



Technical Report

OnCommand Workflow Automation Workflows for the Storage as a Service

Rick Ehrhart and Ganesh Kamath, NetApp

February 2014 | TR-4272

Abstract

This tech report walks through the storage-as-a-service workflows that were created by NetApp experts using NetApp® OnCommand® Workflow Automation (WFA). As of today, these workflows have been tested, but are not certified. The roadmap plan is to have these workflows certified in a future release of WFA.

TABLE OF CONTENTS

1	OnCommand Workflow Automation (WFA) Workflows for Storage as a Service (StaaS)	4
1.1	Overview	4
1.2	Storage Service Concepts	4
2	Storage Service Components and Design	7
	Storage Domains	8
	Provisioning Policies	8
	Protection Policies	9
	Schedule	10
	Storage Services	10
	Consumer Information	12
	Storage Objects	12
3	Environment Setup and Installation	14
	Day Zero Requirements	14
	Importing Workflows and Creating Playground Schema	14
4	WFA Components and Workflows	19
	WFA Components	19
	Storage Service Catalog Package	19
5	Building Storage Services	25
	Instantiating Consumer and Provisioning Storage with Storage Service	25
	Creating Storage Service	26
	Creating Consumer	34
6	Viewing Storage Services Components	35
	View Consumer Objects	36
	View Provisioning Policy	38
	View Storage Domains	38
	View Storage Objects by Consumer	39
	View Storage Service	39
	Appendix	40
	Procedure to Call Workflows from External Orchestrator	40

LIST OF TABLES

Table 1)	Gold service level technology attributes	5
----------	--	---

Table 2) Silver service level technology attributes.....	6
Table 3) Bronze service level technology attributes.....	6
Table 4) Playground scheme tables.....	15
Table 5) Workflow list.....	20

LIST OF FIGURES

Figure 1) Technical attributes to service levels.....	5
Figure 2) Storage services.....	11
Figure 3) Storage services technical attributes.....	11
Figure 4) Scheme table relationships for storage services.....	13
Figure 5) Restored playground scheme tables.....	16
Figure 6) Importing the workflows.....	17
Figure 7) Storage service catalog category after import.....	18
Figure 8) Finders and filters filtered by playground scheme.....	23
Figure 9) Instantiating a consumer with a storage service.....	25
Figure 10) Data ONTAP cluster credentials in WFA.....	26
Figure 11) Storage service catalog.....	26
Figure 12) Executing the workflow.....	27
Figure 13) Creating a storage domain (aggregates).....	28
Figure 14) Creating a provisioning policy.....	29
Figure 15) Creating a schedule.....	30
Figure 16) Creating a local protection policy.....	31
Figure 17) Creating a remote protection policy (mirror).....	32
Figure 18) Creating a storage service.....	33
Figure 19) Creating a consumer (tenant).....	34
Figure 20) Provisioning storage for a consumer (tenant).....	35
Figure 21) View consumer objects workflow.....	37
Figure 22) View provisioning policy workflow.....	38
Figure 23) View storage domain workflow.....	38
Figure 24) View storage objects by consumer workflow.....	39
Figure 25) View storage service workflow.....	39

1 OnCommand Workflow Automation (WFA) Workflows for Storage as a Service (StaaS)

1.1 Overview

The workflows that are described in this tech report are the result of the OnCommand Workflow Automation workflows that were developed for a storage-as-a service project. The goal of this project is to have a few out-of-the-box constructs with which custom “service-level” WFA workflows can be built. The workflows use storage capabilities and features in clustered Data ONTAP® for deploying generic use-case scenarios. These workflows can be used in a proof of concept (PoC) to demonstrate the value of a storage service catalog. They can also be modified and extended by customers to reflect the specifics of their storage environment and service-level requirements and to create a complete and customized storage service catalog for their business.

