by running ocsetup on Linux

You can run the `ocsetup` file on the DataFabric Manager 5.x server to configure read-only access of the DataFabric Manager 5.x database to OnCommand Workflow Automation.

Steps

1. Download the `wfa_ocsetup.sh` file to your home directory on the DataFabric Manager 5.x server using the following command in the terminal:

```
wget https://WFA_Server_IP/download/wfa_ocsetup.sh
```

`WFA_Server_IP` is the IP address (IPv4 or IPv6 address) of your WFA server.

If you have specified a non-default port for WFA, you must include the port number as follows:

```
wget https://wfa_server_ip:port/download/wfa_ocsetup.sh
```

`port` is the TCP port number that you have used for the WFA server during installation.

If you are specifying an IPv6 address, you must enclose it with square brackets.

2. Use the following command in the terminal to change the `wfa_ocsetup.sh` file to an executable:

```
`chmod +x wfa_ocsetup.sh`
```

3. Run the script by entering the following in the terminal:

```
./wfa_ocsetup.sh OpenJDK_path
```

`OpenJDK_path` is the path to OpenJDK.

Example

```
/opt/NTAPdfm/java
```

The following output is displayed in the terminal, indicating a successful setup:

```
Verifying archive integrity... All good.  
Uncompressing WFA OnCommand Setup.....  
*** Welcome to OnCommand Setup Utility for Linux ***  
    <Help information>  
*** Please override the default credentials below ***  
Override DB Username [wfa] :
```

4. Enter a user name and password to override the default credentials.

A new database user account is created with access to the DataFabric Manager 5.x database.



If you do not create a user account, the default credentials are used. You must create a user account for security purposes.

The following output is displayed in the terminal, indicating a successful setup:

```
***** Start of response from the database *****  
>>> Connecting to database  
<<< Connected  
*** Dropped existing 'wfa' user  
==> Created user 'username'  
>>> Granting access  
<<< Granted access  
***** End of response from the database *****  
***** End of Setup *****
```

Configure a database user on Active IQ Unified Manager

You must create a database user on Active IQ Unified Manager to configure read-only access of the Active IQ Unified Manager database to OnCommand Workflow Automation.

Steps

1. Log in to Active IQ Unified Manager with administrator credentials.
2. Click **Settings > Users**.
3. Click **Add a New User**.
4. Select **Database User** as the type of user.

The same user should be used in OnCommand Workflow Automation while adding Active IQ Unified Manager as a data source in OnCommand Workflow Automation.

Create workflow help content

OnCommand Workflow Automation (WFA) admins and architects who design workflows can create help content for the workflows and include it in the workflow.

What you'll need

You must be aware of how to create web pages using HTML.

About this task

The help should provide information about the workflow and the user inputs for the workflow to the storage operator who executes the workflow.

Steps

1. Create a folder with the following name: `workflow-help`.
2. Author the help content using an HTML editor or a text editor and save it as an `index.htm` file in the `workflow-help` folder.

You must not include JavaScript files as part of the help content. The following are the supported file extensions:

- ° `.jpg`
- ° `.jpeg`
- ° `.gif`
- ° `.png`
- ° `.xml`
- ° `.thmx`
- ° `.htm`
- ° `.html`
- ° `.css`

You can also include the `Thumbs.db` file, which is created by Windows.

3. Verify that the `index.htm` file and other files associated with the help content, such as images, are available in the `workflow-help` folder.
4. Create a `.zip` file of the folder and ensure that the size of the `.zip` file is not more than 2 MB.

Example: Create an NFS `volume-help.zip`

5. Edit the workflow for which you have created the help content, and then click **Setup > Help Content > Browse** to upload the `.zip` file.

Reserved words

OnCommand Workflow Automation (WFA) includes some reserved words. You must not use the reserved words in workflows for any attribute or parameters such as variable names, user input, constants, and return parameters.

The following is a list of the reserved words in WFA:

• and	• Float	• proto
• Array	• float	• return
• assert	• for	• Runtime
• Boolean	• foreach	• SecurityManager
• boolean	• function	• Short
• Byte	• if	• short
• byte	• import	• soundslike
• char	• import_static	• StrictMath
• Character	• in	• String
• CharSequence	• instanceof	• StringBuffer
• Class	• int	• StringBuilder
• ClassLoader	• Integer	• strsim
• Compiler	• is	• switch
• contains	• isdef	• System
• convertable_to	• Long	• Thread
• def	• long	• ThreadLocal
• do	• Math	• true
• Double	• new	• until
• double	• null	• var
• else	• Number	• Void
• empty	• Object	• while
• false	• or	• with

Where to find information about MVEL

You can use the *MVEL Language Guide* to learn more about MVFLEX Expression Language (MVEL).

MVEL is an expression language based on Java syntax. You can use MVEL expression syntaxes in OnCommand Workflow Automation (WFA) workflows—for example, in functions and variables.

MVEL-enabled fields in OnCommand Workflow Automation

Several fields in OnCommand Workflow Automation (WFA) are enabled for MVFLEX Expression Language (MVEL). You can use MVEL syntaxes in these fields while designing workflows.

The following table explains when and where you can use MVEL syntaxes in the WFA interface:

When...	Where to use...
Creating or editing a workflow	<ul style="list-style-type: none"> • Workflow > Details > Return Parameters > Parameter Value • Workflow > Details > Constants > Value
Creating or editing command details in a workflow	<ul style="list-style-type: none"> • Commands > Parameters for parameter_name > Enter search criteria > Resource Selection > Finder > Parameters • Commands > Parameters for parameter_name > Enter search criteria > Resource Selection > Advanced > Execute search only when the following expression evaluates to TRUE • Commands > Parameters for parameter_name > Advanced > If the following expression is TRUE • Commands > Parameters for parameter_name > Other Parameters • Commands > parameter_name > Attributes
Creating or editing a command	<ul style="list-style-type: none"> • Commands > Command Definition command_name > Properties > String Representation
Creating or editing a function	<ul style="list-style-type: none"> • Functions > Function function_name > Function definition
Creating or editing a template	<ul style="list-style-type: none"> • Template > Parameters for parameter_name > Template template_name > Attributes > Value
Editing a row	<ul style="list-style-type: none"> • Workflow > Workflow > Repeat row > Repeats > Number of times • Workflow > Workflow > Repeat row > Variables > Variables for initial_value and expression • Workflow > Workflow > Repeat row > Repeats > For every resource in a group > Resource Search Criteria > Parameters to filter • Workflow > Workflow > Add condition > If the following expression is TRUE

Examples of MVEL syntax

MVFLEX Expression Language (MVEL) syntaxes are used in several OnCommand Workflow Automation (WFA) sample workflows. You should review some sample MVEL

syntaxes to learn how to use MVEL in WFA.

The following sections provide some examples of MVEL syntaxes used in WFA.

Conditional execution

The following MVEL expression is used for conditional execution of a command when the number of volumes found is lesser than 4:

```
$NoOfVolumes < 4
```

Incremental naming

The following MVEL expression is used for incremental naming of an object:

```
last_volume.name+last_volume.state
```

This MVEL expression uses the last created volume name and the state of the last created volume for naming.

String representation in commands

The following MVEL syntax is used as a string representation:

```
DestinationCluster + ":" + DestinationVserver + "/" + DestinationVolume
```

Templates

The following MVEL syntax is used in a template:

```
calculateSnapReserveSize(calculateVolumeSizeFromDataSize((int)($fs_size*1.01),$snap_space),$snap_space)
```

This MVEL syntax is used to calculate the percentage of volume capacity to be reserved for Snapshot copies.

Command details

The following MVEL function is used in the attributes section of a parameters:

```
actualVolumeSize($VolumeSizeInGB * 1024, volume.snapshot_reserved_percent)
```

The following MVEL syntax is used in the attributes section of a parameter:

```
$VolumeName+'test001'
```

Workflow constants

The following MVEL syntax are used for constants in workflows:

```
convertNullToZero(infinite_volume.max_namespace_constituent_size_mb)
```

```
$Size_TB*1048576L
```

Return parameters

The following MVEL syntax is use to validate whether the requested size can be allocated:

```
size_remaining == 0 ? '' : throwException('Not sufficient space in
capacity_class_aggregate or data constituent of size less than 1 TB can
not be created: Total size requested='+$Size_TB+'TB'+', Size
remaining='+size_remaining/TB_TO_MB+'TB'+', Infinite volume
name='+infinite_volume.name+', Storage class='+CAPACITY_CLASS_LABEL)
```

Function definition

The following MVEL syntax is used in a function definition, which is used to convert null to zero:

```
def convertNullToZero (data)
{
    if(data == null)
    {
        return 0;
    }
    else
    {
        return data;
    }
}
```

Repetition of a specific row

The following MVEL expression uses a user input to indicate the number of times a row has to be repeated to create LUNs:

```
$NumberOfLunsToBeCreated
```

Conditional expressions for a row

The following MVEL expression uses a user input to indicate if the row will be executed:

```
$SetupSnapMirror
```

References to learning material

You should be aware of certain scripting and programming practices to create advanced Workflow Automation (WFA) workflows. You can use reference material to learn about the required options before creating WFA building blocks or workflows.

Windows PowerShell

WFA uses PowerShell scripts for workflow operations. The following table includes references to learning material for PowerShell:

Getting Started with Windows PowerShell	http://msdn.microsoft.com/en-us/library/windows/desktop/aa973757(v=vs.85).aspx
PowerShell Development—Integrated Scripting Environment (ISE)	https://docs.microsoft.com/en-us/powershell/scripting/windows-powershell/ise/introducing-the-windows-powershell-ise?view=powershell-7.2
.NET Framework Naming Guidelines	http://msdn.microsoft.com/en-us/library/xzf533w0%28v=vs.71%29.aspx
PowerShell code style	http://get-powershell.com/post/2011/04/13/Extra-Points-for-Style-when-writing-PowerShell-Code.aspx
PowerShell Try/Catch Finally	http://technet.microsoft.com/en-us/library/dd315350.aspx
PowerShell Automatic Variables	http://technet.microsoft.com/en-us/library/dd347675.aspx
PowerShell Error Reporting	https://docs.microsoft.com/en-us/powershell/scripting/developer/cmdlet/error-reporting-concepts?view=powershell-7.2
PowerShell Common Parameters	https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_commonparameters?view=powershell-7.2

Data ONTAP PowerShell toolkit

The Data ONTAP PowerShell toolkit is bundled along with WFA. You can use the PowerShell toolkit cmdlets to invoke Data ONTAP commands from a PowerShell script. For more information, see the *Data ONTAP PowerShell Toolkit Help*, which you can access from the following location:

`WFA_install_location\WFA\PoSH\Modules\DataONTAP\webhelp\index.html`.

`WFA_install_location` is the WFA installation directory, and `C:\Program Files\NetApp` is the default installation directory.

The following table includes references to information about the Data ONTAP PowerShell toolkit:

ONTAP PowerShell Toolkit Articles	https://community.netapp.com/t5/Tech-OnTap-Articles/The-Data-ONTAP-PowerShell-Toolkit/ta-p/85933
ONTAP PowerShell Toolkit NetApp Community	https://community.netapp.com/t5/forums/filteredbylabelpage/board-id/microsoft-cloud-and-virtualization-discussions/label-name/powershell%20toolkit

Perl

WFA supports Perl commands for workflow operations. When you install WFA, the required Perl and Perl modules are installed on the WFA server.

[ActivePerl User Guide](#)

You can also access the *ActivePerl User Guide* from the following location:

`WFA_install_location\WFA\Perl64\html\index.html`.

`WFA_install_location` is the WFA installation directory, and `C:\Program Files\NetApp` is the default installation directory.

WFA uses Perl scripts for workflow operations. The following table includes references to learning material for Perl:

Modern Perl: 2014	http://modernperlbooks.com/books/modern_perl_2014/index.html
Perl programming documentation	http://perldoc.perl.org/
Perl programming language	http://www.perl.org/

NetApp Manageability SDK

The required Perl modules of the NetApp Manageability SDK are bundled along with WFA. These Perl modules are required for using the Perl commands in WFA. For more information, see the NetApp Manageability SDK documentation, which you can access from the following location:

`WFA_install_location\WFA\perl\NMSDK\html`.

WFA_install_location is the WFA installation directory, and C:\Program Files\NetApp is the default installation directory.

Structured Query Language (SQL)

The SQL SELECT syntax is used in filters and to populate user inputs.

[MySQL SELECT syntax](#)

MVFLEX Expression Language (MVEL)

You can use MVEL expression syntaxes in WFA workflows—for example, in functions and variables.

For more information, see the *MVEL Language Guide*.

Regular expressions

You can use regular expression (regex) in WFA.

[ActionScript 3.0 Using regular expressions](#)

Supported workflows in ONTAP

You must be aware of the workflows that are supported when OnCommand Workflow Automation (WFA) is paired with different versions of the Unified Manager server.

The following table lists the workflows that are supported for different Unified Manager server versions.

Workflow name	Supported in Unified Manager server 6.3 and higher	Supported in Unified Manager server 5.x
Abort SnapMirror Relationship	Yes	No
Add or expand capacity storage class to Infinite Volume	Yes	No
Add or expand performance storage class to Infinite Volume	Yes	No
Break SnapMirror Relationship	Yes	No
CIFS/SMB Server Configuration	Yes	No
Controller and shelf upgrade of an HA pair	Yes	Supported only for clusters running versions earlier than Data ONTAP 8.3.

Workflow name	Supported in Unified Manager server 6.3 and higher	Supported in Unified Manager server 5.x
Create a basic Clustered Data ONTAP Volume	Yes	Yes
Create a Clustered Data ONTAP NFS Volume	Yes	Yes
Create a Clustered Data ONTAP Qtree CIFS Share	Yes	Yes
Create a Clustered Data ONTAP Volume	Yes	Yes
Create a Clustered Data ONTAP Volume CIFS Share	Yes	Yes
Create a Clustered Data ONTAP Volume with QoS Policy Group	Yes	No
Create a Cron Schedule	Yes	Yes
Create and configure a Storage Virtual Machine with Infinite Volume	Yes	No
Create and configure NAS Storage Virtual Machine	Yes	Yes Supported only for clusters running versions earlier than Data ONTAP 8.3.
Create and configure SAN Storage Virtual Machine	Yes	Yes Supported only for clusters running versions earlier than Data ONTAP 8.3.
Create Clustered Data ONTAP SnapMirror Relationship on 8.1.x	Yes	No
Create Schedules and Policies	Yes	No
Create SnapMirror Relationship	Yes	No
Create Schedules and Policies	Yes	No

Workflow name	Supported in Unified Manager server 6.3 and higher	Supported in Unified Manager server 5.x
Create SnapMirror then SnapMirror cascade	Yes	No
Create SnapMirror then SnapVault cascade	Yes	No
Create SnapVault Relationship	Yes	No
Create SnapVault then SnapMirror cascade	Yes	No
Create VMware NFS Datastore on Clustered Data ONTAP Storage	Yes	Yes
Create, map, and protect Clustered Data ONTAP LUNs with SnapMirror Relationship	Yes	No
Establish Cluster Peering	Yes	No
Expand an Infinite Volume without storage classes	Yes	No
Modify SnapMirror Relationship	Yes	No
Move a Clustered Data ONTAP Volume	Yes	Yes
Multiprotocol File Access	Yes	No
Multiprotocol Server Configuration	Yes	No
NFSv3 File Access	Yes	No
NFSv3 Server Configuration	Yes	No
Protect volume with SnapMirror Relationship	Yes	No
Protect volume with SnapVault Relationship	Yes	No
Quiesce SnapMirror Relationship	Yes	No

Workflow name	Supported in Unified Manager server 6.3 and higher	Supported in Unified Manager server 5.x
Remove a Clustered Data ONTAP Volume	Yes	Yes
Remove SnapMirror Relationship	Yes	No
Resume SnapMirror Relationship	Yes	No
Resync SnapMirror Relationship	Yes	No
Reverse Resync SnapMirror Relationship	Yes	No
Setup Data Protection for an Infinite Volume	Yes	No
SMB File Access	Yes	No
Storage Virtual Machine Peering	Yes	No
Storage Virtual Machine Root Volume Promotion	Yes	No
Storage Virtual Machine Root Volume Protection	Yes	No
Transfer SnapMirror Relationship	Yes	No

Related information

[Interoperability Matrix Tool](#)

Import OnCommand Workflow Automation content

You can import user-created OnCommand Workflow Automation (WFA) content such as workflows, finders, and commands. You can also import content that is exported from another WFA installation, content that is downloaded from the Storage Automation Store or the WFA community, as well as packs, including Data ONTAP PowerShell toolkits and Perl NMSDK toolkits.

What you'll need

- You must have access to the WFA content that you want to import.
- The content that you want to import must have been created on a system that is running the same version or an earlier version of WFA.

For example, if you are running WFA 2.2, you cannot import content that was created using WFA 3.0.

- You can import content developed on N-2 versions of WFA only into WFA 5.1.
- If the .dar file references NetApp-certified content, the NetApp-certified content packs must be imported.

The NetApp-certified content packs can be downloaded from the Storage Automation Store. You must refer to the documentation of the pack to verify that all requirements are met.

Steps

1. Log in to WFA through a web browser.
2. Click **Settings**, and under **Maintenance** click **Import Workflows**.
3. Click **Choose File** to select the .dar file that you want to import, and then click **Import**.
4. In the Import Success dialog box, click **OK**.

Related information

[NetApp community: OnCommand Workflow Automation](#)

Considerations while importing OnCommand Workflow Automation content

You must be aware of certain considerations when you import user-created content, content that is exported from another OnCommand Workflow Automation (WFA) installation, or content that is downloaded from the Storage Automation Store or the WFA community.

- WFA content is saved as a .dar file and can include the entire user-created content from another system or specific items such as workflows, finders, commands, and dictionary terms.
- When an existing category is imported from a .dar file, the imported content is merged with the existing content in the category.

For example, consider there are two workflows WF1 and WF2 in category A in the WFA server. If workflows WF3 and WF4 in category A are imported to the WFA server, category A will contain workflows WF1, WF2, WF3, and WF4 after the import.

- If the .dar file contains dictionary entries, then the cache tables corresponding to the dictionary entries are automatically updated.

If the cache tables are not updated automatically, an error message is logged in the wfa.log file.

- When importing a .dar file that has a dependency on a pack that is not present in the WFA server, WFA tries to identify whether all the dependencies on the entities are met.
 - If one or more entities are missing or if a lower version of an entity is found, the import fails and an error message is displayed.

The error message provides details of the packs that should be installed in order to meet the dependencies.

- If a higher version of an entity is found or if the certification has changed, a generic dialog box about the version mismatch is displayed, and the import is completed.

The version mismatch details are logged in a `wfa.log` file.

- Questions and support requests for the following must be directed to the WFA community:
 - Any content downloaded from the WFA community
 - Custom WFA content that you have created
 - WFA content that you have modified

Export OnCommand Workflow Automation content

You can save user-created OnCommand Workflow Automation (WFA) content as a `.dar` file and share the content with other users. The WFA content can include the entire user-created content or specific items such as workflows, finders, commands, and dictionary terms.

What you'll need

- You must have access to the WFA content that you want to export.
- If content that is to be exported contains references to certified content, the corresponding certified content packs must be available on the system when the content is imported.

These packs can be downloaded from the Storage Automation Store.

About this task

- You cannot export the following types of certified content:
 -  - NetApp-certified content
 -  - content developed by Professional Services (PS), which is available only on custom installations made by PS
 -  - packs developed by users
- All of the objects that are dependent on the exported object are also exported.

For example, exporting a workflow also exports the dependent commands, filters, and finders for the workflow.

- You can export locked objects.

The objects remain in the locked state when they are imported by other users.

Steps

1. Log in to WFA through a web browser.
2. Export the necessary content:

If you want to...	Do this...
Export all user-created content as a single .dar file	<ol style="list-style-type: none"> Click Settings, and under Maintenance click Export All Workflows. Specify a file name for the .dar file, and then click Export.
Export specific content	<ol style="list-style-type: none"> Navigate to the window from which you want to export content. Select one or more items from the window, and then click . In the Export As dialog box, specify a file name for the .dar file, and then click Export.

- In the Save As dialog box, specify the location where you want to save the .dar file, and then click **Save**.

Disable cache acquisition for dictionary entries

When it is not necessary to cache dictionary objects in OnCommand Workflow Automation (WFA), you can disable the caching of these objects. Disabling the caching of unwanted objects can reduce the time taken by WFA to complete data source acquisitions.

Steps

- Click **Data Source Design > Dictionary**.
- Select the dictionary entry for which you want to disable data acquisition.
- Click  on the toolbar, and then click **Yes**.
- If you see the error message *Cache acquisition for the dictionary entry cannot be disabled*, disable the cache acquisition for the dictionary entries listed in the error message, and then retry disabling the cache acquisition for the current dictionary object.

Create WFA workflow packs

You can create workflow packs on OnCommand Workflow Automation (WFA) for your storage automation and integration requirements.

Steps

- Log in to the **WFA** window through a web browser.
- Click **Content Management > Packs**.
- Click the **New Pack** icon.
- In the **New Pack** dialog box, enter values for the **Name**, **Author**, **Version**, and **Description** fields.
- Click **Save**.
- Verify that the new pack is created in the **Packs** window.

Delete OnCommand Workflow Automation packs

You can delete a pack from OnCommand Workflow Automation (WFA) if you no longer require it. If you delete a pack, all the entities associated with the pack are deleted.

About this task

- You cannot delete a pack if there are any dependencies on the entities that are part of the pack.

For example, if you attempt to delete a pack that includes a command that is part of a custom workflow, the delete operation will fail because the custom workflow is dependent on the pack. You can delete the pack only after you delete the custom workflow.

- Entities that are part of a pack cannot be deleted individually.

To delete an entity that is part of a pack, you must delete the pack containing that entity. If an entity is part of multiple packs, the entity will be deleted from the WFA server only when all the packs containing that entity are deleted.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Content Management > Packs**.
3. Select the pack you want to delete and click .
4. In the **Delete Pack** confirmation dialog box, click **OK**.

Add approval points

You can add an approval point as a checkpoint in a workflow to pause the workflow execution and resume it based on your approval. You can use approval points for incremental execution of a workflow, where sections of the workflow are executed only after a certain condition is met—for example, when the next section has to be approved or when successful execution of the first section is validated.

Steps

1. Log in to WFA through a web browser as an architect or an admin.
2. Click **Workflow Design > Workflows**.
3. In the **Workflows** window, double-click the workflow that you want to modify.
4. In the **Workflow <workflow name>** window, click the  icon to the left of the step for which you want to add the approval point.

You can add approval points for one or more steps.

5. In the **New Approval Point** dialog box, provide the comment and condition details.
6. Click **OK**.

Define filter rules

You can define a set of rules for filtering dictionary entry resources such as vFiler units, aggregates, and virtual machines. You can customize the rules for existing workflows and for new workflows while you are creating them.

Steps

1. Log in to WFA through a web browser as an admin.
2. Click **Workflow Design > Workflows**.
3. In the **Workflows** window, double-click the workflow that you want to modify.

The Workflow <workflow name> window is displayed.

4. Define a set of rules by choosing one of the following options:

If you want to...	Then do this...
Search for resources when the commands in a row are repeated	<ol style="list-style-type: none">a. Click a row number and select Repeat row.b. In the Row Repetitions dialog box, select the For every resource in the group option from the Repeats drop-down list.c. Select a resource type.d. Click the Enter search criteria link.
Search for resources required in command inputs	<ol style="list-style-type: none">a. Click .b. In the Parameters for <command_name> dialog box, select the by searching for an existing <dictionary object> option from the Define <dictionary object> drop-down list.c. Click the Enter search criteria link.
Search for resources referenced by variables in command inputs	<ol style="list-style-type: none">a. Click .b. In the Parameters for <command_name> dialog box, select the by filling in attributes option from the Define <dictionary object> drop-down list.c. Click  for a field marked with .
Name command inputs of String type	<ol style="list-style-type: none">a. Click .b. In the Parameters for <command_name> dialog box, select the by filling in attributes option from the Define <dictionary object> drop-down list.c. Click  for a string field.

5. In the **Resource Selection** dialog box, select the **Define filter rules** check box.

If you have selected one of the options from the Finder drop-down of Resource Selection dialog box, the Define filter rules check box is disabled. The value for the finder must be set to "None" for the Define filter rules to be enabled.

6. Enter the attribute, operator, and value for the rule.

The value must be provided within single quotation marks. The filter rules can contain one or more groups.

7. Click **OK**.

Create a scheme

You must create a scheme when data has to be cached from a new data source type or when data has to be stored by workflows directly in the database.

What you'll need

- You must have installed WFA.
- You must have administrator privileges or architect credentials for WFA.

About this task

By default, WFA supports two types of schemes: Data Source Acquisition and Other.

- Data Source Acquisition schemes: The tables in these schemes are cached by data sources defined to acquire data from remote systems.
- Other schemes: The tables in these schemes are populated with data through workflows that are customized to solve specific problems.

Steps

1. Log in to WFA through a web browser as an administrator or architect.
2. Click **Data Source Design > Schemes**.
3. Create a new scheme by clicking  on the toolbar.
4. Enter the required information such as the name, type, description, and entity version of the scheme in the **New Schemes** dialog box.
5. Click **Save**.

Edit a scheme

You can edit the display name, description, and entity version of a scheme that you created.

What you'll need

- You must have installed WFA.
- You must have administrator privileges or architect credentials for WFA.

About this task

You cannot modify predefined schemes.

Steps

1. Log in to WFA through a web browser as an administrator or architect.
2. Click **Data Source Design > Schemes**.
3. Select the scheme that you want to modify and then click  on the toolbar.
4. Modify the required information in the **Schemes <SchemeName>** dialog box.
5. Click **Save**.

Delete a scheme

You can delete a scheme that you created from OnCommand Workflow Automation (WFA) if you no longer require it.

What you'll need

- You must have installed WFA.
- You must have administrator privileges or architect credentials for WFA.

About this task

You cannot delete predefined schemes.

Steps

1. Log in to WFA through a web browser as an administrator or architect.
2. Click **Data Source Design > Schemes**.
3. Select the scheme that you want to delete and then click  on the toolbar.
4. In the **Delete Scheme** confirmation dialog box, click **Yes**.

Add a new remote system type

You can add a new remote system type to OnCommand Workflow Automation (WFA) if a predefined system type does not meet your requirement or if you want to change the configuration of a predefined system type.

What you'll need

- You must have installed WFA.
- You must have administrator privileges or architect credentials for WFA.

Steps

1. Log in to WFA through a web browser as an administrator or architect.
2. Click **Data Source Design > Remote System Types**.
3. Create a new remote system type by clicking  on the toolbar.
4. Enter the required information such as the name, type, description, and entity version of the remote system type in the **New Remote System Types** dialog box.
5. Click **Save**.

Log Viewer window

The Log Viewer window lists all of the logs available in OnCommand Workflow Automation. You can access this window by selecting **Settings**, and under **Maintenance** click **Log Viewer**.

The Log Viewer window lists the log files (in alphabetical order) and includes the file size and date when each file was generated.



This feature is not supported on Microsoft Windows Server 2003.

Backup and Restore window

The Backup and Restore window enables you to back up the OnCommand Workflow Automation (WFA) database, which contains the system configuration settings, cache information, and core packs that include PowerShell toolkits and Perl toolkits. You can also restore the saved database when you upgrade or reinstall WFA.

Backup section

You can access this window by selecting **Settings**, and under **Maintenance** click **Backup & Restore**.

- **Backup**

Opens the File Download dialog box, which enables you to save the current database as a compressed file to a specified location.

Restore section

- **Choose File**

Enables you to search for the backup database file.

- **Restore**

Enables you to reinstate the configuration settings and cache information from the backup database file to the current WFA configuration, if applicable.

The latest version of the toolkits and core packs are available after a restore operation.



If backup includes newer packs than the packs present on the system, the after restore the newer packs from backup are available.

You must stop any workflows that are running before you import a core pack.

Users Window

The Users window enables you to view, create, edit, and delete the OnCommand Workflow Automation (WFA) users. You can access this window by selecting **Settings**,

and under **Management** click **Users**.



Depending on your role and account privileges, this window might not be displayed.

Users table

The Users window displays existing users. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

- enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking clears and resets the filtering selections.
- on each column header enables you to filter based on the content of the columns. Clicking in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows (for ascending and for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The following table lists the supported operators for alphabetic and numeric columns in the Users table, available from the **Search** filter text box.

Alphabetic	Numeric
<ul style="list-style-type: none">• Contains• Does not contain• Starts with• Ends with• Equals• Does not equal• Reset	<ul style="list-style-type: none">• Equals• Does not equal• Less than• Greater than• Less than or equal to• Greater than or equal to• Between• Reset

The Users table contains the following columns:

- **Username**

Displays the user name for the account.

- **Role**

Displays the assigned role for the user. Roles can include any of the following:

- **Guest**

This user can only view the status of a workflow execution or can be notified of a change in the status of a workflow execution.

- **Operator**

This user is allowed to preview and execute workflows for which the user is provided access.

- **Approver**

This user is allowed to preview, execute, approve, and reject workflows for which the user is provided access.



You must provide the email ID of the approver and the status of the workflow that is to be notified to the approver. If there are multiple approvers, you can provide a group email ID in the **E-mail** field.

- **Architect**

This user has full access to create workflows, but is restricted from modifying global WFA server settings.

- **Admin**

This user has complete access to the WFA server.



You must configure at least one admin user.

- **Backup**

This is the only user who can remotely generate backups of the WFA server; however, this user is restricted from all other access.

- **Categories**

Displays the workflow category assigned to an operator, which provides workflow authorization and privileges to the designated operator.

You can set this authorization setting by clicking **Content Management > Categories**.

- **E-mail**

Displays the email address for the user. You can use this email for notification about the workflow status.

- **Notifications Enabled**

Indicates whether the user can receive email notifications (true or false) about the status of workflow executions that were triggered by that user.

- **LDAP**

Indicates whether the user is provisioned through the External Active Directory server accessed by LDAP (true or false).

- **Active Directory Group**

Indicates whether the user is a part of LDAP or Active Directory groups.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New User dialog box, which enables you to add a new user account.

-  **(Edit)**

Opens the Edit User dialog box, which enables you to edit the selected user account.

-  **(Delete)**

Opens the Delete User confirmation dialog box, which enables you to delete the selected user account.

New User dialog box

The New User dialog box enables you to create a new user account.

- **Username**

Specifies the user name.

- **Role**

Enables you to select one of the following user roles:

- Guest: This user can only view the status of a workflow execution or can be notified about a change in the status of a workflow execution.
- Operator: This user is allowed to preview and execute workflows for which the user is provided access.
- Architect: This user has full access to create workflows, but is restricted from modifying global WFA server settings.
- Admin: This user has complete access to the WFA server.
- Backup: This user is the only user who can remotely generate backups of the WFA server; however, the user is restricted from all other access.

- **E-mail**

Enables you to specify the user's email address to which notifications are sent if the Notify On option is selected.

- **Password**

Specifies the password for the user.

- **Confirm**

Specifies the password again.

- **Notify On**

Enables you to select when the account user should be notified by email. The workflow execution status notifications are specific to the workflows executed by the user. You can select any combination of the following options:

- Workflow execution started: Notifies the user when the workflow execution begins.
- Workflow execution failed/partially successful: Notifies the user if the workflow execution fails or if workflow has been successfully executed even if one or more steps have failed.

The execution is completed because the failed steps have been configured such that the workflow execution continues even when the step has failed.

- Workflow execution completed successfully: Notifies the user when the workflow execution is completed successfully.
- Workflow execution waiting for approval: Notifies the user if the workflow execution is waiting for approval from either an Operator or Architect user, depending on the WFA configuration setting.
- Acquisition failure: Notifies the user when data acquisition for data sources fails.

This option is enabled only for Admin and Architect users.

Command buttons

- **Save**

Saves the configuration settings and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Edit User dialog box

The Edit User dialog box enables you to view and modify user account settings.

Depending on your account privileges and role, you might not have access to the Users window. However, you can edit the settings for your account by selecting **Settings**, and under **Management** click **Account Settings**.



The fields that are available to you for editing depend on your assigned role and account privileges.

- **Name**

Displays the user name for the user account.

- **Role**

Displays the assigned role for the user account.

- **E-mail**

Specifies the email for the user account to which notifications are sent if a Notify On option is selected.

- **Change password**

Enables you to change the current password for the user account. When this check box is selected, the following fields are required:

- **New Password**--Specifies the new password.
- **Confirm**--Specifies the new password again.

- **Notify On**

Enables you to select when the account user should be notified by email during the workflow execution. When this check box is selected, you can select any combination of the following options:

- Workflow execution started: Notifies the user when the workflow execution begins.
- Workflow execution failed/partially successful: Notifies the user if the workflow execution fails or if workflow has been successfully executed even if one or more steps have failed.

The execution is completed because the failed steps have been configured such that the workflow execution continues even when the step has failed.

- Workflow execution completed successfully: Notifies the user when the workflow execution is completed successfully.
- Workflow execution waiting for approval: Notifies the user if the workflow execution is waiting for approval from either an Operator or Architect user, depending on the WFA configuration setting.
- Acquisition failure: Notifies the user when data acquisition for data sources fails.

This option is enabled only for Admin and Architect users.

Command buttons

- **Save**

Saves the configuration settings and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Preferences window

The Preferences window enables you to view and modify the refresh and logging options for all of the clients associated with the specific Workflow Automation (WFA) server. You can access this window by selecting **Settings**, and under **Setup** click **Preferences**.

- **Enable Auto Refresh**

Enables you to choose the automatic refresh of the client, which is triggered by the server. This option is selected by default. When selected, the following WFA tables are refreshed automatically:

- Execution Status
- Data Sources
- Reservations

If not selected, the tables are not updated until you click **Refresh**. This check box is selected by default.

- **Refresh Interval (seconds)**

Enables you to select the refresh interval. The default value is 3 seconds.

- **Max number of rows to display**

Enables you to click the value in the **Max Size** column and modify the number of rows that the server retrieves and displays to the user per list type.

For example, the **Workflow Execution** setting displays the number of past history executions to the user. The following are the default values:

- Workflow Execution: 100
- Acquisition History: 30
- Reservations: 100
- Add inventory rows: 1000

Enables you to specify the maximum number of results that should be obtained by inventory. The default value is 1000.

Command button

- **Save**

Saves the configuration settings.

Active Directory Groups window

The Active Directory Groups window enables you to view, create, edit, and delete the OnCommand Workflow Automation (WFA) Active Directory groups. You can access this window by selecting **Settings**, and under **Management** click **Active Directory Groups**.



Depending on your role and account privileges, this window might not be displayed.

- Active Directory groups table
- Toolbar

Active Directory Groups table

The Active Directory Groups window displays existing Active Directory groups. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

- enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking clears and resets the filtering selections.
- on each column header enables you to filter based on the content of the columns. Clicking in a column allows you to filter on a specific item from the drop-down list or on all available items.

- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows (▲ for ascending and ▼ for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The following table lists the supported operators for alphabetic and numeric columns in the Active Directory Groups table, available from the **Search** filter text box.

Alphabetic	Numeric
<ul style="list-style-type: none"> Contains Does not contain Starts with Ends with Equals Does not equal Reset 	<ul style="list-style-type: none"> Equals Does not equal Less than Greater than Less than or equal to Greater than or equal to Between Reset

The Active Directory Groups table contains the following columns:

- Group Name**

Displays the group name for the Active Directory.

- Role**

Displays the assigned role for the group. Roles can include any of the following:

- Guest**

This group can only view the status of a workflow execution or be notified of a change in the status of a workflow execution.

- Operator**

This group is allowed to preview and execute workflows for which the group is provided access.

- Approver**

This group is allowed to preview, execute, approve, and reject workflows for which the group is provided access.



You must provide the email ID of the approver and the status of the workflow that is to be notified to the approver. If there are multiple approvers, you can provide a group email ID in the **E-mail** field.

- **Architect**

This group has full access to create workflows, but is restricted from modifying global WFA server settings.

- **Admin**

This group has complete access to the WFA server.

- **Categories**

Displays the workflow category assigned to an operator, which provides workflow authorization and privileges to the designated operator.

You can set this authorization setting by clicking **Content Management > Categories**.

- **E-mail**

Displays the email address for the group. You can use this email for notification about the workflow status.

- **Notifications Enabled**

Indicates whether the group can receive email notifications (true or false) about the status of workflow executions that were triggered by that group.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Active Directory Group dialog box, which enables you to add a new Active Directory group.



Before adding the Active Directory group, you need to enable or configure the Active Directory group.

-  **(Edit)**

Opens the Edit Active Directory Group dialog box, which enables you to edit the selected Active Directory group.

-  **(Delete)**

Opens the Delete Active Directory Group dialog box confirmation dialog box, which enables you to delete the selected Active Directory group.

New Active Directory Group dialog box

The New Active Directory Group dialog box enables you to create a new OnCommand Workflow Automation Active Directory group.

- **Group Name**

Specifies the group name.

- **Role**

Enables you to select one of the following group roles:

- **Guest**

This group can only view the status of a workflow execution or be notified about a change in the status of a workflow execution.

- **Operator**

This group is allowed to preview and execute workflows for which the group is provided access.

- **Approver**

This group is allowed to preview, execute, approve, and reject workflows for which the group is provided access.



You must provide the email ID of the approver and the status of the workflow that is to be notified to the approver. If there are multiple approvers, you can provide a group email ID in the **E-mail** field.

- **Architect**

This group has full access to create workflows, but is restricted from modifying global WFA server settings.

- **Admin**

This group has complete access to the WFA server.



You must configure at least one admin group.

- **E-mail**

Enables you to specify the group's email address to which notifications are sent if the **Notify On** option is selected.

- **Notify On**

Enables you to select when the active directory group should be notified by email. The workflow execution status notifications are specific to the workflows executed by the group. You can select any combination of the following options:

- Workflow execution started: Notifies the group when the workflow execution begins.
- Workflow execution failed/partially successful: Notifies the group if the workflow execution fails or if the workflow was successfully executed even though one or more steps failed.

The execution is completed because the failed steps were configured to allow the workflow execution to continue even when the step fails.

- Workflow execution completed successfully: Notifies the group when the workflow execution is

completed successfully.

- Workflow execution waiting for approval: Notifies the group if the workflow execution is waiting for approval from the Approver, Architect, or Operator group, depending on the WFA configuration setting.

Command buttons

- **Save**

Saves the configuration settings and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Edit Active Directory Group dialog box

The Edit Active Directory Group dialog box enables you to view and modify the OnCommand Workflow Automation Active Directory group settings.

Depending on your Active Directory group privileges and role, you might not have access to the Active Directory Groups window. However, you can edit the settings for your Active Directory group by selecting **Settings**, and under **Management** click **Active Directory Groups**.



The fields that are available to you for editing depend on your Active Directory group privileges.

- **Group Name**

Displays the group name for the Active Directory group.

- **Role**

Displays the assigned role for the user account.

Displays one of the following group roles assigned for the Active Directory group:

- **Guest**

This group can only view the status of a workflow execution or be notified about a change in the status of a workflow execution.

- **Operator**

This group is allowed to preview and execute the workflows for which the group is provided access.

- **Approver**

This group is allowed to preview, execute, approve, and reject the workflows for which the group is provided access.



You must provide the email ID of the approver and the status of the workflow that is to be notified to the approver. If there are multiple approvers, you can provide a group email ID in the **E-mail** field.

- **Architect**

This group has full access to create workflows, but is restricted from modifying global WFA server settings.

- **Admin**

This group has complete access to the WFA server.



You must configure at least one admin group.

- **E-mail**

Specifies the email for the Active Directory group to which notifications are sent if the **Notify On** option is selected.

- **Notify On**

Enables you to select when the active directory group should be notified by email during the workflow execution triggered by that active directory group. You can select any combination of the following options:

- Workflow execution started: Notifies the user when the workflow execution begins.
- Workflow execution failed/partially successful: Notifies the user if the workflow execution fails or if the workflow was successfully executed even though one or more steps failed.

The execution is completed because the failed steps were configured to allow the workflow execution to continue even when the step fails.

- Workflow execution completed successfully: Notifies the user when the workflow execution is completed successfully.
- Workflow execution waiting for approval: Notifies the user if the workflow execution is waiting for approval from either an Operator or Architect user, depending on the WFA configuration setting.

Command buttons

- **Save**

Saves the configuration settings and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Approval Portal window

The Approval Portal window enables you to approve or reject a workflow.

You can log in to the Approval Portal window from the link provided in the email notification for the approval of the workflow.

- **Return to WFA**

Takes you back to the WFA application from the Approval Portal window.

Approval Portal

- **Enter Comment**

Enables you to enter your comment for the approval or rejection of the workflow.

Command buttons

- **Approve & Resume**

Enables you to approve the workflow, which resumes the workflow.

- **Reject & Abort**

Enables you to reject the workflow, which aborts the workflow.

Data Sources window

The Data Sources window enables you to view, edit, and delete existing data sources, create new data sources, and acquire a data source. You can access this window by selecting **Settings**, and under **Setup** click **Data Sources**.

A data source is a read-only data structure that contains information about where the data in a specific database originates from. Data sources must be defined before environments can be polled. For example, a data source can be either an Active IQ Unified Manager database that contains information about a storage environment or a VMware database with information about a data center.

- Data Sources table
- History table
- Toolbar

Data Sources table

The Data Sources table lists the existing data sources. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The following table lists the supported operators for alphabetic and numeric columns in the Data Sources table, available from the **Search** filter text box.