Some example workflows to deploy multitier service levels based on different technology attributes could be:

- Performance-optimized storage with SSD and SAS drives (Flash Pool™, thick provisioning, local data protection, and remote data protection)
- A midlevel tier with SAS drives that can host general-purpose applications, VDI workloads, or datastores for a virtualization environment (performance drives with storage efficiency features and local data protection)
- Cost-optimized storage provisioned on lower tier storage controllers with SATA (high density with dedupe, compression, thin provisioning, and thin replication)

1.2 Storage Service Concepts

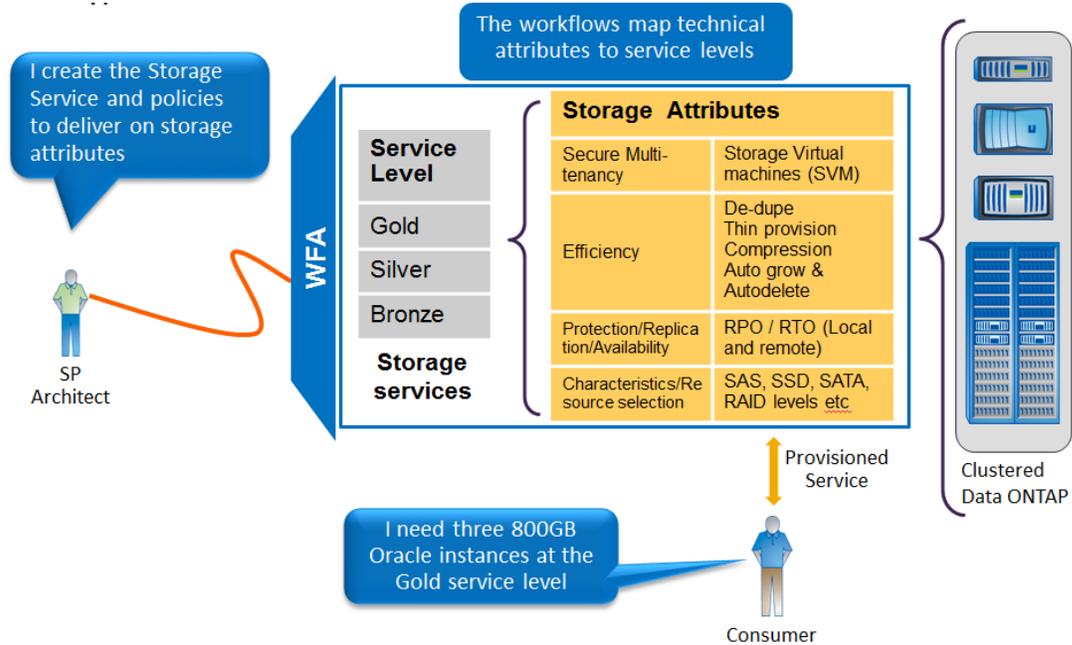
The goal is to differentiate storage services using different technology attributes for gold/silver/bronze deployments. For instance, a volume deployed using gold service will be space guaranteed with no deduplication and will also be protected with local Snapshot™ copies and SnapMirror® (or SnapVault®), whereas a volume deployed using bronze service will be thin provisioned and deduplicated, and protection will be using local Snapshot copies only.

Some of the key reasons to leverage WFA for our automation needs are its capabilities around workflows:

- A flexible set of commands to execute a workflow. The execution of these commands can be conditionally handled (example: create if not found, create based on a condition)
- Ability to have flexible resource selection in terms of filters and finders (resource selection criteria)
- Customizable workflows to fit our customers’ and partners’ unique requirements

Figure 1 gives a representation of mapping technical attributes to service levels by using the various storage characteristics that form the core of clustered Data ONTAP.

Figure 1) Technical attributes to service levels.



Example technology attributes for gold, silver, and bronze workflows have been detailed in the following tables. These values should be customized to suit the customer's environment and requirements.

Table 1) Gold service level technology attributes.