Alphabetic	Numeric
<ul style="list-style-type: none"> Contains Does not contain Starts with Ends with Equals Does not equal Reset 	<ul style="list-style-type: none"> Equals Does not equal Less than Greater than Less than or equal to Greater than or equal to Between Reset

The Data Sources table contains the following columns:

- Name**

Displays the name of the data source.

- Data Source Type**

Displays the data source type.

- Host Name**

Displays the host name or IP address of the data source.

- Scheme**

Displays the caching scheme associated with the data source. For example, a VM caching scheme contains data that is relevant to your virtual environment such as virtual machines, hosts, and datastores. The relevant scheme information is obtained from the data source.

- Interval (minutes)**

Displays the interval (in minutes) between two consecutive acquisitions of a data source.

- Start Time**

Displays the date and time when the data acquisition process was started.

Click the calendar icon in the **Search** filter text box to search for data sources with a specific start date.

- Duration (seconds)**

Displays the time (in seconds) that it takes to process the last data acquisition from the data source.

- Status**

Displays the status of the current data acquisition process (if any) or the last data acquisition process. The status options include the following:

- All

- Aborting
- Canceled
- Completed
- Failed
- NoAcquisition
- Obsolete
- Pending
- Executing
- Scheduled

- **Message**

Displays an error message when the data acquisition process encounters an error and stops.

History table

The History table displays in the header the name of the data source that was selected in the Data Source table, and lists the details of each data acquisition process for the selected data source. The list of processes is updated dynamically, as the data acquisition processes occur. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The following table lists the supported operators for alphabetic and numeric columns in the History table, available from the **Search** filter text box.

Alphabetic	Numeric
<ul style="list-style-type: none"> Contains Does not contain Starts with Ends with Equals Does not equal Reset 	<ul style="list-style-type: none"> Equals Does not equal Less than Greater than Less than or equal to Greater than or equal to Between Reset

The History table contains the following columns:

- **ID**

Displays the identification number of the data acquisition process.

The identification number is unique and is assigned by the server when it starts the data acquisition process.

- **Start Time**

Displays the date and time when the data acquisition process was started.

Click the calendar icon in the **Search** filter text box to search for data acquisition processes started on a specific date.

- **Duration (seconds)**

Displays the length of time (in seconds) of the last acquisition process from the data source.

- **Planned Acquisition**

Displays the scheduled date and time for the data acquisition process.

Click the calendar icon in the **Search** filter text box to search for data acquisitions scheduled for a specific date.

- **Scheduling Type**

Displays the type of schedule. The schedule types include the following:

- All
- Immediate
- Recurring
- Unknown

- **Status**

Displays the status of the current data acquisition process (if any) or the last data acquisition process. The

status options include the following:

- All
- Aborting
- Canceled
- Completed
- Failed
- Obsolete
- Pending
- Executing
- Scheduled
- NoAcquisition

- **Message**

Displays a message about the error that was encountered during the data acquisition process, when the process stopped and could not continue.

Toolbar

The toolbar is located above the column headers of the Data Sources table. You can use the icons in the toolbar to perform various actions. You can also perform these actions by using the right-click menu in the window.

-  **(New)**

Opens the New Data Source dialog box, which enables you to add a new data source.

-  **(Edit)**

Opens the Edit Data Source dialog box, which enables you to edit the selected data source.

-  **(Delete)**

Opens the Delete Data Source confirmation dialog box, which enables you to delete the selected data source.

-  **(Acquire Now)**

Initiates the acquisition process for the selected data source.

-  **(Reset Scheme)**

Opens the Reset Scheme confirmation dialog box. This dialog box enables you to reset the cache storage for the selected scheme. The cache is reset during the next data acquisition process.



The reset process deletes all the cached data, including all the tables. The entire cache is built from start during the next data acquisition process.

New Data Source dialog box

The New Data Source dialog box enables you to add a new data source.

Data Source Configuration properties

- **Name**

Enables you to specify a name for the data source.

- **Data source type**

Enables you to select the data source type—for example, Active IQ Unified Manager - 6.0 (MYSQL).

When you select a data source type, the port, scheme, user name, password, interval (minutes), and timeout (sec) fields are populated with data.



If you want to execute OnCommand Workflow Automation (WFA) data protection workflows from an Active IQ Unified Manager server, you must set up WFA in the Active IQ Unified Manager server.

For more information, see the *OnCommand Unified Manager Online Help*.

- **Host name**

Enables you to specify the host name or IP address (IPv4 or IPv6) for the data source.



The Active IQ Unified Manager data source type does not support IPv6 addresses.

- **Port**

Displays the associated port number, if any, for the selected data source type.

You can change the default port and specify a different port for the data source.

- **Scheme**

Displays the scheme associated with the selected data source type—for example *cm_storage* for Active IQ Unified Manager - 6.0 (MYSQL).

This property cannot be modified.

- **User name**

Displays the associated user name, if any, for the selected data source type.

You must specify the appropriate user name for this data source to override the default user name:

- For Active IQ Unified Manager 6.0 and later, you must enter the user name of the database user account that you created on the Active IQ Unified Manager server.
- For Performance Advisor, you must enter the user name of an Active IQ Unified Manager user with a minimum role of GlobalRead.

- **Password**

Enables you to specify a password for the selected data source type.

You must specify a password to override the default password for the user name of this data source:

- For Active IQ Unified Manager 6.0 and later, you must enter the password of the database user account that you created on the Active IQ Unified Manager server.
- For Performance Advisor, you must enter the password of an Active IQ Unified Manager user with a minimum role of GlobalRead.

- **Database**

- **Interval (minutes)**

Enables you to click the value and select an interval (in minutes) using the up and down arrows.



If the interval is set to zero (0), automatic sampling is disabled.

The default interval for data sampling is:

- Script-based data source type: 1440 (performance and vc)
- SQL-based data source type: 30 (cm_storage, storage, and cm_performance)

- **See Setup Instruction Guide**

Opens the Setup Instruction Guide dialog box, which enables you to specify the instructions for setting up the data source.

Command buttons

- **Save**

Saves the configurations settings and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Edit Data Source dialog box

The Edit Data Source dialog box enables you to modify an existing data source.

Data Source Configuration properties

- **Name**

Enables you to modify the name of the data source.

- **Data source type**

Displays the data source type for the data source. You cannot modify this field.

- **Host name**

Enables you to specify the host name or IP address (IPv4 or IPv6) for the data source.



The Active IQ Unified Manager data source type does not support IPv6 addresses.

- **Port**

Displays the associated port number, if any, for the selected data source type. You can change the default port number for the data source.

- **Scheme**

Displays the scheme associated with the selected data source type—for example *cm_storage* for Unified Manager - 6.0 (MYSQL).

You cannot modify this property.

- **User name**

Displays the associated user name, if any, for the selected data source type.

You must specify the appropriate user name for this data source to override the default user name:

- For Unified Manager 6.0 and later, you must enter the user name of the database user account that you created on the Unified Manager server.
- For Performance Advisor, you must enter the user name of an Active IQ Unified Manager user with a minimum role of GlobalRead.

- **Password**

Enables you to specify a password for the selected data source type.

You must specify a password to override the default password for the user name of this data source:

- For Unified Manager 6.0 and later, you must enter the password of the database user account that you created on the server.
- For Performance Advisor, you must enter the password of an Unified Manager user with a minimum role of GlobalRead.

- **Database**

Displays the associated database name, if any, for the selected data source type.

- **Interval (minutes)**

Enables you to click the value and select an interval (in minutes) using the up and down arrows.



If the interval is set to zero (0), automatic sampling is disabled.

The default interval for data sampling is:

- Script based data source type: 1440 (performance and vc)
- SQL based data source type: 30 (*cm_storage*, *storage*, and *cm_performance*)

- **See Setup Instruction Guide**

Opens the Setup Instruction Guide dialog box, which outlines the steps for setting up the data source.

Command buttons

- **Reset Scheme**

Opens the Reset Scheme confirmation dialog box. This dialog box enables you to reset the cache storage for the selected scheme, which occurs during the next data acquisition.



The reset process deletes all the cached data, including all the tables. The entire cache is built from start during the next data acquisition.

- **Save**

Saves the configurations settings and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Credentials window

The Credentials window enables you to view, create, edit, and delete credentials. You can access this window by selecting **Settings > Setup > Credentials**.

Credentials are information (such as IP address or host name, user name and password) that is stored on a target system (server or controller) and is used to connect to that specific system and to execute commands.

- Credentials table
- Toolbar

Credentials table

The Credentials window displays existing credentials in a table format. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

- enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking clears and resets the filtering selections.
- on each column header enables you to filter based on the content of the columns. Clicking in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows (for ascending and for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The following table lists the supported operators for alphabetic and numeric columns in the Credentials table, available from the **Search** filter text box.

Alphabetic	Numeric
<ul style="list-style-type: none"> Contains Does not contain Starts with Ends with Equals Does not equal Reset 	<ul style="list-style-type: none"> Equals Does not equal Less than Greater than Less than or equal to Greater than or equal to Between Reset

The Credentials table contains the following columns:

- Type**

Displays the type of credential.

- Name/IP Address**

Displays the IP address for the credential.

- Host Name**

Displays the host name for the credential.

- Login**

Displays the user name associated with the credential.

- Match**

Displays the match type for the credential. Match types are as follows:

- Exact: defines credentials for a specific IP address or host name
- Pattern: defines credentials for the entire subnet or IP range, or host name range

Pattern is a regular expression that is used for matching a string—for example, 10.10.10. matches any IP address in the range 10.10.10.0 through 10.10.10.255, and host matches host names starting with the string host.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(Test Connectivity)**

Opens the Test Connectivity dialog box, which enables you to test the credentials.

-  **(New)**

Opens the New Credentials dialog box, which enables you to create a new credential.

-  **(Edit)**

Opens the Edit Credentials dialog box, which enables you to edit the selected credential.

-  **(Delete)**

Opens the Delete Credentials confirmation dialog box, which enables you to delete the selected credential.

Test Connectivity dialog box

The Test Connectivity dialog box enables you to test the connection to the system associated with a credential. The test connectivity operation triggers the search path and first looks for an exact match, then for a pattern match, and finally for LDAP service authentication.

The Internet Control Message Protocol (ICMP) is used to check if the system is running before testing for connectivity.

Test Connectivity parameters

- **Type**

Enables you to select the type of credential from the drop-down list.

- **Name/IP**

Enables you to specify the host name or IP address for the credential.

Command buttons

- **Test**

Opens a connection attempt to the host name or IP address.

- **Close**

Cancels changes, if any, and closes the dialog box.

New Credentials dialog box

The New Credentials dialog box enables you to create a new credential.

Credential parameters

You can set the following parameters for the newly created credential:

- **Match**

Enables you to select the match type of the credential, which defines the credential for either a specific IP address or host name, or an entire subnet or IP range. You can select one of the following options:

- Exact: defines credentials for a specific IP or host name
- Pattern: defines credentials for the host name range or the entire subnet or IP range

Pattern is a regular expression that is used for matching a string. For example, 10.10.10. matches any IP in the range 10.10.10.0 through 10.10.10.255, and host matches all host names starting with host.

- **Type**

Enables you to select the host type from the drop-down list.



VMware vCenter is not supported for Linux.

- **Name/IP**

Enables you to specify the host name or IP address for the credential.

- **User Name**

Enables you to specify a user name for the credential.

- **Password**

Enables you to specify a password for the user name that you created for the credential.

- **Overwrite Default Values**

Enables you to overwrite the default values of the protocol, port, and timeout that are associated with the selected remote system type for the credential.

By default, this check box is not selected. You must select the check box if you want to overwrite the default values.

Command buttons

- **Test**

Enables you to test the connection to the host or IP address by logging in using the required credentials.

- **Save**

Saves the configuration settings and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Edit Credentials dialog box

The Edit Credentials dialog box enables you to modify an existing credential.

Credential parameters

- **Match**

Enables you to select the match type of the credential, which defines the credential for either a specific IP address or host name, or an entire subnet or IP range. Available options include the following:

- Exact: defines a specific IP address or host name
- Pattern: defines an entire subnet or IP range

- **Type**

Enables you to select the type of credential from the drop-down list.

VMware vCenter is not supported for Linux.

- **Name/IP**

Enables you to specify the host name or IP address for the credential.

- **User Name**

Enables you to specify the user name for the credential.

- **Password**

Enables you to specify the password for the user name.

- **Overwrite Default Values**

Enables you to overwrite the default values of the protocol, port, and timeout that are associated with the selected remote system type for the credential.

By default, this check box is not selected. You must select the check box if you want to overwrite the default values.

Protocol table

The protocol table displays the port numbers and timeout limits (in seconds) for protocols associated with the selected remote system for the credential.

Command buttons

- **Test**

Enables you to test the connection to the host name or IP address by logging in using the required credentials.

- **Save**

Saves the configuration settings and closes the dialog box.

- **Cancel**

Cancels the changes, if any, and closes the dialog box.

About dialog box

The About dialog box displays details about the current version of the installed WFA application.

WFA version information

- **Version**

Displays the current version number of the installed WFA application.

The version number has the following format: A.B.C.

A.B.C reflects the Major.Minor.Maintenance release numbers.

For example: 2.0.0.

- **System ID**

Identifies the WFA system installation for tracking and AutoSupport purposes.

This is a unique system-generated identifier.

Workflows Window

The Workflows window displays the workflows that are marked ready for production. You can edit or execute the workflows for which you have authorization. You can access this window by clicking the Workflows tab.

Category pane

The category pane enables you to search for workflows by categories. A workflow that is not assigned to any category is listed under “No Category”. To the right of the category heading, the total number of workflows within that category is displayed. This number includes the workflows that are marked ready for production. For example, “Data Protection (7)” indicates that seven workflows are ready for production in the “Data Protection” category.

Schemes

The Schemes menu enables you to select the schemes.

The schemes that are displayed are based on the selection made by the administrator in the Show content for schemes option under **Settings > Setup > Workflow Settings**. You can select the schemes that you want to be displayed by selecting or clearing the check boxes. This selection is valid only for the current session.

Workflow pane

Workflows that are marked ready for production are displayed as gray boxes on the Workflow pane. The workflow boxes include the name of the workflow and an icon to indicate that the mode is supported by the workflow. You can execute a workflow by clicking the workflow box.

If you are an administrator or an architect, you can edit a workflow by clicking the  icon. The  icon provides more information about the workflow.

Executions window

The Executions window displays the status of the execution process for each workflow that has been submitted for execution. The window enables you to view details of the execution process and control the execution of a workflow. You can access this window by selecting **Execution > Executions**.

- Workflow table
- Toolbar

Workflow table

The Workflow table lists the workflows submitted for execution. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The following table outlines the supported operators for alphabetic and numeric columns in the Workflow table, available in the **Search** filter text box.

Alphabetic	Numeric
<ul style="list-style-type: none">• Contains• Does not contain• Starts with• Ends with• Equals• Does not equal• Reset	<ul style="list-style-type: none">• Equals• Does not equal• Less than• Greater than• Less than or equal to• Greater than or equal to• Between• Reset

The Workflow table contains the following columns:

- **Job #**

Displays the identification number of the job.

The job identification number is unique and is assigned by the server when it starts the job.

- **Name**

Displays the name of the workflow.

- **Start Time**

Displays the date and time when the workflow was started.

Click the calendar icon in the Search filter text box to search for workflows with a specific start date.

- **End Time**

Displays the date and time when the workflow ended.

Click the calendar icon in the Search filter text box to search for workflows with a specific end date.

- **Status**

Displays the running status of the job. The status can be one of the following:

- Failed

Indicates that the workflow failed during execution. Based on the time that has elapsed since the execution of the workflow, you can resume the workflow from the failed command.

- Executing

Indicates that the workflow is being executed. The workflow executes the plan that was created in the planning phase, which precedes any other execution.

- Successful

Indicates that the workflow has been successfully executed.

- Canceled

Indicates that the workflow has been canceled by a user.

- Waiting for Approval

Indicates that an approval point is part of the workflow. The workflow remains in this state until the specified user approves the workflow execution.

- Scheduled

Indicates that the planning of the workflow is complete and the workflow is scheduled for execution.

- Aborting

Indicates that the workflow is being aborted. An aborted workflow does not continue with the execution;

any previously completed part of the workflow remains completed.

- **Obsolete**

Indicates that the scheduled workflow has not been executed within the specified time per the schedule.

- **Planning**

Indicates the resolution of the workflow design, location of all resources, verification of design viability, and formulation of the execution plan. Planning can be a stand-alone action, part of the design verification, or part of execution because every execution starts with a fresh planning.

- **Pending**

Indicates that the workflow is in the planning queue. This is an internal status. The workflow is retrieved for planning from this status.

- **Partially Successful**

Indicates that although the workflow has been successfully executed, there are one or more steps that have failed. The execution is completed because the failed steps have been configured such that the workflow execution continues even when the step has failed.

- **Completed**

Displays the number of steps that are completed of the total number of steps for the selected workflow.

- **Submitted By**

Displays the user name of the user who submitted the workflow.

- **Submitted At**

Displays the date and time that the workflow was submitted.

Click the calendar icon in the Search filter text box to search for workflows with a specific submitted at date.

- **Execution Comment**

Displays the comment specified for the workflow execution.

- **Scheduled For**

Displays the scheduled date and time for the execution of the workflow.

Click the calendar icon in the Search filter text box to search for workflows with a specific scheduled for date. When a filter to view the jobs at a later date is applied on the column, jobs with "Job # zero" might be displayed. This indicates that the job is not yet created and will be created at the scheduled time.

- **Recurring ID**

Displays the identifier of the recurring schedule.

- **Schedule Name**

Displays the name of the schedule.

- **Last Status Change**

Displays the time when a status was changed.

Click the calendar icon in the Search filter text box to search for workflows with a specific last status change date.

- **Approval Point Comment**

Indicates the message that is displayed to the user at the last approval point, if applicable, during the execution of the workflow.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(Details)**

Opens the Monitoring window for the selected workflow, which contains the following tabs for detailed information about the workflow:

- Flow
- Execution Plan
- User Inputs
- Return Parameters
- History

You can also double-click an entry in the table to open the Monitoring window to view detailed information.

-  **(Abort)**

Stops the execution process from continuing. This option is enabled for workflows that are in execution mode.

-  **(Reschedule)**

Opens a Reschedule Workflow dialog box, which enables you to change the time of execution for the workflow. The option is enabled for workflows that are in the Scheduled state.

-  **(Resume)**

Opens a Resume Workflow dialog box, which enables you to resume the execution of the workflow after modifying environment issues (for example, wrong credentials to an array, missing licenses, or array is down). The option is enabled for workflows that are in the Failed state.

-  **(Approve & Resume)**

Enables you to approve the execution of the workflow and continue the execution process. This option is enabled for workflows that are in the Waiting For Approval state.

-  **(Reject & Abort)**

Enables you to reject the execution of the workflow and stop the execution process. This option is enabled for workflows that are in the **Waiting For Approval** state.

-  **(Clean Reservation)**

Enables you to clean the resource reservation made for a workflow from the local cache. Clean reservation is available only for scheduled, failed, and partially successful workflows. You cannot resume reservation after it is cleaned.

-  **(Refresh)**

Refreshes the list of workflows. The view refreshes automatically. You can toggle auto refresh on and off by clicking  in the status bar.

Recurring Executions window

The **Recurring Executions** window enables you to suspend, resume, or delete a recurring execution that is associated with a workflow. A workflow that has a recurring execution is executed on a recurring basis at a specified frequency. You can access this window by selecting **Execution > Recurring Executions**.

- Recurring Executions table
- Toolbar

Recurring Executions table

The Recurring Schedules table lists the existing recurring schedules that are associated with a workflow in a table format. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The following table outlines the supported operators for alphabetic and numeric columns in the Recurring Executions table, available in the **Search** filter text box.

Alphabetic	Numeric
<ul style="list-style-type: none"> Contains Does not contain Starts with Ends with Equals Does not equal Reset 	<ul style="list-style-type: none"> Equals Does not equal Less than Greater than Less than or equal to Greater than or equal to Between Reset

The Recurring Schedules table contains the following columns:

- **ID**

Displays the identifier of the schedule.

- **Workflow Name**

Displays the name of the workflow.

- **User Inputs**

Displays the names and values of the user inputs associated with the workflow.

- **Schedule**

Displays the name of the schedule associated with the workflow.

- **Status**

Displays the status of the schedule. The possible values are Active and Suspended.

- **Next Execution Date**

Displays the date and time when the workflow associated with the schedule will be executed next.

You can search for a schedule by entering the date and time of execution in the **Search** filter text box.

- **Updated At**

Displays the date and time when the association of the workflow with the schedule was updated.

Click the calendar icon in the **Search** filter text box to search for recurring schedules on a specific date.

- **Updated By**

Displays the name of the user who modified the association of the workflow with the schedule.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(Suspend)**

Opens the Suspend Recurring Execution dialog box, which enables you to suspend the execution of an active workflow.

-  **(Resume)**

Resumes the execution of a suspended workflow.

-  **(Delete)**

Opens the Delete Recurring Execution confirmation dialog box, which enables you to delete the selected schedule.

Reservations window

The Reservations window displays each workflow that is marked for reservation, and enables you to manage the reservations. You can access this window by selecting **Execution > Reservations**.

A reservation helps ensure that a selected resource is available for a specific scheduled workflow when that workflow is executed.

- Reservations table
- Toolbar

Reservations table

The Reservations table lists the workflows that are submitted for reservation. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The following table lists the supported operators for alphabetic and numeric columns in the Reservations table,

available from the **Search** filter text box.

Alphabetic	Numeric
<ul style="list-style-type: none">ContainsDoes not containStarts withEnds withEqualsDoes not equalReset	<ul style="list-style-type: none">EqualsDoes not equalLess thanGreater thanLess than or equal toGreater than or equal toBetweenReset

The Reservations table contains the following columns:

- Job #**

Displays the identification number of the job.

The job identification number is unique and is assigned by the server when it starts the job.

- Workflow**

Displays the workflow name.

- Time of Execution**

Displays the time when the job is scheduled to run or when the job was executed.

Click the calendar icon in the **Search** filter text box to search for a specific execution date.

- Time of reservation**

Displays the time when the job was scheduled, which is the time when the reservation was created.

Click the calendar icon in the **Search** filter text box to search for a specific reservation date.

- Command Name**

Displays the command for which the reservation was made. For example, for a workflow executing the command `Create Volume myVolume of size 20MB`, `Create Volume` is displayed.

- Reservation**

Displays the description of the reservation, which is generated from the string representation of the command.

- Workflow Status**

Displays the status of the workflow execution job. The status options are as follows:

- All
- Failed
- Executing
- Successful
- Canceled
- Waiting for Approval
- Scheduled
- Aborting
- Obsolete
- Planning
- Pending
- Partially Successful

- **Cache Updated**

Displays whether the reservation has been verified and reflected in the cached data (YES or NO). This cache update is performed by the data acquisition process.

- **Last Error**

Displays the error message that is generated when the reservation is simulated.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(Delete)**

Opens the Delete Reservations confirmation dialog box, which enables you to delete the selected reservation.

-  **(Refresh)**

Refreshes the contents of the Reservations table.

Schedules window

The Schedules window enables you to create, edit, and delete the schedule of a workflow. You can schedule a workflow to be executed at a particular date and time. You can access this window by selecting **Execution > Schedules**.

- Schedules table
- Toolbar

Schedules table

The Schedules table lists the existing schedules for workflow execution in a table format. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The following table lists the supported operators for alphabetic and numeric columns in the Schedules table, available from the **Search** filter text box.

Alphabetic	Numeric
<ul style="list-style-type: none">• Contains• Does not contain• Starts with• Ends with• Equals• Does not equal• Reset	<ul style="list-style-type: none">• Equals• Does not equal• Less than• Greater than• Less than or equal to• Greater than or equal to• Between• Reset

The Schedules table contains the following columns:

- **ID**

Displays the identifier of the schedule.

- **Name**

Displays the name of the schedule.

- **Description**

Displays the description of the schedule.

- **Association Count**

Displays the number of workflows that are associated with the schedule.

- **Updated At**

Displays the date and time at which the schedule was modified.

Click the calendar icon in the **Search** filter text box to search for schedules updated on a specific date.

- **Updated By**

Displays the name of the user who modified the schedule.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Schedule dialog box, which enables you to add a new schedule.

-  **(Edit)**

Opens the Edit Schedule dialog box, which enables you to edit the selected schedule.

-  **(Delete)**

Opens the Delete Schedule confirmation dialog box, which enables you to delete the selected schedule.

New Schedule dialog box

You can use the New Schedule dialog box to create a new schedule for any workflow so that the workflow is executed on a specified frequency. For example, you can schedule a workflow to be executed every Monday.

New Schedule Configuration properties

- **Name**

Enables you to specify a name for the schedule.

- **Description**

Enables you to enter a description for the schedule.

- **Frequency**

Enables you to specify the frequency at which the workflow associated with the schedule must be executed. The Hourly option is selected by default. Frequency can be Hourly, Daily, Weekly, or Monthly.

For example, if you want a workflow to be executed every Tuesday at 9 A.M., you must select the weekly option and enter the time as 9:00 and day as Tuesday. You must use a colon when you specify the time.

The 24-hour time format is supported. The entered data will be based on the server time.

Command buttons

- **OK**

Saves the configurations settings and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Workflows window

The Workflows window displays all of the available workflows, in alphabetical order. You can access this window by selecting **Workflow Design > Workflows**.

Depending on your role and account privileges, this window might not be displayed.

- Workflows table
- Toolbar

Workflows table

The Workflows table lists the available workflows. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The Workflows table contains the following columns:

- **Certification**

Indicates whether the workflow is user-created () , PS () , community () , user-locked () , or NetApp-certified () .

You can search for workflows by selecting the required option check boxes from the filter list.

- **Name**

Displays the name of the workflow.

You can search for a workflow by entering its name in the **Search** filter text box.

- **Scheme**

Displays the schemes associated with workflows. A scheme represents the data model for a system. For example, a vc scheme contains data relevant to your virtual environment, such as virtual machines, hosts, and datastores.

You can search for a workflow by entering one of its schemes in the **Search** filter text box.

- **Entity Version**

Displays the version number of the object in `major.minor.revision` format—for example, 1.0.0.

You can search for a workflow by entering the version number in the **Search** filter text box.

- **Description**

Displays the description of the workflow.

You can search for a workflow by entering its description in the **Search** filter text box.

- **OS Compatibility**

Indicates whether the workflow is compatible with the host operating system.

- **Min. Software Versions**

Specifies the minimum versions of the software that is required to execute the workflow. For example, the software can be clustered Data ONTAP 8.2.0 and vCenter 6.0. The versions are displayed as comma-separated values.

- **Categories**

Displays the user-defined labels associated with workflows.

You can use the categories to organize collections of workflows. For example, to help distinguish workflows, there might be a separate category for provisioning tasks, a category for maintenance tasks, and a category for decommissioning tasks. In addition, categories are used to restrict workflow execution for specific operators.

You can search for workflows by entering a category in the **Search** filter text box.

- **Last Updated On**

Displays the date and time that the workflow was last updated.

You can search for workflows by selecting the required time category from the filter drop-down list.

- **Updated By**

Displays the name of the user who updated the workflow.

You can search for workflows by entering the user name in the **Search** filter text box.

- **Locked By**

Displays the name of the user who locked the workflow.

You can search for workflows by entering the user name in the **Search** filter text box.

- **Ready for production**

Identifies whether the workflow is marked ready for production (true or false).

You can search for workflows by selecting the required production option check box from the filter list.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Workflow window, which enables you to enter the details for the new workflow.

-  **(Edit)**

Opens the workflow editor for the selected workflow, which enables you to edit the workflow. You can also double-click the workflow to open the workflow editor.

-  **(Clone)**

Opens the New Workflow <selected_workflow_name> - copy window, which enables you to create a clone or a copy of the selected workflow.

-  **(Lock)**

Opens the Lock Workflow confirmation dialog box, which enables you to lock the selected workflow. This option is enabled only for the workflows that you have created.

-  **(Unlock)**

Opens the Unlock Workflow confirmation dialog box, which enables you to unlock the selected workflow. This option is enabled only for the workflows locked by you. However, administrators can unlock workflows locked by other users.

-  **(Delete)**

Opens the Delete Workflow confirmation dialog box, which enables you to delete the selected workflow. This option is enabled only for the workflows that you have created.

-  **(Export)**

Opens the File Download dialog box, which enables you to save the selected workflow as a .dar file. This option is enabled only for the workflows that you have created.

-  **(Execute)**

Opens the Execute Workflow <selected_workflow_name> dialog box for the selected workflow, which enables you to execute the workflow.

-  **(Add To Pack)**

Opens the Add To Pack Workflow dialog box, which enables you to add the workflow and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for workflows for which the certification is set to **None**.

- **(Remove From Pack)**

Opens the Remove From Pack Workflow dialog box for the selected workflow, which enables you to delete or remove the workflow from the pack.



The Remove From Pack feature is enabled only for workflow for which the certification is set to **None**.

New Workflow window

The **New Workflow** window displays a newly created workflow and its associated commands.

New Workflow

The **New Workflow** window displays the name of the workflow in the header of the window. The associated commands are displayed as blue boxes in columns that are presented in the sequential order of execution—read left to right, top to bottom. The variables and objects for each command are displayed as gray boxes in the column of the command.

- **Insert a row**

Adds a new row above or below the selected row in the workflow.

You can click to add a row below the last available row.

- **Copy row**

Copies the selected row from the workflow and saves it to the clipboard. Copying a row does not copy the repetition details set for the row.

- **Paste row**

Places the last item saved to the clipboard in a new row below the selected row. Unique names are used for variables; however, variables in expressions are not changed.

- **Repeat row**

Opens the **Row Repetition Details** dialog box.

- **Edit row repetition**

Opens the **Row Repetition Details** dialog box, which enables you to modify the **Repeat Row** action set for the selected row.

- **Remove row repetition**

Cancels the **Repeat Row** action set in the **Row Repetition Details** dialog box for the selected row.

- **Add condition**

Opens the Conditions for row <row number> dialog box, which enables you to select a condition. The selected condition is applicable to all the commands in the row. All commands in the row will be executed only if the selected condition is met.

- **Remove row**

Deletes the selected row from the workflow.

- **New Approval Point**

You can click  to open the **New Approval Point** dialog box, which enables you to add an approval point before the execution of the command.

- **Edit Approval Point**

You can click  to open the **Edit Approval Point** dialog box, which enables you to edit the approval point comment.

- **Add Command Parameters**

Adds parameters for a command in the selected row.

You can place your cursor below the selected command, at the required row, and click  to open the Parameters for <command_name> dialog box.

- **Edit Command Parameters**

Opens the Parameters for <command_name> dialog box to enable you to modify the selected parameters for a command.

You can place your cursor over the gray box that you want to edit and click to open the Parameters for <command_name> dialog box.

- **Delete Command Parameters**

Removes the parameter.

You can move the cursor over the gray box that you want to delete and click the “X” on the upper-right corner of the gray box.

Command buttons

The command buttons are available at the bottom of the workflow window. The commands can also be accessed from the right-click menu in the window.

- **Preview**

Opens the **Preview Workflow** dialog box, which enables you to specify user input attributes.

- **Save As**

Enables you to save the workflow with a new name.

- **Save**

Saves the workflow.

- **Close**

Closes the window. In the process, if any changes were made to the configuration, a **Save Changes confirmation** dialog box is opened, prompting you to either save the changes, close the window without saving changes, or cancel the closing action.

Parameters for commands dialog box

You can specify parameters and other settings for the execution of commands through the Parameters for commands dialog box.

The dialog box displays one or more dictionary object tabs and the following tabs:

- Other parameters
- Advanced

<Dictionary objects> tabs

- **Define <dictionary object>**

Enables you to specify the command parameters that are mapped to the dictionary object by specifying attributes, using a previously defined object, or searching for an existing object.

- **by filling-in attributes**

Enables you to enter attributes for this variable. You can use the Resource Selection dialog box for fields marked with  . Optionally, you can use a template that includes predefined values for certain attributes of the object. You can view and use additional attributes by clearing the **Show only attributes used by <dictionary object>** check box. The box next to **Define <dictionary object>** displays the default name of the selected variable. You can edit the name of the variable. The mandatory attributes are marked with an asterisk (*) and red border for the box.

- **by using a previously defined <dictionary object>**

Enables you to select a previously defined variable. You can select the previously defined dictionary object in the box next to **Define <dictionary object>**.

- **by searching for an existing <dictionary object>**

Enables you to define a variable by searching for an existing dictionary object. You can specify the search criteria for the dictionary object. You can specify one of the following action if the dictionary object is found:

- Abort workflow
- Disable the command
- Fill-in attributes for the <dictionary object> and execute the command



This option is similar to the **by filling-in attributes** option.

Other Parameters

Enables you to specify the command parameters that are not mapped to any dictionary object for the execution of the command.

Advanced

Enables you to specify conditions for the execution of the command and provide a description. You can also configure the workflow so that the workflow execution continues even if one or more commands in the workflow have failed.

- **Execute this command**

- **Always**

Executes the command unconditionally.

- **If the following variable was found**

Enables you to execute a command only when the specified variable is found. You can specify the variable in the adjacent box.

- **If the following variable was not found**

Enables you to execute a command only when the specified variable is not found. You can specify the variable in the adjacent box.

- **If the following expression is TRUE**

Enables you to execute a command only when the specified MVFLEX Expression Language (MVEL) expression is “true”. You can specify the expression in the adjacent box.

- **Description**

Enables you to enter a description for the command.

- **If execution fails**

- **Abort workflow execution**

Enables you to end the workflow execution.

- **Continue execution from the next step**

Enables you to continue the workflow execution from the next step.

- **Continue execution from the next row**

Enables you to continue the workflow execution from the next row.

Resource Selection dialog box

The Resource Selection dialog box enables you to search for resources and map the resources to a command.

- Select <dictionary object> by tab

- Advanced tab

Select <dictionary object> by tab

This tab enables you to specify a finder or filters as the search criteria and enter attributes for the selected search criteria. You can also save the selected filters as a finder.

- **Define filter rules**

Enables you to define a set of rules for dictionary entry resources, such as vFiler units, aggregates, and virtual machines.

The filter rules can contain one or more groups of rules.

A rule consists of a dictionary entry attribute, an operator, and a value. The attribute can also include attributes of its references. For example, you can specify a rule for aggregates as follows: List all aggregates with name starting with the string “aggr” and have an available size greater than 5 GB. The first rule in the group is the attribute “name”, with the operator “starts-with”, and the value “aggr”. The second rule for the same group is the attribute “available_size_mb”, with the operator “>” and the value “5000”.

The **Define filter rules** option is disabled if you have selected a finder.

You can clear the values specified by clicking the **Reset** button.

- **Select one resource by its natural keys**

Enables you to select a resource based on the natural keys of the resource.

- **Finder**

Enables you to select a finder from the list. **None** is selected by default.

Filters are displayed below the finder list. You cannot select filters if you have selected a finder.

- **Parameters**

Enables you to enter the values for the selected finder or filters. You must enter all the required values to complete the search criteria.

- **Command buttons**

- **Save As Finder**

Saves the selected filters as a finder.

- **Test**

Enables you to test the parameters for the selected filters before specifying the values in the Parameters area.

- **OK**

Saves the configurations settings and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Advanced tab

This tab enables you to specify when the search should be performed and allows you to sort the resource attributes and the status of the resource attributes.

- **Execute this search always**

Executes the search without any restriction when the command is executed.

- **Execute search only when the following expression evaluates to TRUE**

Executes the search only when the value of the optional attribute specified in the box is “true”. You can use MVFLEX Expression Language (MVEL) syntax to specify the optional attribute.

- **Sorting table**

Displays the attributes and the sort order for the filters selected in the **Select <dictionary object> by** tab. The sort order is significant for resource selection—for example, if you selected descending as the sort order for aggregates, the aggregate with maximum `available_space` is selected as the resource. To change the sort order for an attribute, you can click the status column of the attribute and select the required sort order from the list.

- **Command buttons**

- **Up**

Moves the selected entry up one row in the Sorting table.

- **Down**

Moves the selected entry down one row in the Sorting table.

Incremental Naming Wizard

The Incremental Naming Wizard enables you to define the value of an attribute based on a search for an existing parameter.

- **Search criteria for existing <dictionary object>**

Opens the **Resource Selection** dialog box, which enables you to specify search criteria for the dictionary object.

- **Enter a value for <attribute> if no <dictionary object> matches the above search criteria**

Enables you to specify a value for the <attribute> when the <dictionary object> is not found using the specified search criteria.

- **If <dictionary object> was found using above search criteria, set value for <attribute> by**

Enables you to specify a value for the <attribute> when the <dictionary object> is not found using the specified search criteria.

- **Enter a value for <attribute> if no <dictionary object> matches the above search criteria**

Enables you to select a method for setting the value for the <attribute> if the <dictionary object> is found using the specified search criteria.

- **providing an increment value and suffix**

Enables you to enter a numeric value for the increments and optionally enter a suffix of the attribute name.

- **providing a custom expression**

Enables you to enter a custom expression for the value of the attribute. You can use MVFLEX Expression Language (MVEL) syntax to specify the value.

Row Repetition Details dialog box

The Row Repetition Details dialog box enables you to specify how the parameters in the row should be repeated.

Repeats

Enables you to select the required type of repeat option. The **Number of times** option is selected by default.

- **Number of times**

Enables you to specify the following:

- The number of times the specific row should be executed
- Index Variable
- Variables
- **Number of Times**

Enables you to specify a numeral for the number of times the specific row should be executed.

- **Index Variable**

Enables you to specify the name of the index variable for row repetition.

- **Variables**

Enables you to include additional variables that have to be used during row repetition.

- **Add**

Adds a new row in the variables table.

- **Remove**

Removes the selected row from the variables table.

- **For every resource in a group**

Enables you to specify the following:

- Resource Type
- Resource Search Criteria
- Resource Variable
- Index Variable
- Group Size Variable
- Variables
- **Resource Type**

Enables you to select the resource type.

- **Resource Search Criteria**

Opens the Resource Selection dialog box, which enables you to specify the criteria for searching resources of the selected resource type. The loop is executed for each searched resource based on the specified criteria.

- **Resource Variable**

Enables you to enter a name for the resource variable.

- **Index Variable**

Enables you to specify the index for row repetition.

- **Group Size Variable**

Enables you to enter a name for the group size variable.

- **Variables**

Enables you to include additional variables that have to be used during row repetition.

- **Command buttons**

- **OK**

Saves the settings and closes the dialog box.

- **Cancel**

Does not save any changes, if any, and closes the dialog box.

Workflow <workflow name> window

The Workflow <workflow name> window displays the selected workflow and its associated commands and parameters. The commands associated with the workflow are displayed as blue boxes in columns that are arranged in the sequential order of execution—left to right. The variables and objects for each command are displayed as gray boxes below the command.

Workflow tab

The Workflow tab enables you to select and use child workflows and commands, and customize the display name.

- **Available Steps**

Provides a list of available child workflows and commands that you can add to your workflow.

You can search for child workflows or commands by using the name, scheme, and minimum software version as search strings in the text box.

You can customize the workflow display by rearranging the order of the commands. To rearrange the commands, you can drag and drop commands in the required order. Double-clicking a command moves that command to the end of the list. You can modify the display name by double-clicking the display name of the command in the blue box. You can delete a command by moving the cursor over the blue box, and then clicking "X" on the upper-right corner.

You can click the row number to perform various functions on the row.

- **Insert a row**

Inserts a new row above or below the selected row in the workflow.

- **Copy row**

Copies the selected row from the workflow and saves it to the clipboard. The repetition details set for the row is copied when a row is copied.

- **Repeat row**

Opens the Row Repetition Details dialog box, which enables you to specify the type of repetition.

- **Add condition**

Opens the Condition for row <row number> dialog box, which enables you to select a condition. The selected condition is applicable to all the commands in the row. All commands in the row will be executed only if the selected condition is met.

- **Remove row**

Deletes the selected row from the workflow.

- **Paste Row**

Pastes the copied row above or below the selected row. This option is available after selecting the Copy row function.

Details tab

The Details tab enables you to provide general information about the workflow.

- **Workflow name**

Enables you to specify the name of the workflow using an alphanumeric string.

- **Entity Version**

Enables you to enter a version number for the workflow in the `major.minor.revision` format—for example, 1.0.0.

- **Categories**

Displays the associated category for the workflow. You can assign a workflow to a category from the Category window.

- **Workflow Description**

Enables you to enter a description of the workflow. You can click in the field to open the Edit Description dialog box.

- **Ready for production**

Enables you to mark the workflow ready for production, which allows the workflow to be executed and listed in the Workflows window.

- **Consider Reserved Elements**

Enables you to use the reservation feature for the workflow, when selected resources are available for the workflow until the reservation expiration period you have configured. The reservation feature enables you to exclude resources reserved by other workflows during resource selection.



You must not select this check box if you require the filters to search only the contents of the persistent cache without considering reservations set by other workflows.

- **Enable element existence validation**

Enables you to validate the existence of elements and avoid failure of certain actions during the execution of workflows. For example, you can avoid creating a new volume in an array with the same name as that of an existing volume.

- **Minimum Software Versions**

Specifies the minimum versions of the software that is required to execute the workflow. For example, the software can be clustered Data ONTAP 8.2.0 and vCenter 6.0. The versions are displayed as comma-separated values.

User Inputs tab

The User Inputs tab enables you to view and edit the user input attributes that you created when you defined the workflow objects and created user inputs. The values for the user input attributes are filled when you preview or execute the workflow.

You can double-click the user input that you want to edit, which opens the Edit Variable: <user_input> dialog box, enabling you to edit the user input.