Technology Attributes	Attribute Definition	Attribute Values
<ul style="list-style-type: none"> FAS series 	<ul style="list-style-type: none"> Controller models or storage arrays (FAS series 22xx, 32xx, 62xx) 	<ul style="list-style-type: none"> FAS6290
<ul style="list-style-type: none"> Disk type Mount/map 	<ul style="list-style-type: none"> SSD, SAS, SATA, or a combination Access protocols 	<ul style="list-style-type: none"> SSD only or Flash Pool aggregates NFS3,SMB, iSCSI, FC
<ul style="list-style-type: none"> Media failure (RAID type) 	<ul style="list-style-type: none"> RAID configuration on the aggregates 	<ul style="list-style-type: none"> RAID-DP®
<ul style="list-style-type: none"> Local recovery 	<ul style="list-style-type: none"> Data protection using local Snapshot copies 	<ul style="list-style-type: none"> Snapshot schedules (23H + 6D + 1W)
<ul style="list-style-type: none"> Mirroring and DR (SnapMirror) 	<ul style="list-style-type: none"> Data protection using SnapMirror 	<ul style="list-style-type: none"> SnapMirror update schedules (hourly)
<ul style="list-style-type: none"> Space guarantee 	<ul style="list-style-type: none"> Space guarantees for writes and reserves for Snapshot copies 	<ul style="list-style-type: none"> Thick provisioned (volume guarantees)
<ul style="list-style-type: none"> Deduplication 	<ul style="list-style-type: none"> Data deduplication for different data types (binary yes/no) 	<ul style="list-style-type: none"> No
<ul style="list-style-type: none"> Compression 	<ul style="list-style-type: none"> Data compression for different data types (binary yes/no) 	<ul style="list-style-type: none"> No

<ul style="list-style-type: none"> Autogrow 	<ul style="list-style-type: none"> Automatically provide space in the FlexVol® volume when nearly full 	<ul style="list-style-type: none"> Yes (maximum size/increment size/grow threshold)
<ul style="list-style-type: none"> Autodelete 	<ul style="list-style-type: none"> Automatically provide space in the FlexVol volume when nearly full 	<ul style="list-style-type: none"> No (volume/snap trigger, and order of deletion)

Table 2) Silver service level technology attributes.

Technology Attributes	Attribute Definition	Attribute Values
<ul style="list-style-type: none"> FAS series Disk type Mount/map 	<ul style="list-style-type: none"> Controller models or storage arrays (FAS series 22xx, 32xx, 62xx) SSD, SAS, SATA or a combination Access protocols 	<ul style="list-style-type: none"> FAS3250 SAS aggregates. NFS3,SMB, iSCSI, FC
<ul style="list-style-type: none"> Media failure (RAID type) 	<ul style="list-style-type: none"> RAID configuration on the aggregates 	<ul style="list-style-type: none"> RAID-DP
<ul style="list-style-type: none"> Local recovery 	<ul style="list-style-type: none"> Data protection using local Snapshot copies 	<ul style="list-style-type: none"> Snapshot schedules (12H + 6D + 1W)
<ul style="list-style-type: none"> Mirroring and DR (SnapMirror) 	<ul style="list-style-type: none"> Data protection using SnapMirror 	<ul style="list-style-type: none"> SnapMirror update schedules (4hours)
<ul style="list-style-type: none"> Space guarantee 	<ul style="list-style-type: none"> Space guarantees for writes and reserves for Snapshot copies 	<ul style="list-style-type: none"> Thick provisioned (volume guarantees)
<ul style="list-style-type: none"> Deduplication 	<ul style="list-style-type: none"> Data deduplication for different data types (binary yes/no) 	<ul style="list-style-type: none"> Yes
<ul style="list-style-type: none"> Compression 	<ul style="list-style-type: none"> Data compression for different data types (binary yes/no) 	<ul style="list-style-type: none"> No
<ul style="list-style-type: none"> Autogrow 	<ul style="list-style-type: none"> Automatically provide space in the FlexVol volume when nearly full 	<ul style="list-style-type: none"> Yes (maximum size/increment size/grow threshold)
<ul style="list-style-type: none"> Autodelete 	<ul style="list-style-type: none"> Automatically provide space in the FlexVol volume when nearly full 	<ul style="list-style-type: none"> Yes (volume/snap trigger, and order of deletion)

Table 3) Bronze service level technology attributes.