You can customize the table display by rearranging the column order. To rearrange the columns, you can drag and drop columns in the required order. However, you cannot hide or delete any of these columns.

- **Name**

Displays the name of the user input attribute.

- **Display Name**

Displays the name that is displayed to the workflow user.

- **Type**

Displays the user input type, such as string, query, boolean, table, or password.

- **Values**

Displays the allowed values for the user input—for example, range for numbers and regular expression for strings.

- **Default Value**

Displays the default value of the user input.

- **Input Dependency**

Displays another user input from the list that provides a value to the selected user input.

- **Group**

Displays the name of the group for the user input attributes.

- **Mandatory**

Displays the status of the user input. If the check box is displayed as selected, the user input attributes are mandatory for the execution of the workflow.

- **Command buttons**

- **Up**

Moves the selected entry up one row in the table.

- **Down**

Moves the selected entry down one row in the table.

Constants tab

The Constants tab enables you to define the value of the constants that can be used multiple times in the workflow. You can specify the following as the value of constants:

- Numbers
- Strings
- MVEL expressions
- Functions
- User inputs
- Variables

You can customize the table display by sorting each column, as well as by rearranging the column order.

- **Name**

Displays the name of the constant.

- **Description**

Enables you to specify a description for the constant.

- **Value**

Enables you to specify a value for the constant.

- **Command buttons**

- **Add**

Adds a new row in the Constants table.

- **Remove**

Deletes the selected row from the Constants table.

You can also right-click the constants to use the copy and paste functionality.

Return Parameters tab

The Return Parameters tab enables you to define and provide a description of the return parameters for the workflow that can be viewed from the Monitoring window or from web services.

- **Parameter Value**

Enables you to specify the parameter value.

- **Parameter Name**

Enables you to specify the parameter name.

- **Description**

Enables you to specify a description for the selected parameter.

- **Command buttons**

- **Add Row**

Adds a new row in the Return Parameters table.

- **Remove Row**

Deletes the selected row from the Return Parameters table.

Help Content tab

The Help Content tab enables you to add, view, and remove the Help content for the workflow. The workflow Help content provides information about the workflow for storage operators.

Advanced tab

The Advanced tab enables you to configure a custom URI path for workflow execution through API calls. Each segment in the URI path can be a string or a valid name of the user input of the workflow in brackets.

For example, /devops/{ProjectName}/clone. The workflow can be invoked as a call to *https://WFA-Server:HTTPS_PORT/rest/devops/Project1/clone/jobs*.

Command buttons

The command buttons are available at the bottom of the workflow window. The commands can also be accessed from the right-click menu in the window.

- **Preview**

Opens the Preview Workflow dialog box, which enables you to specify user input attributes.

- **Save As**

Enables you to save the workflow with a new name.

- **Save**

Saves the configurations settings.

Execute Workflow dialog box

The Execute Workflow dialog box enables you to specify the required user inputs, time for execution, and execution comment for a workflow.

User Inputs

Enables you to specify the inputs for the workflow. The mandatory attributes are marked with an asterisk (*) and the border for the boxes are in red.

Options

Enables you to specify the time for the workflow execution.

- **Execute now**

Enables you to execute the workflow immediately.

- **Execute recurrently**

Enables you to set the workflow to recur periodically. This option is displayed only when a schedule has been created.

- **Execute once**

Enables you to schedule the execution of the workflow.

Execution Comment

Enables you to specify a comment for the workflow execution. This comment is displayed in the Execution Status window.

Edit Variable dialog box

The Edit Variable: <UserInputName> dialog box enables you to edit the user input attributes of a workflow.

Properties

- **Variable Name**

Enables you to edit the name of the user input. Renaming the user input updates all the references to the user input in the workflow.

- **Display Name**

Enables you to specify or edit the display name for the user input. You can specify a unique display name if you have created user input attributes with a similar name.

- **Description**

Enables you to specify or edit a description for the user input. The description appears as a tooltip when you execute or preview the workflow.

- **Type**

Enables you to select the type of user input. Fields or a link to a dialog box is displayed based on the selected options. Available options include the following:

- String: Enables you to enter a regular expression for valid values in the RegEx field—for example, a*.
- Number: Enables you to enter a numerical range in the Range field—for example, 1 to 15.
- Enum: Enables you to enter a closed list of values in the Enum Values field.
- Query: Enables you to enter a simple SQL query to obtain the result.

Users can select only the values from the first column of the result.

- Query (Multi-select): Enables you to enter an SQL query that obtains a list of values.

Users can select a single value or multiple values while executing the workflow—for example, you can select a single volume, multiple volumes, or volumes with their shares and exports.

You can select values from any column displayed in the result. Selecting a row selects the values from all the columns of the selected row.

- Boolean: Enables you to display a check box in the User Inputs dialog box.
- Table: Enables you to specify the column headers of the table that can be used to enter multiple values during the execution of the workflow—for example, a table that can be used to specify a list of node

names and port names.

You can also configure the type of column user input and the properties associated with the column user input. A dialog box is displayed based on the selected user input type. Available options include the following:

- String: Enables you to enter a regular expression for valid values in the RegEx field—for example, a*.
- Number: Enables you to enter a numerical range in the Range field—for example, 1 to 15.
- Enum: Enables you to enter a closed list of values in the Enum Values field.
- Query: Enables you to enter a simple SQL query to obtain the result.

Users can select only the values from the first column of the result. A query within the column of the table user input cannot reference other user inputs within the query.

- Boolean: Enables you to select “true” or “false” as the boolean value from the drop-down list.
- Password: Enables you to encrypt a password that is provided as an input by the user.

When encrypted, the password is displayed as a sequence of asterisk characters across the WFA application and in the log files.

- Dictionary: Enables you to add the table data for the selected dictionary entry.

The dictionary entry attribute selects the attribute that is to be returned. You can select a single value or multiple values while executing the workflow. For example, you can select a single volume, multiple volumes, or volumes with their shares and exports. By default, single values are selected. You can also select Rules for filtering. A rule consists of a dictionary entry attribute, an operator, and a value. The attribute can also include attributes of its references. For example, you can specify a rule for aggregates by listing all aggregates with name starting with the string “aggr” and have an available size greater than 5 GB. The first rule in the group is the attribute `name`, with the operator `starts-with`, and the value `kaggr`. The second rule for the same group is the attribute `available_size_mb`, with the operator `>` and the value `5000`.

- **Lock Values**

Enables you to specify whether the user is allowed to input a value other than the query-returned values. Selecting the check box does not allow the user to provide a value; only the query-returned values can be selected. This option is used with the Enum and Query type options.

- **Default Value**

Enables you to set the default value of the user input.

- **Mandatory**

Enables you to specify whether user input is mandatory for the execution of the workflow.

Group

- **Group Name**

Enables you to group the related user input attributes together. The user input attributes are displayed in groups that you have defined when previewing or executing the workflow. For example, you can group the

user input attributes related to Volume Details together.

- **Expand group by default**

Enables you to display all the user input attributes specified in a group as an expanded list. If the check box is not selected, the user input group appears collapsed.

Dependency

Enables you to enable a user input based on the value entered for another user input. For example, in a workflow that configures the NAS protocol, you can specify the required user input for protocol as “NFS” to enable the “read-write host lists” user input or “CIFS” to enable “CIFS ACL configuration”.

- **Choose user input to depend on**

Enables you to select the user input required to enable the user input displayed in the Variable Name field.

- **Applicable values (comma separated)**

Enables you to specify the value of the conditional user input that enables the user input displayed in the Variable Name field.

Preview Workflow dialog box

The Preview Workflow dialog box enables you to specify values for user inputs associated with a workflow and then to preview the execution of that workflow with the specified values.

User Inputs values

The Preview Workflow dialog box displays the user inputs associated with the selected workflow and enables you to set the values for user inputs.

Command buttons

- **Preview**

Opens the Monitoring window for the associated workflow.

- **Cancel**

Closes the dialog box.

Monitoring window

The Monitoring window displays details about results of the planning or execution phase of the workflow. The tabs in the window display read-only content. You can access this window from various areas of the WFA application when selecting the **Details**, **Preview**, or **Execute** options.

- **Status area**

- Flow tab
- Details tab
- Execution Plan tab
- User Inputs
- Return Parameters tab
- History tab

Status area

The Status area above the tabs displays detailed information about the result of the planning or execution process.

- **Workflow status**

At the top of the window, a color-coded header displays the result of the planning or execution process:

- Green indicates a successful action—for example, Planning Completed or Execution Completed.
- Red indicates failure—for example, Planning Failed or Execution Failed.

- **Less/More info action**

This action link is available when the preview or execution process fails. You can toggle between **Less info** and **More info** using the action link. Depending on the action, the link opens or closes the information box to show more or less information about the failed result.

- **Detailed status**

This box is available when the preview or execution process fails. It displays details about the planning or execution process. When a process fails, this box is open by default and provides details of the cause of the failure.

Flow tab

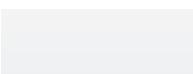
The Flow tab provides a graphical presentation of the workflow.

Commands and child workflows are displayed in the top row with their objects and variables displayed below. The status of the command details and child workflows during preview or execution is displayed using various colors.

The following table shows the following:

- The color scheme used to display the status of the command or workflow execution
- The icons used in the window

Color scheme and icon	Status
	Execution is successful

Color scheme and icon	Status
	Execution failed
	Execution is in progress
	Execution skipped
	Child workflow header
	Opens the logs
	The child workflow includes an approval point

Details tab

The Details tab provides more detailed information about the workflow. The information is provided in table format, with the commands in the header row and the associated objects and variables listed below. This tab provides each command with all of its arguments and parameters.

This tab is useful for debugging a failed preview or execution of the workflow.

Execution Plan tab

The Execution Plan tab provides a translated list of the workflow commands and their arguments that are being executed (scripts or commands).

You can use this tab for debugging a failed preview or execution of the workflow.

User Inputs tab

The User Inputs tab displays values entered by users during the preview or execution of the workflow.

You can use the sort arrows (▼ or ▲) in the column header to sort the column entries in ascending or descending order.

Return Parameters tab

The Return Parameters tab lists the output of the workflow, by parameter name and parameter value.

You can access these return parameters by using web services.

History tab

The History tab lists changes in the statuses for the workflow, when the status change occurred, who initiated the action, and the message associated with the status change.

Command buttons

- **Download Logs**

Enables you to download a `zip` file of all the log files, including the specific execution logs.

- **Column Visibility**

Enables you to show or hide columns in the Details tab.

- **OK**

Closes the Monitoring window.

New Approval Point dialog box

The New Approval Point dialog box enables you to add an approval point as a checkpoint in a workflow to pause the workflow execution and resume it based on your approval.

- **Enter comment (optional)**

Enables you to provide information, such as time of change, user, and comments, which enables you to see when and why the workflow execution was paused or resumed.

The approval point comments can include MVFLEX Expression Language (MVEL) expressions.

- **Enter the execution condition (if any)**

Enables you to specify conditions for the execution of a command:

- Always

Executes the command unconditionally.

- If the following variable was found

Enables you to execute a command only when the specified variable is found. You can specify the variable in the box.

- If the following variable was not found

Enables you to execute a command only when the specified variable is not found. You can specify the variable in the box.

- If the following expression is TRUE

Enables you to execute a command only when the specified MVEL expression is “true”. You can specify the expression in the box.

Command buttons

- **OK**

Enables you to add the approval point.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Edit Approval Point dialog box

The Edit Approval Point dialog box enables you to modify an approval point that was added as a check point in a workflow to pause the workflow execution and resume it based on your approval.

- **Enter comment (optional)**

Enables you to provide information, such as time of change, user, and comments, which enables you to see when and why the workflow execution was paused or resumed.

The approval point comments can include MVFLEX Expression Language (MVEL) expressions.

- **Enter the execution condition (if any)**

Enables you to specify the following conditions for the execution of a command:

- Always

Executes the command unconditionally.

- If the following variable was found

Enables you to execute a command only when the specified variable is found. You can specify the variable in the box.

- If the following variable was not found

Enables you to execute a command only when the specified variable is not found. You can specify the variable in the box.

- If the following expression is TRUE

Enables you to execute a command only when the specified MVEL expression is “true”. You can specify the expression in the box.

Command buttons

- **Remove Approval**

Enables you to remove the approval point before the command is executed.

- **OK**

Enables you to modify the changes to the approval point settings.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Finders window

The Finders window displays the available finders in alphabetical order. You can access this window by selecting **Workflow Design > Finders**.



Depending on your role and account privileges, this window might not be displayed.

A finder is a search operation for locating resources. A finder consists of filtering rules that remove irrelevant resources. Finders search for information about WFA objects in the WFA-cached repositories.

- Finders table
- Toolbar

Finders table

The Finders table lists the available finders.

Each entry is identified as one of the following:

- - content created by users
- - content developed by Professional Services (PS), which is available only on custom installations made by PS
- - packs developed by users
- - content created by users that is locked
- - NetApp-certified content

You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

- enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking clears and resets the filtering selections.
- on each column header enables you to filter based on the content of the columns. Clicking in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows (for ascending and for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The Finders table contains the following columns:

- **Certification**

Indicates whether the finder is user-created () , PS () , community () , user-locked () , or NetApp-certified () .

You can search for finders by selecting the required check boxes from the filter list.

- **Name**

Displays the name of the finder.

You can search for a finder by entering its name in the Search filter text box.

- **Scheme**

Displays the scheme associated with the finders. A scheme represents the data model for a system. For example, a vc scheme contains data relevant to your virtual environment, such as virtual machines, hosts, and datastores.

You can search for a finder by entering its scheme in the Search filter text box.

- **Type**

Displays the dictionary object type of the finder (for example, aggregate array or CIFS).

You can search for finders by entering the type in the Search filter text box.

- **Entity Version**

Displays the version number of the object in `major.minor.revision` format—for example, 1.0.0.

You can search for a finder by entering the version number in the Search filter text box.

- **Description**

Displays the description of the finder.

You can search for a finder by entering its description in the Search filter text box.

- **Last Updated On**

Displays the date and time when the finder was last updated.

You can search for finders by selecting the time category from the filter drop-down list.

- **Updated By**

Displays the user who updated the finder.

You can search for finders by entering the user name in the Search filter text box.

- **Locked By**

Displays the user who locked the finder.

You can search for finders by entering the user name in the Search filter text box.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Finder window, which enables you to create a finder.

-  **(Edit)**

Opens the Finder <finder_name> window for the selected finder, which enables you to edit the finder.

You can also double-click the finder to open the Edit Finder window.

-  **(Clone)**

Opens the New Finder <finder_name> - copy window, which enables you to create a copy of the selected finder.

-  **(Lock)**

Opens the Lock Finder confirmation dialog box, which enables you to lock the selected finder.

-  **(Unlock)**

Opens the Unlock Finder confirmation dialog box, which enables you to unlock the selected finder.

This option is enabled only for the finders that you have locked. Administrators can unlock finders that were locked by other users.

-  **(Delete)**

Opens the Delete Finder confirmation dialog box, which enables you to delete the selected user-created finder.



You cannot delete a WFA finder, PS finder, or sample finder.

-  **(Export)**

Enables you to export the selected user-created finder.



You cannot export a WFA finder, PS finder, or sample finder.

-  **(Test)**

Opens the Test Finder dialog box, which enables you to test the selected finder.

-  **(Add To Pack)**

Opens the Add To Pack Finders dialog box, which enables you to add the finder and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for finders for which the certification is set to **None**.

-  **(Remove From Pack)**

Opens the Remove From Pack Finders dialog box for the selected finder, which enables you to delete or

remove the finder from the pack.



The Remove From Pack feature is enabled only for finders for which the certification is set to **None**.

New Finder dialog box

The New Finder dialog box enables you to create new finders.

- Properties tab
- Filters
- Returned Attributes tab

Properties tab

The **Properties** tab enables you to specify a name for the finder, select the type of WFA dictionary object, and enter a description for the finder. The tab also displays the input parameters after a filter is selected in the **Filters** tab.

- **Name**

Enables you to enter a name for the finder.

- **Type**

Enables you to select the type of object from the list such as aggregate, igroup, and cluster. The filters for the selected object are displayed in the **Filters** tab.

- **Entity Version**

Enables you to enter the version number for the finder in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to enter a description for the finder.

- **Input Parameters table**

Displays the input parameters for the selected filters.

Filters tab

Enables you to select the required filters from the available list of filters.

- **Available Filters**

Displays the list of available filters for the selected object in the **Properties** tab.

- **Selected Filters**

Displays the list of filters selected from the available filters.

- **Selection buttons**

Moves the selected entry from one box to another.

Returned Attributes tab

Enables you to see details about the available filters, select the required attributes of the available filters, and specify the sort order for the attributes.

- **Available**

Displays the available filters and the attributes for the filters.

- **Selected**

Displays the selected filters and the attributes, the aliases, and the sort order for the selected filters.

- **Command buttons**

- **Up**

Moves the selected entry up one row in the Selected table.

- **Down**

Moves the selected entry down one row in the Selected table.

- **Selection buttons**

Moves the selected entry from one table to another.

Command buttons

- **Test**

Enables you to test the finder that you have defined.

- **Save**

Saves the finder and closes the dialog box.

- **Cancel**

Does not save the finder and closes the dialog box.

Edit Finder dialog box

The Edit Finder dialog box enables you to edit the finder.

- Properties tab
- Filters
- Returned Attributes tab

Properties tab

The **Properties** tab enables you to edit the name of the finder, the type of WFA dictionary object, and the description for the finder. The tab also displays the input parameters of the filter used in the finder.

- **Name**

Enables you to edit the name of the finder.

- **Type**

Enables you to select the type of WFA dictionary object from the list such as aggregate, igrp, and cluster. The filters for the selected object are displayed in the **Filters** tab.

- **Entity Version**

Enables you to enter the version number for the finder in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to edit the description for the finder.

- **Input Parameters table**

Displays the input parameters of the selected filters.

Filters tab

Enables you to select the required filters from the available list of filters.

- **Available Filters**

Displays the list of available filters for the selected object in the **Properties** tab.

- **Selected Filters**

Displays the list of filters selected from the available filters.

- **Selection buttons**

Moves the selected entry from one box to another.

Returned Attributes tab

Enables you to see details about the available filters, select the required attributes of the available filters, and specify the sort order for the attributes.

- **Available**

Displays the available filters and the attributes for the filters.

- **Selected**

Displays the selected filters and the attributes, the aliases, and the sort order for the selected filters.

- **Command buttons**

- **Up**

Moves the selected entry up one row in the Selected table.

- **Down**

Moves the selected entry down one row in the Selected table.

- **Selection buttons**

Moves the selected entry from one table to another.

Command buttons

- **Test**

Enables you to test the finder that you have selected to edit.

- **Save**

Saves the changes and closes the dialog box.

- **Cancel**

Does not save the changes, if any, and closes the dialog box.

Clone Finder dialog box

The Clone Finder dialog box enables you to create a copy of the finder and edit the cloned finder.

- Properties tab
- Filters
- Returned Attributes tab

Properties tab

The Properties tab enables you to edit the name of the finder, the type of object, and the description for the finder. The tab also displays the input parameters of the filter used in the finder.

- **Name**

Enables you to edit the name of the finder. The name of the finder you have selected to clone is used as the name for the clone and appended with – copy by default.

- **Type**

Enables you to select the type of object from the list. The filters for the selected object are displayed in the Filters tab.

- **Entity Version**

Enables you to enter the version number for the finder in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to edit the description for the finder.

- **Input Parameters table**

Displays the input parameters of the selected filters.

Filters tab

Enables you to select the required filters from the available list of filters.

- **Available Filters**

Displays the list of available filters for the selected object in the Properties tab.

- **Selected Filters**

Displays the list of filters selected from the available filters.

- **Selection buttons**

Enables you to move the selected entry from one box to another.

Returned Attributes tab

Enables you to see the details of the available filters, select the required attributes of available filters, and specify the sort order for the attributes.

- **Available**

Displays the available filters and the attributes for the filters.

- **Selected**

Displays the selected filters and the attributes, the aliases, and the sort order for the selected filters.

- **Command buttons**

- **Up**

Moves the selected entry up one row in the Selected table.

- **Down**

Moves the selected entry down one row in the Selected table.

- **Selection buttons**

Moves the selected entry from one table to another.

Command buttons

- **Test**

Enables you to test the finder that you have selected to clone.

- **Save**

Saves the finder and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Filters window

The Filters window displays the available filters in alphabetical order. You can access this window by selecting **Workflow Design > Filters**.



Depending on your role and account privileges, this window might not be displayed.

Filters are query rules that remove irrelevant resources during the query-based search for locating resources. Filters are used to develop a finder.

- Filters table
- Toolbar

Filters table

The Filters table lists the available filters. Each entry is identified as one of the following:

- - content created by users
- - content developed by Professional Services (PS), which is available only on custom installations made by PS
- - packs developed by users
- - content created by users that is locked
- - NetApp-certified content

You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

- enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking clears and resets the filtering selections.
- on each column header enables you to filter based on the content of the columns. Clicking in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows (for ascending and for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.

- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The Filters table contains the following columns:

- Certification**

Indicates whether the filter is user-created (User icon), PS (PS icon), community (Community icon), user-locked (Locked icon), or NetApp-certified (NetApp icon).

You can search for filters by selecting the one or more check boxes from the filter list.

- Name**

Displays the name of the filter.

You can search for a filter by entering its name in the Search filter text box.

- Scheme**

Displays the scheme associated with the filters. A scheme represents the data model for a system. For example, a vc scheme contains data relevant to your virtual environment, such as virtual machines, hosts, and datastores.

You can search for a filter by entering its scheme in the Search filter text box.

- Type**

Displays the dictionary object type of the filter (for example, aggregate array or CIFS).

You can search for filters by entering the type in the Search filter text box.

- Entity Version**

Displays the version number of the object in major.minor.revision format—for example, 1.0.0.

You can search for a filter by entering the version number in the Search filter text box.

- Description**

Displays the description of the filter.

You can search for a filter by entering its description in the Search filter text box.

- Last Updated On**

Displays the date and time when the filter was last updated.

You can search for filters by selecting the time category from the filter drop-down list.

- Updated By**

Displays the user who updated the filter.

You can search for filters by entering the user name in the Search filter text box.

- **Locked By**

Displays the user who locked the filter.

You can search for filters by entering the user name in the Search filter text box.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Filter window, which enables you to create a filter.

-  **(Edit)**

Opens the Filter <filter_name> window for the selected filter, which enables you to edit the filter.

You can also double-click the filter to open the Edit Filter window.

-  **(Clone)**

Opens the New Filter <filter_name>_copy window, which enables you to create a copy of the selected filter.

-  **(Lock)**

Opens the Lock Filter confirmation dialog box, which enables you to lock the selected filter.

-  **(Unlock)**

Opens the Unlock Filter confirmation dialog box, which enables you to unlock the selected filter.

This option is enabled only for the filters that you have locked. Administrators can unlock filters that were locked by other users.

-  **(Delete)**

Opens the Delete Filter confirmation dialog box, which enables you to delete the selected user-created filter.



You cannot delete a WFA filter, PS filter, or sample filter.

-  **(Export)**

Enables you to export the selected user-created filter.



You cannot export a WFA filter, PS filter, or sample filter.

-  **(Test)**

Opens the Test Filter dialog box, which enables you to test the selected filter.

-  **(Add To Pack)**

Opens the Add To Pack Filters dialog box, which enables you to add the filter and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for filters for which the certification is set to None.

-  **(Remove From Pack)**

Opens the Remove From Pack Filters dialog box for the selected filter, which enables you to delete or remove the filter from the pack.



The Remove From Pack feature is enabled only for filters for which the certification is set to None.

New Filter dialog box

You can use the New Filter dialog box to create a new filter. You can use the filter to develop finders.

- Properties tab
- Query tab

Properties tab

The **Properties** tab enables you to specify a name for the filter, select a dictionary object, and enter a description for the filter.

- **Name**

Enables you to enter a name for the filter.

- **Type**

Enables you to select a dictionary object from the list.

- **Entity Version**

Enables you to enter the version number for the filter in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to enter a description for the filter.

Query tab

Enables you to enter an SQL query, and edit the label and the description for the attributes.

- **SQL Query**

Enables you to enter an SQL query for the filter.

- **Input Parameters table**

Displays the list of parameters from the SQL query that you have entered which require user input when the filter is used. You can click the **Label** or the **Description** column of a parameter to edit the label or enter a description.

- **Returned Attributes table**

Displays the list of returned attributes from the SQL query.

Command buttons

- **Test**

Enables you to test the filter that you have defined.

- **Refresh**

Populates the **Input Parameters** table and the **Returned Attributes** table with changed values, if any.

- **Save**

Saves the filter and closes the dialog box.

- **Cancel**

Does not save the filter and closes the dialog box.

Edit Filter dialog box

The Edit Filter dialog box enables you to edit a filter.

- Properties tab
- Query tab

Properties tab

The **Properties** tab enables you to edit the name of the filter, select a dictionary object, and edit the description for the filter.

- **Name**

Enables you to edit the name of the filter.

- **Type**

Enables you to select a dictionary object from the list.

- **Entity Version**

Enables you to enter the version number for the filter in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to edit the description for the filter.

Query tab

Enables you to edit the SQL query, the label, and the description for the attributes.

- **SQL Query**

Enables you to edit the SQL query of the filter.

- **Input Parameters table**

Displays the list of parameters from the SQL query that you have entered which require user input when the filter is used. You can click the **Label** or the **Description** column of a parameter to edit the label or the description.

- **Returned Attributes table**

Displays the list of returned attributes from the SQL query.

Command buttons

- **Test**

Enables you to test the filter.

- **Refresh**

Populates the **Input Parameters** table and the **Returned Attributes** table with changed values, if any.

- **Save**

Saves the filter and closes the dialog box.

- **Cancel**

Does not save the changes, if any, and closes the dialog box.

Clone Filter dialog box

The Clone Filter dialog box enables you to copy a filter and edit the filter.

- Properties tab
- Query tab

Properties tab

The **Properties** tab enables you to edit the name of the filter, select a dictionary object, and edit the description for the filter.

- **Name**

Enables you to edit the name of the filter. The name of the filter you have selected to clone is used as the name for the clone and appended with `_copy` by default.

- **Dictionary type**

Enables you to select a dictionary object from the list.

- **Entity Version**

Enables you to enter the version number for the filter in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to edit the description for the filter.

Query tab

Enables you to edit the SQL query, the label, and the description for the attributes.

- **SQL Query**

Enables you to edit the SQL query of the filter.

- **Input Parameters table**

Displays the list of parameters from the SQL query that you have entered which require user input when the filter is used. You can click the **Label** or the **Description** column of a parameter to edit the label or the description.

- **Returned Attributes table**

Displays the list of returned attributes from the SQL query.

Command buttons

- **Test**

Enables you to test the filter.

- **Refresh**

Populates the **Input Parameters** table and the **Returned Attributes** table with changed values, if any.

- **Save**

Saves the filter and closes the dialog box.

- **Cancel**

Does not save the filter and closes the dialog box.

Commands window

The Commands window displays the available commands, in alphabetical order. You can access this window by selecting **Workflow Design > Commands**.



Depending on your role and account privileges, this window might not be displayed.

A command is defined as an action taken as part of a workflow.

- Commands table
- Toolbar

Commands table

The Commands table lists the available commands.

You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

- enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking clears and resets the filtering selections.
- on each column header enables you to filter based on the content of the columns. Clicking in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows (for ascending and for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The Commands table contains the following columns:

- **Certification**

Indicates whether the command is user-created () , PS () , community () , user-locked () , or NetApp-certified () .

- **Name**

Displays the name of the command.

- **Scheme**

Displays the scheme associated with commands. A scheme represents the data model for a system. For example, a vc scheme contains data relevant to your virtual environment, such as virtual machines, hosts, and datastores.

- **Entity Version**

Displays the version number of the object in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Displays the description of the command.

You can search for commands by entering the description in the **Search** filter text box.

- **OS Compatibility**

Indicates whether the command is compatible with the host operating system.

- **Last Updated On**

Displays the date and time when the command was last updated.

- **Updated By**

Displays the user who updated the command.

- **Locked By**

Displays the user who locked the command.

You can search for commands by entering the user name in the **Search** filter text box.

- **Command Language**

Displays the programming language in which the command is written (Perl or PowerShell).

- **Min. Software Versions**

Specifies the minimum versions of the software that is required to execute the workflow. For example, the software can be clustered Data ONTAP 8.2.0 and vCenter 6.0. The versions are displayed as comma-separated values.

- **Mandatory Parameters**

Displays the parameters selected as mandatory for the command.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Command Definition window, which enables you to create a command.

-  **(Edit)**

Opens the Edit Command Definition window for the selected command, which enables you to edit the command. You can also double-click the command to open the Edit Command Definition window.

-  **(Clone)**

Opens the Edit Command Definition <command_name> - copy window, which enables you to create a clone or copy of the selected command.

-  **(Lock)**

Opens the Lock Command confirmation dialog box, which enables you to lock the selected command. This option is enabled only for the commands you have created.

-  **(Unlock)**

Opens the Unlock Command confirmation dialog box, which enables you to unlock the selected command. This option is enabled only for the commands locked by you. However, administrators can unlock commands locked by other users.

-  **(Delete)**

Opens the Delete Command confirmation dialog box, which enables you to delete the selected user-created command.



You cannot delete a WFA or PS command.

-  **(Export)**

Enables you to export the selected user-created command.



You cannot export a WFA or PS command.

-  **(Test)**

Opens the Testing Command <CommandName> in <ScriptLanguage> dialog box, which enables you to test the selected command.

-  **(Add To Pack)**

Opens the Add To Pack Command dialog box, which enables you to add the command and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for commands for which the certification is set to **None**.

-  **(Remove From Pack)**

Opens the Remove From Pack Command dialog box for the selected command, which enables you to delete or remove the command from the pack.



The Remove From Pack feature is enabled only for commands for which the certification is set to **None**.

New Command Definition dialog box

The New Command Definition dialog box enables you to define a new command. You can create new commands for your workflows if the predefined commands do not meet

your requirements.

- Properties tab
- Code tab
- Parameters Definition tab
- Parameters Mapping tab
- Reservation tab
- Verification tab

Properties tab

Enables you to specify the properties of the command, such as the name, description, and entity version.

- **Name**

Enables you to specify a name for the command. You must specify a name in order to save the command.

- **Entity Version**

Enables you to specify the version number for the command in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to enter a description for the command.

- **String Representation**

Enables you to specify a string representation for the command using MVFLEX Expression Language (MVEL) syntax.

You must specify the string representation in order to save the command. The string representation is used to display the details of the command in the workflow design during planning and execution. You must use only the parameters of a command in the string representation for that command.

- **Timeout (s)**

Enables you to specify the timeout value (in seconds) for the command. The default value is 600 seconds.

- **Command Type**

Enables you to specify the type of execution for the command.

- **Standard execution**

Enables the command to be executed without any waiting period. Standard execution is selected by default.

- **Wait for condition**

Enables you to specify the time (in seconds) for which the command has to wait before execution. The default value is 60 seconds.

- **Minimum Software Versions**

Specifies the minimum versions of the software that are required for the command to operate. For example, the software can be clustered Data ONTAP 8.2.0 and vCenter 6.0. The versions are displayed as comma-separated values.

Code tab

Enables you to enter the code for the command in the selected scripting language. You can add a new scripting language by selecting the required language from the Script language drop-down list.

- **Discover Parameters**

Copies the parameters defined in the PowerShell code to the Parameters Definition table and the Parameters Mapping table.

Parameters Definition tab

Displays the parameters defined in the code that you entered in the Code tab.

- **Name**

Displays the name of the parameter.

- **Description**

Displays the description of the parameter.

- **Mandatory**

Displays a selected check box for the mandatory parameters.

- **Type**

Displays the type of the parameter, such as string, enum, array, or password.

- **Values**

Displays the value that is set for the parameter.

- **Add Parameter**

Enables you to add a parameter to the command if the selected scripting language is Perl.

- **Remove Parameter**

Enables you to remove a parameter from the command if the selected scripting language is Perl.

Parameters Mapping tab

Enables you to map the parameters to dictionary objects and to specify the attribute and object name.

- **Name**

Displays the name of the parameter.

- **Type**

Enables you to select a dictionary object for the parameter.

- **Attribute**

Enables you to specify the required attribute. You can either select the attribute (if available), or enter the attribute.

- **Object Name**

Enables you to specify a name for the dictionary object.

Reservation tab

Enables you to reserve the resources that are required by the command.

- **Reservation Script**

Enables you to enter an SQL query to reserve the resources that are required by the command. This helps ensure that the resources are available during a scheduled workflow execution.

- **Reservation Representation**

Enables you to specify a string representation for the reservation using MVEL syntax. The string representation is used to display the details of the reservation in the Reservations window.

Verification tab

Enables you to verify a reservation and remove the reservation after the command is executed.

- **Verification Script**

Enables you to enter an SQL query to verify the usage of the resources that were reserved by the reservation script. The verification script also verifies whether the WFA cache is updated, and removes the reservation after a cache acquisition.

- **Test Verification**

Opens the Verification dialog box, which enables you to test the parameters of the verification script.

Command buttons

- **Test**

Opens the Testing Command <CommandName> in <ScriptLanguage> dialog box, which enables you to test the command.

- **Save**

Saves the command and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Edit Command Definition dialog box

The **Edit Command Definition** dialog box enables you to edit the selected command.

- Properties tab
- Code tab
- Parameters Definition tab
- Parameters Mapping tab
- Reservation tab
- Verification tab

Properties tab

The Properties tab enables you to edit the properties of the command, such as the name, description, and scripting language.

- **Name**

Enables you to edit the name of the command. You must enter a name to save the command.

- **Entity Version**

Enables you to enter the version number for the command in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to edit the description for the command.

- **String Representation**

Enables you to specify a string representation for the command using MVEL syntax. You must specify the string representation to save the command.

- **Timeout (s)**

Enables you to specify the timeout value (in seconds) for the command. The default value is 600 seconds.

- **Command Type**

Enables you to specify the type of execution for the command.

- **Standard execution**

Enables the command to be executed without any waiting period. Standard execution is selected by default.

- **Wait for condition**

Enables you to specify the time (in seconds) for the command to wait before execution. The default value is 60 seconds.

- **Minimum Software Versions**

Specifies the minimum versions of the software that is required for the command to operate. For example, the software can be clustered Data ONTAP 8.2.0 and vCenter 6.0. The versions are displayed as comma-separated values.

Code tab

This tab enables you to edit the code for the command in Perl, PowerShell, or both. You can add a new scripting language by clicking  and then selecting the required language from the Script language drop-down list.

- **Test**

Opens the Verification dialog box, which enables you to test the parameters for the specified verification script.

- **Discover Parameters**

Copies the parameters defined in the PowerShell code to the Parameters Definition table and the Parameters Mapping table. The Discover Parameters button is disabled for the Perl codes.

Parameters Definition tab

This tab displays the parameters defined in the code that you have entered in the Code tab.

- **Name**

Displays the name of the parameter.

- **Description**

Displays the description of the parameter.

- **Mandatory**

Displays a selected check box for the mandatory parameters.

- **Type**

Displays the type of the parameter, such as string and enum.

- **Values**

Displays the value that is set for the parameter.

- **Add Parameter**

Enables you to add a parameter to the command if the selected scripting language is Perl.

- **Remove Parameter**

Enables you to remove a parameter from the command if the selected scripting language is Perl.

Parameters Mapping tab

This tab enables you to map the parameters to dictionary objects and to specify the attribute and object name.

- **Name**

Displays the name of the parameter.

- **Type**

Enables you to select the dictionary object for the parameter.

- **Attribute**

Enables you to specify the required attribute. You can select the attribute, if available, or enter the attribute.

- **Object Name**

Enables you to enter a name for the object.

Reservation tab

This tab enables you to reserve resources that are required by the command. For more information about reservations, see the *OnCommand Workflow Automation Workflow Developer's Guide*.

- **Reservation Script**

Enables you to enter an SQL query to reserve resources required by the command. This ensures that the resources are available during a scheduled workflow execution.

- **Reservation Representation**

Enables you to specify a string representation for the reservation using MVEL syntax. The string representation is used to display the details of the reservation in the Reservations window.

Verification tab

This tab enables you to verify a reservation and remove the reservation after the execution of the command is completed. For more information about verifying reservations, see the *OnCommand Workflow Automation Workflow Developer's Guide*.

- **Verification Script**

Enables you to enter an SQL query to verify the usage of resources that were reserved by the reservation script. It also verifies if the WFA cache is updated and removes the reservation after a cache acquisition.

Command buttons

- **Save**

Saves the changes and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Clone Command Definition dialog box

The Clone Command Definition dialog box enables you to copy a command and edit the cloned command.

- Properties tab
- Code tab
- Parameters Definition tab
- Parameters Mapping tab
- Reservation tab
- Verification tab

Properties tab

Enables you to edit the properties of the cloned command, such as the name, description, and entity version.

- **Name**

Enables you to edit the name of the cloned command. By default, the name of the command that you have selected to clone is used as the name for the clone, appended with “- copy”.

- **Entity Version**

Enables you to edit the version number of the command in major.minor.revision format—for example, 1.0.0.

- **Description**

Enables you to edit the description of the command.

- **String Representation**

Enables you to specify a string representation for the command using MVFLEX Expression Language (MVEL) syntax. You must specify the string representation in order to save the command.

- **Timeout (s)**

Enables you to specify the timeout value (in seconds) for the command. The default value is 600 seconds.

- **Command Type**

Enables you to specify the type of execution for the command.

- **Standard execution**

Enables the command to be executed without any waiting period. Standard execution is selected by default.

- **Wait for condition**

Enables you to specify the time (in seconds) for which the command has to wait before execution. The default value is 60 seconds.

- **Minimum Software Versions**

Specifies the minimum versions of the software that are required for the command to operate. For example, the software can be clustered Data ONTAP 8.2.0 and vCenter 6.0. The versions are displayed as comma-separated values.

- **Retain reservation and verification scripts from original command**

Enables you to retain the reservation and verification scripts that were originally specified for the command that you want to clone.

Code tab

Enables you to edit the code for the command.

- **Discover Parameters**

Copies the parameters defined in the PowerShell code to the Parameters Definition table and the Parameters Mapping table.

Parameters Definition tab

Displays the parameters defined in the code that you entered in the Code tab.

- **Name**

Displays the name of the parameter.

- **Description**

Displays the description of the parameter.

- **Mandatory**

Displays a selected check box for the mandatory parameters.

- **Type**

Displays the type of the parameter, such as string or enum.

- **Values**

Displays the values that are set for the parameter.

- **Add Parameter**

Enables you to add a parameter to the command if the selected scripting language is Perl.

- **Remove Parameter**

Enables you to remove a parameter from the command if the selected scripting language is Perl.

Parameters Mapping tab

Enables you to map the parameters to dictionary objects and to specify the attribute and object name.

- **Name**

Displays the name of the parameter.

- **Type**

Enables you to select a dictionary object for the parameter.

- **Attribute**

Enables you to specify the required attribute. You can either select the attribute (if available), or enter the name of the attribute.

- **Object Name**

Enables you to specify a name for the dictionary object.

Reservation tab

Enables you to reserve the resources that are required by the command.

- **Reservation Script**

Enables you to enter an SQL query to reserve the resources that are required by the command. This helps ensure that the required resources are available during a scheduled workflow execution.

- **Reservation Representation**

Enables you to specify a string representation for the reservation using MVEL syntax. The string representation is used to display the details of the reservation in the Reservations window.

Verification tab

Enables you to verify a reservation and remove the reservation after the command is executed.

- **Verification Script**

Enables you to enter an SQL query to verify the usage of the resources that were reserved by the reservation script. The verification script also verifies whether the WFA cache is updated, and removes the reservation after a cache acquisition.

- **Test Verification**

Opens the Verification dialog box, which enables you to test the parameters of the verification script.

Command buttons

- **Test**

Opens the Testing Command <CommandName> in <ScriptLanguage> dialog box, which enables you to test the command.

- **Save**

Saves the command and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Functions window

The Functions window displays the available OnCommand Workflow Automation (WFA) functions and enables you to manage these functions.

Functions are complementary tools for performing significant or black-boxed operations that are required for planning workflow executions. Functions are processed during the planning phase. Functions can be used to complete repetitive and complex tasks, such as defining a complex naming convention. Functions run in their own sandbox and might use other functions during execution. Functions are written in MVFLEX Expression Language (MVEL).

Functions table

The Functions table lists the available WFA functions. Each entry is identified as one of the following:

-  - content created by users
-  - content developed by Professional Services (PS), which is available only on custom installations made by PS
-  - packs developed by users
-  - content created by users that is locked
-  - NetApp-certified content

You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search

using supported operators for the applicable column type, Alphabetic or Numeric.

The Functions table contains the following columns:

- **Certification**

Indicates whether the function is user-created (👤), PS (⚙️), community (👤), user-locked (🔒), or NetApp-certified (🌐).

- **Name**

Displays the name of the function.

- **Entity Version**

Displays the version number of the object in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Displays the description of the functions.

- **Last Updated On**

Displays the date and time when the function was last updated.

- **Updated By**

Displays the user who updated the function.

- **Locked By**

Displays the user who locked the function.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Function window, which enables you to create a new function.

-  **(Edit)**

Opens the Function <EntryName> window for the selected function, which enables you to edit the function.

-  **(Clone)**

Opens the New Function <EntryName>_copy window, which enables you to create a copy of the selected function.

-  **(Lock)**

Opens the Lock the Function confirmation dialog box, which enables you to lock the selected function. This

option is enabled only for the functions that you have created.

-  **(Unlock)**

Opens the Unlock the Function confirmation dialog box, which enables you to unlock the selected function.

This option is enabled only for the functions that you have locked. Administrators can unlock functions that were locked by other users.

-  **(Delete)**

Opens the Delete Function confirmation dialog box, which enables you to delete the selected user-created function.



You cannot delete a WFA or PS function.

-  **(Export)**

Enables you to export the selected user-created function.



You cannot export a WFA or PS function.

-  **(Test)**

Opens the Test dialog box, which enables you to test the selected function.

-  **(Add To Pack)**

Opens the Add To Pack Functions dialog box, which enables you to add the function and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled for functions with certification set to None and Lock.

-  **(Remove From Pack)**

Opens the Remove From Pack Functions dialog box for the selected function, which enables you to delete or remove the function from the pack.



The Remove From Pack feature is enabled for functions with certification set to None and Lock.

Templates window

The Templates window displays the available templates in alphabetical order. A template is a collection of settings that can be used when creating workflows. You can use templates to create workflows quickly.



Depending on your role and account privileges, this window might not be displayed.

- Templates table

- Toolbar

Templates table

The Templates table lists the available templates. Each entry is identified as one of the following:

-  - content created by users
-  - content developed by Professional Services (PS), which is available only on custom installations made by PS
-  - packs developed by users
-  - content created by users that is locked
-  - NetApp-certified content

You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The Templates table contains the following columns:

- **Certification**

Indicates whether the template is user-created () , PS () , community () , user-locked () , or NetApp-certified () .

You can search for templates by selecting the required check boxes from the filter list.

- **Name**

Displays the name of the template.

- **Scheme**

Displays the scheme associated with the template. A scheme represents the data model for a system. For example, a vc scheme contains data relevant to your virtual environment, such as virtual machines, hosts, and datastores.

- **Type**

Displays the dictionary object type of the template—for example, volume and Snapshot_Policy.

- **Entity Version**

Displays the version number of the object in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Displays the description of the template.

- **Last Updated On**

Displays the date and time when the template was last updated.

- **Updated By**

Displays the name of the user who updated the template.

- **Locked By**

Displays the name of the user who locked the template.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Template dialog box, which enables you to create a new template.

-  **(Edit)**

Opens the Template <template_name> dialog box, which enables you to modify the settings of the selected template. You can also double-click the template to open the Template <template_name> dialog box.

-  **(Clone)**

Opens the New Template <template_name> - copy dialog box, which enables you to create a clone or copy of the selected template.

-  **(Lock)**

Opens the Lock Template confirmation dialog box, which enables you to lock the selected template. This option is enabled only for the templates that you have created.

-  **(Unlock)**

Opens the Unlock Template confirmation dialog box, which enables you to unlock the selected template. This option is enabled only for the templates that are locked by you. However, administrators can unlock templates that were locked by other users.

-  **(Delete)**

Opens the Delete Template confirmation dialog box, which enables you to delete the selected user-created template.



You cannot delete a sample template.

- **(Export)**

Enables you to export the selected user-created template.



You cannot export a sample template.

- **(Add To Pack)**

Opens the Add To Pack Templates dialog box, which enables you to add the template and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for templates for which the certification is set to **None**.

- **(Remove From Pack)**

Opens the Remove From Pack Templates dialog box for the selected template, which enables you to delete or remove the template.



The Remove From Pack feature is enabled only for templates for which the certification is set to **None**.

New Template dialog box

The New Template dialog box enables you to create a new template.

- **Name**

Enables you to enter a name for the template.

- **Type**

Enables you to select a dictionary object from the list.

- **Entity Version**

Enables you to enter the version number for the template in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to enter a description for the template.

- **Attributes**

Displays the attributes for the selected dictionary object and enables you to enter a value for each attribute, such as enum and function.

Command buttons

- **Save**

Saves the template and closes the dialog box.

- **Cancel**

Does not save the template and closes the dialog box.

Edit Template dialog box

The Edit Template dialog box enables you to edit a template.

- **Name**

Enables you to edit the name of the template.

- **Type**

Enables you to select a dictionary object from the list.

- **Entity Version**

Enables you to enter the version number for the template in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to edit the description for the template.

- **Attributes**

Displays the attributes for the selected dictionary object and enables you to enter a value for each attribute, such as enum and function.

Command buttons

- **Save**

Saves the changes and closes the dialog box.

- **Cancel**

Does not save the changes, if any, and closes the dialog box.

Clone Template dialog box

The Clone Template dialog box enables you to create a copy of the template and edit the template.

- **Name**

Enables you to edit the name of the template.

- **Type**

Enables you to select a dictionary object from the list. The name of the template you have selected to clone is used as the name for the clone and appended with - copy by default.

- **Entity Version**

Enables you to enter the version number for the template in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Enables you to edit the description for the template.

- **Attributes**

Displays the attributes for the selected dictionary object and enables you to enter a value for each attribute, such as enum and function.

Command buttons

- **Save**

Saves the template and closes the dialog box.

- **Cancel**

Does not save the template and closes the dialog box.

Schemes window

The Schemes window displays the available schemes and enables you to manage the schemes. OnCommand Workflow Automation (WFA) uses schemes to acquire data relevant to an environment.

A scheme represents the data model for a system. The data model is a collection of dictionary entries. You can define a scheme and then define a data source type. The data source defines how the data is acquired and the scheme is populated. For example, a `vc` scheme contains data relevant to your virtual environment, such as virtual machines, hosts, and datastores.

Schemes table

The Schemes table lists the available scheme entries. Each entry is identified as one of the following:

-  - content created by users
-  - content developed by Professional Services (PS), which is available only on custom installations made by PS
-  - packs developed by users
-  - content created by users that is locked

-  - NetApp-certified content

You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The Schemes table contains the following columns:

- **Certification**

Indicates whether the scheme is user-created () , PS () , community () , user-locked () , or NetApp-certified ().

- **Name**

Displays the name of the scheme.

You can search for a scheme by entering its name in the **Search** filter text box.

- **Display Name**

Displays the name of the scheme.

You can search for a scheme by entering its name in the **Search** filter text box.

- **Entity Version**

Displays the version number of the object in `major.minor.revision` format—for example, 1.0.0.

You can search for a scheme by entering the version number in the **Search** filter text box.

- **Description**

Displays the description of the scheme.

You can search for a scheme by entering its description in the **Search** filter text box.

- **Type**

Indicates whether the scheme is used to acquire data from a remote system or directly from workflows or commands. Possible values are Data Source Acquisition and Other.

You can search for schemes by entering the type in the **Search** filter text box.

- **Reset Flag**

Indicates whether the scheme is reset during the next data acquisition cycle. Possible values are true and false.

- **Last Updated On**

Displays the date and time when the scheme was last updated.

You can search for schemes by selecting the required time category from the filter drop-down list.

- **Updated By**

Displays the name of the user who updated the scheme.

You can search for schemes by entering the user name in the **Search** filter text box.

- **Locked By**

Displays the name of the user who locked the scheme.

You can search for schemes by entering the user name in the **Search** filter text box.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Schemes dialog box, which enables you to create a new scheme.

-  **(Edit)**

Opens the Scheme <SchemeName> dialog box for the selected scheme, which enables you to edit the scheme.

-  **(Lock)**

Opens the Lock the Scheme confirmation dialog box, which enables you to lock the selected scheme. This option is enabled only for the schemes that you created.

-  **(Unlock)**

Opens the Unlock the Scheme confirmation dialog box, which enables you to unlock the selected scheme. This option is enabled only for the schemes that you locked. Administrators can unlock schemes that are locked by other users.

-  **(Delete)**

Opens the Delete Scheme confirmation dialog box, which enables you to delete the selected user-created schemes.



You cannot delete a WFA or PS scheme.

-  **(Export)**

Enables you to export the selected user-created scheme.



You cannot export a WFA or PS scheme.

-  **(Reset Scheme)**

Enables you to reset the scheme during the next data acquisition cycle.

-  **(Add To Pack)**

Opens the Add To Pack Schemes dialog box, which enables you to add the scheme and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for schemes for which the certification is set to **None**.

-  **(Remove From Pack)**

Opens the Remove From Pack Schemes dialog box for the selected scheme, which enables you to delete or remove the scheme from the pack.



The Remove From Pack feature is enabled only for schemes for which the certification is set to **None**.

Dictionary window

The Dictionary window displays the available dictionary entries in alphabetical order.

A dictionary entry is the definition of an object type that is supported by OnCommand Workflow Automation (WFA). Each dictionary entry represents an object type and its relationships in the storage and storage-related environments. Dictionary entries are converted to tables in the databases.



Depending on your role and account privileges, this window might not be displayed.

Dictionary table

The Dictionary table lists the available dictionary entries. Each entry is identified as one of the following:

-  - content created by users
-  - content developed by Professional Services (PS), which is available only on custom installations made by PS
-  - packs developed by users
-  - content created by users that is locked
-  - NetApp-certified content

You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The Dictionary table contains the following columns:

- **Certification**

Indicates whether the dictionary entry is user-created () , PS () , community () , user-locked () , or NetApp-certified ().

- **Name**

Displays the name of the dictionary entry.

- **Scheme**

Displays the scheme associated with the dictionary entries. A scheme is a collection of dictionary entries that contain data relevant to your environment. (For example, a vc scheme contains data relevant to your virtual environment, such as virtual machines, hosts, and datastores.)

- **Entity Version**

Displays the version number of the object in `major.minor.revision` format—for example, 1.0.0.

- **Description**

Displays the description of the dictionary entry.

- **Acquisition Enabled**

Displays a check mark for the dictionary entries that are enabled for data acquisition.

- **Last Updated On**

Displays the date and time when the dictionary entry was last updated.

- **Updated By**

Displays the name of the user who updated the dictionary entry.

- **Locked By**

Displays the name of the user who locked the dictionary entry.

- **Natural Keys**

Displays the natural keys associated with the dictionary entries.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Dictionary Entry window, which enables you to create a new dictionary entry.

-  **(Edit)**

Opens the Dictionary Entry <EntryName> window for the selected dictionary entry, which enables you to edit the dictionary entry.

-  **(Clone)**

Opens the New Dictionary Entry <EntryName>_copy window, which enables you to create a copy of the selected dictionary entry.

-  **(Lock)**

Opens the Lock the Dictionary Entry confirmation dialog box, which enables you to lock the selected dictionary entry.

This option is enabled only for the dictionary entries that you have created.

-  **(Unlock)**

Opens the Unlock the Dictionary Entry confirmation dialog box, which enables you to unlock the selected dictionary entry.

This option is enabled only for the dictionary entries that you have locked. Administrators can unlock dictionary entries that were locked by other users.

-  **(Delete)**

Opens the Delete Dictionary Entry confirmation dialog box, which enables you to delete the selected user-created dictionary entry.



You cannot delete a WFA dictionary entry or PS dictionary entry.

-  **(Export)**

Enables you to export the selected user-created dictionary entry.



You cannot export a WFA dictionary entry or PS dictionary entry.

-  **(Enable Acquisition)**

Provides the option to enable cache acquisition for the selected dictionary entry.

-  **(Disable Acquisition)**

Enables you to disable cache acquisition for the selected dictionary entry.

-  **(Reset Scheme)**

Enables you to reset the scheme associated with the selected dictionary entry.

-  **(Add To Pack)**

Opens the Add To Pack Dictionary dialog box, which enables you to add the dictionary entry and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for the dictionary entries for which the certification is set to **None**.

-  **(Remove From Pack)**

Opens the Remove From Pack Dictionary dialog box for the selected dictionary entry, which enables you to delete or remove the dictionary entry from the pack.



The Remove From Pack feature is enabled only for dictionary entries for which the certification is set to **None**.

-  **(Inventory)**

Opens the Inventory dialog box for the selected dictionary entry, which enables you to see the table data.

New Dictionary Entry dialog box

The New Dictionary Entry dialog box enables you to create a new dictionary object and specify a definition for the object.

- **Name of object type**

Enables you to specify a name for the dictionary object.

- **Description**

Enables you to provide a description for the dictionary object.

- **Scheme**

Enables you to select a scheme that should be associated with the object. You can add custom dictionary entries to the certified schemes.

- **Entity Version**

Enables you to enter the version number for the dictionary entry in `major.minor.revision` format—for example, 1.0.0.

Attribute table

Enables you to specify a list of attributes that define the dictionary object.

- **Name**

Enables you to enter a name for the attribute.

- **Type**

Enables you to select the type of the attribute—such as string, boolean, and integer—for type checking. You can also reference a dictionary entry from the selected scheme.

- **String length**

Enables you to specify the length of the string in numerals, if string type is selected for the attribute.

- **Values**

Enables you to specify a list of comma-separated values, if enum type is selected for the attribute.

- **Natural key**

Enables you to specify whether the dictionary attribute is part of the dictionary object's natural key.

A natural key is the unique identifier of a dictionary object. For example, qtrees are identified by the qtree name, the volume name, and the array IP address.

- **To be Cached**

Enables you to specify whether the attribute should be cached.

A dictionary entry can include both attributes that are cached and attributes that are not cached. A cache table is created and populated with the attributes that are marked for caching during the cache acquisition of the scheme. A cache table is created for a dictionary object when at least one attribute is selected to be cached.

- **Can be Null**

Enables you to specify whether the value of the attribute can be null. This setting is valid only for attributes that are selected to be cached.

- **Description**

Enables you to specify a description for the attribute.

- **Values in natural key columns are case sensitive**

Enables you to specify that the natural keys are case-sensitive.

- **Command buttons**

- **Add row**

Enables you to add a row to the table that is used to specify an attribute.

- **Remove row**

Enables you to delete the selected row.

Edit Dictionary Entry dialog box

The Dictionary Entry <EntryName> dialog box enables you to edit a dictionary object and its definition.

- **Name of object type**

Enables you to specify a name for the dictionary object.

- **Description**

Enables you to provide a description for the dictionary object.

- **Scheme**

You must not edit the scheme of a dictionary object. If you change the scheme, saving the dictionary entry fails. You can clone the dictionary entry to edit its scheme.

- **Entity Version**

Enables you to enter the version number for the dictionary entry in `major.minor.revision` format—for example, 1.0.0.

Attribute table

Enables you to specify a list of attributes that define the dictionary object.

- **Name**

Enables you to enter the name of the attribute.

- **Type**

Enables you to select the type of the attribute—such as string, boolean, and integer—for type checking. You can also reference a dictionary entry from the selected scheme.

- **String length**

Enables you to specify the length of the string in numerals, if string type is selected for the attribute.

- **Values**

Enables you to specify a list of comma-separated values, if enum type is selected for the attribute.

- **Natural key**

Enables you to specify whether the dictionary attribute is part of the dictionary object's natural key.

A natural key is the unique identifier of a dictionary object. For example, qtrees are identified by the qtree name, the volume name, and the array IP address.

- **To be Cached**

Enables you to specify whether the attribute should be cached.

A dictionary entry can include both attributes that are cached and attributes that are not cached. A cache table is created and populated with the attributes that are marked for caching during the cache acquisition of the scheme. A cache table is created for a dictionary object when at least one attribute is selected to be cached.

- **Can be Null**

Enables you to specify whether the value of the attribute can be null. This setting is valid only for attributes that are selected to be cached.

- **Description**

Enables you to specify a description for the attribute.

- **Values in natural key columns are case sensitive**

Enables you to specify that the natural keys are case-sensitive.

- **Command buttons**

- **Add row**

Enables you to add a row to the table that is used to specify an attribute.

- **Remove row**

Enables you to delete the selected row.

Clone Dictionary Entry dialog box

The New Dictionary Entry <EntryName_copy> dialog box enables you to copy a dictionary object and edit the definition of the cloned dictionary object.

- **Name of object type**

Enables you to specify a name for the cloned dictionary object.

- **Description**

Enables you to provide a description for the cloned dictionary object.

- **Scheme**

Enables you to select a scheme that should be associated with the cloned dictionary object. You can add custom dictionary entries to the certified schemes.

- **Entity Version**

Enables you to enter the version number for the cloned dictionary entry in `major.minor.revision` format—for example, 1.0.0.

Attribute table

Enables you to specify a list of attributes that define the cloned dictionary object.

- **Name**

Enables you to enter the name of the attribute.

- **Type**

Enables you to select the type of the attribute—such as string, boolean, and integer—for type checking. You can also reference a dictionary entry from the selected scheme.

- **String length**

Enables you to specify the length of the string in numerals, if string type is selected for the attribute.

- **Values**

Enables you to specify a list of comma-separated values, if enum type is selected for the attribute.

- **Natural key**

Enables you to specify whether the dictionary attribute is part of the dictionary object's natural key.

A natural key is the unique identifier of a dictionary object. For example, qtrees are identified by the qtree name, the volume name, and the array IP address.

- **To be Cached**

Enables you to specify whether the attribute should be cached.

A dictionary entry can include both attributes that are cached and attributes that are not cached. A cache table is created and populated with the attributes that are marked for caching during the cache acquisition of the scheme. A cache table is created for a dictionary object when at least one attribute is selected to be cached.

- **Can be Null**

Enables you to specify whether the value of the attribute can be null. This setting is valid only for attributes that are selected to be cached.

- **Description**

Enables you to specify a description for the attribute.

- **Values in natural key columns are case sensitive**

Enables you to specify that the natural keys are case-sensitive.

- **Command buttons**

- **Add row**

Enables you to add a row to the table that is used to specify an attribute.

- **Remove row**

Enables you to delete the selected row.

Data Source Types window

The Data Source Types window displays the available data source types. You can manage the data source types from this window.

A data source type is used to define a data source, which contains information about where the data in a specific database originates from and the method used to acquire data from the data source. For example, a data source can be either an Active IQ Unified Manager database that contains information about a storage environment or a VMware database with information about a data center.

Data Source Types table

The Data Source Types table lists the available data source types. Each entry is identified as one of the following:

-  - content created by users
-  - content developed by Professional Services (PS), which is available only on custom installations made by PS
-  - packs developed by users
-  - content created by users that is locked
-  - NetApp-certified content

You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The Data Source Types table contains the following columns:

- **Certification**

Indicates whether the data source type is user-created () , PS () , community () , user-locked () , or NetApp-certified ().

- **Data Source**

Displays the name of the data source type.

- **Scheme**

Displays the scheme associated with the data source type. A scheme represents the data model for a system. For example, a vc scheme contains data relevant to your virtual environment, such as virtual machines, hosts, and datastores.

The following are the default schemes that are selected by default:

- cm_performance
- cm_storage
- performance
- storage
- vc

- **Entity Version**

Displays the version number of the object in `major.minor.revision` format—for example, 1.0.0.

- **OS Compatibility**

Indicates whether the data source type is compatible with the host operating system.

- **Last Updated On**

Displays the date and time when the data source type was last updated.

- **Updated By**

Displays the user who updated the data source type.

- **Data Source Version**

Displays the version of the data source that is associated with the data source type.

- **Data Source Driver**

Displays the driver type that is used to acquire data from the data source.

- **Method**

Displays the method that is used to acquire data from the data source, such as SQL and script.

- **Script Language**

Displays the scripting language that is used by the data source type.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Data Source Type window, which enables you to create a new data source type.

-  **(Edit)**

Opens the Data Source Type <EntryName> window for the selected data source type, which enables you to edit the data source type.

-  **(Clone)**

Opens the New Data Source Type <EntryName> window, which enables you to create a copy of the selected data source type.

-  **(Lock)**

Opens the Lock the Data Source Type confirmation dialog box, which enables you to lock the selected user-created source data type.

-  **(Unlock)**

Opens the Unlock the Data Source Type confirmation dialog box, which enables you to unlock the selected data source type. This option is enabled only for the data source types that you locked. Administrators can unlock data source types that are locked by other users.

-  **(Delete)**

Opens the Delete Data Source Type confirmation dialog box, which enables you to delete the selected user-created data source type.



You cannot delete a WFA or PS data source type.

-  **(Export)**

Enables you to export the selected user-created data source type.



You cannot export a WFA or PS data source type.

-  **(Add To Pack)**

Opens the Add To Pack Data Source Type dialog box, which enables you to add the data source type and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for data source types for which the certification is set to None.

-  **(Remove From Pack)**

Opens the Remove From Pack Data Source dialog box for the selected data source type, which enables you to delete or remove the data source type from the pack.



The Remove From Pack feature is enabled only for data sources types for which the certification is set to None.

Remote System Types window

The Remote System Types window displays the type of remote systems that OnCommand Workflow Automation (WFA) can work with. The remote systems include clustered Data ONTAP, Cloud Manager, Active IQ Unified Manager, DataFabric Manager server, and E-Series systems.

- Remote System Types table
- Toolbar

Remote System Types table

The remote system types table lists in tabular format the remote systems that WFA can work with. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The remote system types table contains the following columns:

- **Name**

Displays the name of the remote system type.

You can search for a remote system type by entering its name in the **Search** filter text box.

- **Entity Version**

Displays the current version of the remote system type in `major.minor.revision` format—for example, 1.0.0.

You can search for a remote system type by entering the version number in the **Search** filter text box.

- **Description**

Displays the description of the remote system type.

You can search for a remote system type by entering its description in the **Search** filter text box.

- **Last Updated On**

Displays the date and time when the remote system type was last updated.

You can search for remote system types by selecting the required time category from the filter drop-down list.

- **Updated By**

Displays the name of the user who updated the remote system type.

You can search for remote system types by entering the user name in the **Search** filter text box.

- **Locked By**

Displays the name of the user who locked the remote system type.

You can search for remote system types by entering the user name in the **Search** filter text box.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Remote System Type dialog box, which enables you to enter the details to add a new remote system type to WFA.

-  **(Edit)**

Opens the Remote System Type <RemoteSystemTypeName> dialog box for the selected remote system type, which enables you to edit the remote system type.

-  **(Clone)**

Opens the New Remote System Type <RemoteSystemTypeName> - copy dialog box, which enables you to create a clone or copy of the selected remote system type.

-  **(Lock)**

Opens the Lock Remote System Type confirmation dialog box, which enables you to lock the selected remote system type. This option is enabled only for the remote system types that you created.

-  **(Unlock)**

Opens the Unlock Remote System Type confirmation dialog box, which enables you to unlock the selected remote system type. This option is enabled only for the remote system types that you locked. Administrators can unlock remote system types that are locked by other users.

-  **(Delete)**

Opens the Delete Remote System Type confirmation dialog box, which enables you to delete the selected remote system type.

-  **(Export)**

Enables you to export the selected remote system type.

-  **(Add To Pack)**

Opens the Add To Pack Remote System Types dialog box, which enables you to add the remote system type and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for remote system types for which the certification is set to **None**.

-  **(Remove From Pack)**

Opens the Remove From Pack Remote System Types dialog box for the selected remote system type, which enables you to delete or remove the remote system type from the pack.



The Remove From Pack feature is enabled only for remote system types for which the certification is set to **None**.

New Remote System Type dialog box

The New Remote System Type dialog box enables you to add a new remote system type to OnCommand Workflow Automation (WFA) if a predefined system type does not meet your requirement or if you want to change the configuration of a predefined system type.

- Details tab
- Validation Script tab

Details tab

Enables you to specify the details of the remote system type, such as the name, description, version, and connection protocol.

- **Name**

Enables you to specify a name for the remote system type. You must specify a name in order to save the remote system type.

- **Description**

Enables you to enter a description for the remote system type.

- **Version**

Enables you to specify the version number for the remote system type in `major.minor.revision` format—for example, 1.0.0.

- **Connection Protocol**

Enables you to select one of the following protocols that should be used by WFA when connecting to the remote system:

- HTTPS with fallback to HTTP

The connection primarily uses HTTPS. If connection through HTTPS fails, HTTP is used. If connection

through HTTP also fails, the connection attempt is dropped.

- HTTPS Only
- HTTP Only
- Custom

When you select a connection protocol, the protocol, default port, and default timeout (sec) fields are populated with data.

Validation Script tab

Enables you to test the connectivity of the selected protocol with the remote system type.

You can click **Test Perl Script** to run the script for testing the connectivity.

Command buttons

- **Save**

Saves the configuration settings of the remote system type and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Edit Remote System Type dialog box

The Edit Remote System Type dialog box enables you to modify the name, description, version, protocol, default port, and default timeout of an existing remote system type. You cannot modify a NetApp-certified remote system type.

- Details tab
- Validation Script tab

Details tab

Enables you to edit the details of the remote system type, such as the name, description, version, and connection protocol.

- **Name**

Enables you to edit the name of the remote system type.

- **Description**

Enables you to modify the description of the remote system type.

- **Version**

Enables you to edit the version number of the remote system type in `major.minor.revision` format—for example, 1.0.0.

- **Connection Protocol**

Enables you to select one of the following protocols that should be used by OnCommand Workflow Automation (WFA) when connecting to the remote system:

- HTTPS with fallback to HTTP

The connection primarily uses HTTPS. If connection through HTTPS fails, HTTP is used. If connection through HTTP also fails, the connection attempt is dropped.

- HTTPS Only
- HTTP Only
- Custom

When you select a connection protocol, the protocol, default port, and default timeout (sec) fields are populated with data.

Validation Script tab

Enables you to provide a validation script to test the connectivity of the remote system. The validation script must be written in Perl and must include the following function: `checkCredentials ($host, $user, $password, $protocol, $port, $timeout)`. During validation, WFA calls the `checkCredentials` function with the values that are configured for the remote system type and the connection protocol. The function returns the value “1” on successful validation of connectivity or “0” on failure of connection validation.

You can click **Test Perl Script** to run the script for testing the connectivity.

Command buttons

- **Save**

Saves the modified configuration settings of the remote system type and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Cache Queries window

The Cache Queries window enables you to manage cache queries for the WFA dictionary entries and their associated data source types.

A cache query is an SQL query that retrieves the required data from the table specified in the query. A cache query is associated with a dictionary entry and one or more data source types. You can define a cache query when you want retrieve some information, such as volume from a table of a data source OnCommand Unified Manager 6.0 into the WFA cache.

- Cache Queries table
- Toolbar

Cache Queries list

The Cache Queries table lists the dictionary entries and their associated data source types. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows ( for ascending and  for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The Cache Queries table contains the following columns:

- **Certification**

Indicates whether the cache query is user-created () , PS () , community () , user-locked () , or NetApp-certified ().

- **Scheme**

Indicates a scheme name (schema) that contains data that is relevant to your environment. For example, a **cm_storage** caching scheme contains data relevant to clustered Data ONTAP. The relevant scheme information is obtained from the data source.

- **Dictionary Entry**

Displays the dictionary entry associated with the scheme.

- **Entity Version**

Displays the version number of the object in `major.minor.revision` format—for example, 1.0.0.

- **Data Source Types**

Displays the data source type associated with the dictionary entry.

- **Last Updated On**

Displays the date and time when the cache query was last updated.

- **Updated By**

Displays the user who updated the cache query.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the Add Cache Query dialog box, which enables you to create a cache query.

-  **(Edit)**

Opens the Edit Cache Query dialog box for the selected cache query, which enables you to edit the cache query.

-  **(Clone)**

Opens the Add Cache Query dialog box, which enables you to create a clone or copy of the selected cache query.

-  **(Lock)**

Opens the confirmation dialog box, which enables you to lock the selected cache query.

-  **(Unlock)**

Opens the confirmation dialog box, which enables you to unlock the selected cache query. This option is enabled only for the cache queries that are locked by you. However, administrators can unlock cache queries locked by other users.

-  **(Delete)**

Opens the confirmation dialog box, which enables you to delete the selected user-created cache query.



You cannot delete a WFA, PS, or sample cache queries.

-  **(Export)**

Opens the confirmation dialog box, which enables you to export the selected user-created cache query.



You cannot export a WFA, PS, or sample cache queries.

-  **(Test)**

Opens the Test Cache Query dialog box, which enables you to test the selected cache query.

-  **(Add To Pack)**

Opens the Add To Pack Cache Query dialog box, which enables you to add the cache query and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for the cache query for which the certification is set to **None**.

-  **(Remove From Pack)**

Opens the Remove From Pack Cache Query dialog box for the selected cache query, which enables you to delete or remove the cache query from the pack.



The Remove From Pack feature is enabled only for cache query for which the certification is set to **None**.

Add Cache Query dialog box

The Add Cache Query dialog box enables you to create a new cache query for a dictionary entry and associate the query with a data provide type, such as a specific Active IQ Unified Manager version.

- **Dictionary Entry**

Enables you to select a dictionary entry for which you want to create a cache query.

- **Entity Version**

Enables you to enter the version number for the cache query in `major.minor.revision` format—for example, 1.0.0.

- **Data Source Types**

Enables you to select the data source types you want to associate with the cache query—for example, Active IQ Unified Manager_6.0.

Table Structure

- **Attributes tab**

Displays the attributes associated with the dictionary entry.

- **Create Table SQL tab**

Displays the Create Table script for that dictionary entry.

SQL select query

Enables you to enter the SQL select query that retrieves data from the table of the specified data provider.

Command buttons

- **Test**

Enables you to test the SQL query entered in the **SQL select query** field.

- **Save**

Saves the cache query and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Edit Cache Query dialog box

The Edit Cache Query dialog box enables you to edit a cache query associated with a dictionary entry.

- **Dictionary Entry**

Specifies a dictionary entry associated with the cache query.

- **Entity Version**

Enables you to enter the version number for the cache query in `major.minor.revision` format—for example, 1.0.0.

- **Data Source Types**

Specifies the data source type associated with the cache query.

Table Structure

Displays the attributes and the SQL syntax associated with the dictionary entry.

- **SQL select query**

Enables you to edit the SQL query associated with the dictionary entry and the selected data provider type.

Command buttons

- **Test**

Enables you to test the SQL query entered in the **SQL select query** field.

- **Save**

Saves the cache query and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Clone Cache Query dialog box

The Edit Cache Query dialog box enables you to edit a cache query associated with a dictionary entry.

- **Dictionary Entry**

Specifies a dictionary entry associated with the cache query.

- **Entity Version**

Enables you to enter the version number for the cache query in `major.minor.revision` format—for example, 1.0.0.

- **Data Source Types**

Specifies the data source type associated with the cache query.

Table Structure

Displays the attributes and the SQL syntax associated with the dictionary entry.

- **SQL select query**

Specifies the SQL query associated with the dictionary entry and the selected data provider type.

Command buttons

- **Test**

Enables you to test the SQL query entered in the SQL select query field.

- **Save**

Saves the cache query as a new entry in the Cache Queries table and closes the dialog box.

- **Cancel**

Cancels changes, if any, and closes the dialog box.

Packs window

The Packs window displays the OnCommand Workflow Automation WFA packs that are imported and available in the WFA server. Each pack contains the pack information file and WFA content, such as workflows, commands, filters, functions, finders, and templates.

- **Packs table**
- **Toolbar**

Packs table

The packs table lists in tabular format the WFA packs that are available in the WFA server. You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

-  enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking  clears and resets the filtering selections.
-  on each column header enables you to filter based on the content of the columns. Clicking  in a

column allows you to filter on a specific item from the drop-down list or on all available items.

- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows (▲ for ascending and ▼ for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The packs table contains the following columns:

- **Certification**

Indicates whether the pack is user-created (👤), PS (✳), community (👤), user-locked (🔒), or NetApp-certified (✳)

You can search for packs by selecting the required option check boxes from the filter list.

- **Name**

Displays the name of the pack.

You can search for a pack by entering its name in the **Search** filter text box.

- **Description**

Displays the description of the pack.

You can search for a pack by entering its description in the **Search** filter text box.

- **Entity Version**

Displays the version number of the pack in `major.minor.revision` format—for example, 1.0.0.

- **Last Updated On**

Displays the date and time when the pack was updated.

You can search for packs by selecting the required time category from the filter drop-down list.

- **Updated By**

Displays the name of the user who updated the pack.

You can search for a pack by entering the user name in the **Search** filter text box.

- **Details**

Displays the details of the pack in the Storage Automation Store website.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Pack dialog box, which enables you to create a new pack.

-  **(Edit)**

Opens the Pack Contents dialog box, which contains the following tabs for detailed information about the contents of the packs:

- Workflows
- Finders
- Filters
- Commands
- Functions
- Templates
- Dictionary
- Schemes
- Cache Queries
- SQL Data Source Types
- Script Data Source Types
- Remote System Types
- Categories

-  **(Delete)**

Opens the Delete Pack confirmation dialog box, which enables you to delete the selected pack.

-  **(Unlock)**

Opens the Unlock Pack confirmation dialog box, which enables you to unlock the selected pack. This option is enabled only for the packs locked by you. However, administrators can unlock packs locked by other users.

-  **(Export)**

Opens the Export dialog box, which enables you to export the selected pack.

-  **(Import From Server Folder)**

Opens the Import From Server Folder dialog box, which enables you to import the pack from the selected folder location in the server system.

-  **(Export To Server Folder)**

Opens the Export To Server Folder dialog box, which enables you to export the pack to the selected folder location in the server system.

New Pack dialog box

The New Pack dialog box enables you to create a new pack.

- **Name**

Enables you to enter a name and save the pack.

- **Version**

Enables you to enter a version and save the pack.

- **Author**

Enables you to enter author's name and save the pack.

- **Description**

Enables you to enter a description and save the pack.

Command buttons

- **Save**

Saves the pack and closes the dialog box.

- **Cancel**

Closes the dialog box without saving.

Edit Pack dialog box

The Edit Pack dialog box enables you to edit the pack.

- Details tab
- Pack Contents tab

Details tab

- **Name**

Enables you to enter a name and save the pack.

- **Version**

Enables you to enter a version and save the pack.

- **Author**

Enables you to enter author's name and save the pack.

- **Description**

Enables you to enter a description and save the pack.

Pack Contents tab

- **Workflow**

The **Workflow** option enables you to view the **Name** and **Entity Version** of the workflow.

- **Finders**

The **Finders** option enables you to view the **Name** and **Entity Version** of the finder.

- **Filters**

The **Filters** option enables you to view the **Name** and **Entity Version** of the filter.

- **Commands**

The **Commands** option enables you to view the **Name** and **Entity Version** of the command.

- **Functions**

The **Functions** option enables you to view the **Name** and **Entity Version** of the function.

- **Templates**

The **Templates** option enables you to view the **Name** and **Entity Version** of the template.

- **Dictionary**

The **Dictionary** option enables you to view the **Name** and **Entity Version** of the dictionary.

- **Schemes**

The **Schemes** option enables you to view the **Name** and **Entity Version** of the scheme.

- **Cache Queries**

The **Cache Queries** option enables you to view the **Name** and **Entity Version** of the cache query.

- **SQL Data Source Types**

The **SQL Data Source Types** option enables you to view the **Name** and **Entity Version** of the SQL data source type.

- **Script Data Source Types**

The **Script Data Source Types** option enables you to view the **Name** and **Entity Version** of the script data source type.

- **Categories**

The **Categories** option enables you to view the **Name** and **Entity Version** of the category.

- **Remote System Types**

The **Remote System Types** option enables you to view the **Name** and **Entity Version** of the remote system type.

Command buttons

- **Save**

Saves the pack and closes the dialog box.

- **Cancel**

Closes the dialog box without saving.

Categories window

The Categories window enables you to manage the workflow categories.



Depending on your role and account privileges, this window might not be displayed.

- Categories table
- Toolbar

A category is a set of workflows that enables you to complete a task. You can create a category by grouping a set of related workflows. You can also grant access to specific users to operate a category.

Categories table

The Categories table lists the workflow categories. The categories are identified by one of the following:

- - content created by users
- - content developed by Professional Services (PS), which is available only on custom installations made by PS
- - packs developed by users
- - content created by users that is locked
- - NetApp-certified content

You can customize the table display by using the filtering and sorting features available for each column, as well as by rearranging the column order.

- enables or disables filtering for the entire table. A red "x" appears over the icon if filtering is disabled.
- Double-clicking clears and resets the filtering selections.
- on each column header enables you to filter based on the content of the columns. Clicking in a column allows you to filter on a specific item from the drop-down list or on all available items.
- Clicking the column header toggles between ascending and descending order of sorting. You can identify the applied sort order by the sort arrows (for ascending and for descending).
- To rearrange the location of columns, you can drag and drop columns to place them in any required order. However, you cannot hide or delete any of these columns.
- Clicking the **Search** filter text box allows you to search for specific content. In addition, you can search using supported operators for the applicable column type, Alphabetic or Numeric.

The Categories table contains the following columns:

- **Certification**

Indicates whether the category is user-created () , PS () , community () , user-locked () , or NetApp-certified () .

- **Name**

Displays the name of the category.

- **Description**

Displays a description of the category.

- **Workflows**

Displays the workflows that are available in the category.

- **Used for workflow authorization**

- Displays “true” for the categories that are restricted to certain users with operator role.
- Displays “false” for the categories that are available for all users with operator role.

- **Users**

Displays the name of the user with approver or operator role who is authorized to execute the category.

- **Active Directory Group**

Indicates the name of the Active Directory group who is authorized to execute the category.

You can search for a group by entering the group name in the **Search** filter text box.

- **Last Updated On**

Displays the date and time when the category was last updated.

- **Updated By**

Displays the name of the user who updated the category.

Toolbar

The toolbar is located above the column header. You can use the icons in the toolbar to perform various actions. These actions can also be accessed from the right-click menu in the window.

-  **(New)**

Opens the New Category dialog box, which enables you to create a new category.

-  **(Edit)**

Opens the Category <category_name> dialog box, which enables you to edit the selected category. You can also double-click the category to open the Category <category_name> dialog box.

-  **(Clone)**

Opens the New Category <category_name> - copy dialog box, which enables you to create a clone or copy of the selected category.

-  **(Delete)**

Opens the Delete Category confirmation dialog box, which enables you to delete the selected category.

-  **(Export)**

Enables you to export the selected category.

-  **(Add To Pack)**

Opens the Add To Pack Categories dialog box, which enables you to add the category and its dependable entities to a pack, which is editable.



The Add To Pack feature is enabled only for categories for which the certification is set to **None**.

-  **(Remove From Pack)**

Opens the Remove From Pack Categories dialog box for the selected category, which enables you to delete or remove the category from the pack.



The Remove From Pack feature is enabled only for categories for which the certification is set to **None**.

New Category dialog box

The New Category dialog box enables you to create a new workflow category.

- **Name**

Enables you to enter a name for the category. You must enter a name to save the category.

- **Description**

Enables you to enter a description for the category.

- **Available Workflows**

Displays all the available workflows that are not moved to the **Selected Workflows** box.

- **Selected Workflows**

Displays all the workflows that are selected for the category.

- **Category Used for Workflow Authorization**

Enables you to select users with operator role who can execute the category. By default, the check box is not selected and the category is available for all users.

- **Restrict access to workflows in this category to the following users and Active Directory groups**

Enables you to restrict access to workflows in the selected category to selected users and Active Directory groups.

- **Available Approvers and Operators**

Displays all users with approver and operator roles.

- **Selected Approvers and Operators**

Displays the users who are allowed to execute the category.

- **Available Approver and Operator Groups**

Displays all groups with approver and operator roles.

- **Selected Approver and Operator Groups**

Displays the groups that are allowed to execute the category.

Command buttons

- **Selection buttons**

Enables you to move the selected entry from one box to another.

- **Save**

Saves the category and closes the dialog box.

- **Cancel**

Does not save the category and closes the dialog box.

Edit Category dialog box

The Edit Category dialog box enables you to edit a workflow category.

- **Name**

Enables you to edit the name of the category.

- **Description**

Enables you to edit the description for the category.

- **Available Workflows**

Displays all the available workflows that are not moved to the Selected Workflows box.

- **Selected Workflows**

Displays all the workflows that are selected for the category.

- **Category Used for Workflow Authorization**

Enables you to grant access to users with operator role to execute the category. The check box is not selected by default.

- **Available Operators**

Displays all the users with operator role who do not have access to the category.

- **Selected Operators**

Displays the users who are granted access to the category.

Command buttons

- **Selection buttons**

Moves the selected entry from one box to another.

- **Save**

Saves the category and closes the dialog box.

- **Cancel**

Does not save the category and closes the dialog box.

Clone Category dialog box

The Clone Category dialog box enables you to copy a workflow category and edit the category.

- **Name**

Enables you to edit the name for the category. The name of the category you have selected to clone is used as the name for the clone and appended with - copy by default.

- **Description**

Enables you to enter a description for the category.

- **Available Workflows**

Displays all the available workflows that are not moved to the Selected Workflows box.

- **Selected Workflows**

Displays all the workflows you have selected for the category.

- **Category Used for Workflow Authorization**

Enables you to select users with operator role who can execute the category. By default, the check box is not selected and the category is available for all users.

- **Available Operators**

Displays all the users with operator role.

- **Selected Operators**

Displays the users who are allowed to execute the category.

Command buttons

- **Selection buttons**

Enables you to move the selected entry from one box to another.

- **Save**

Saves the category and closes the dialog box.

- **Cancel**

Does not save the category and closes the dialog box.

Legal notices

Legal notices provide access to copyright statements, trademarks, patents, and more.

Copyright

<https://www.netapp.com/company/legal/copyright/>

Trademarks

NETAPP, the NETAPP logo, and the marks listed on the NetApp Trademarks page are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.

<https://www.netapp.com/company/legal/trademarks/>

Patents

A current list of NetApp owned patents can be found at:

<https://www.netapp.com/pdf.html?item=/media/11887-patentspage.pdf>

Privacy policy

<https://www.netapp.com/company/legal/privacy-policy/>

Open source

Notice files provide information about third-party copyright and licenses used in NetApp software.

[Notice for Workflow Automation 5.1.1](#)

[Notice for Workflow Automation 5.1](#)

Copyright information

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—with prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

Trademark information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.