Technology Attributes	Attribute Definition	Attribute Values
<ul style="list-style-type: none"> FAS series Disk type Mount/map 	<ul style="list-style-type: none"> Controller models or storage arrays (FAS series 22xx, 32xx, 62xx) SSD, SAS, SATA, or a combination Access protocols 	<ul style="list-style-type: none"> FAS22xx or 32xx or Data ONTAP Edge SAS/SATA or RAID 0 (edge) aggregates NFS3,SMB, iSCSI, FC (no Data ONTAP Edge support for FC)

<ul style="list-style-type: none"> Media failure (RAID type) 	<ul style="list-style-type: none"> RAID configuration on the aggregates 	<ul style="list-style-type: none"> RAID-DP
<ul style="list-style-type: none"> Local recovery 	<ul style="list-style-type: none"> Data protection using local Snapshot copies 	<ul style="list-style-type: none"> Snapshot schedules (6H + 2D + 1W)
<ul style="list-style-type: none"> Mirroring and DR (SnapMirror) 	<ul style="list-style-type: none"> Data protection using SnapMirror 	<ul style="list-style-type: none"> SnapMirror update schedules (once a day)
<ul style="list-style-type: none"> Space guarantee 	<ul style="list-style-type: none"> Space guarantees for writes and reserves for Snapshot copies 	<ul style="list-style-type: none"> Thin provisioned and no snap reserves
<ul style="list-style-type: none"> Deduplication 	<ul style="list-style-type: none"> Data deduplication for different data types (binary yes/no) 	<ul style="list-style-type: none"> Yes
<ul style="list-style-type: none"> Compression 	<ul style="list-style-type: none"> Data compression for different data types (binary yes/no) 	<ul style="list-style-type: none"> No
<ul style="list-style-type: none"> Autogrow 	<ul style="list-style-type: none"> Automatically provide space in the FlexVol volume when nearly full 	<ul style="list-style-type: none"> Yes (maximum size/increment size/grow threshold)
<ul style="list-style-type: none"> Autodelete 	<ul style="list-style-type: none"> Automatically provide space in the FlexVol volume when nearly full 	<ul style="list-style-type: none"> Yes (volume/snap trigger, and order of deletion)

2 Storage Service Components and Design

The storage services will be consumed by “consumers” or “tenants” that will subscribe to different storage service levels depending on their deployment needs. The storage administrator assigns the storage services to the consumer. The relationships between consumers, storage services, and other storage objects will be stored in a database that will be referenced during any consumer-related tasks. The tasks could be provisioning additional services, listing services, deleting services, or any other provisioning or protection tasks. The consumer mapping information will be updated in the database as necessary.

The database that is used to store the storage service metadata information is the playground scheme of the WFA database, which is included in the OnCommand Workflow Automation (WFA) server installation. The playground scheme is part of a MySQL database to which schema and tables can be built to include custom information and relationship matrixes, subsequently used by filters and SQL Server® queries. The tags or metadata can then be used along with the information in other WFA cache tables by WFA filters and user input queries.

All the metadata regarding the relationships between the different entities that make up a consumer will be stored in the playground scheme tables of the WFA database. The tables can be seen in the dictionary section under the Designer tab. These tables will be referred to during workflow execution and also populated postexecution. For example, a consumer creation will populate the consumer table, and associated entities within the playground scheme.

The playground scheme cannot be accessed using 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 directly access the database.

The information stored in the playground scheme includes:

- Storage domains
- Provisioning policies

- Protection policies (local and remote; only SnapMirror and SnapVault not supported in this release)
- Storage services
- Schedules
- Consumer information
- Storage objects

Storage Domains

A storage domain consists of a set of aggregates. There will be separate storage domains for each controller node in the protection topology. The primary and secondary controller nodes will have storage domains associated with each of them. Each provisioning policy is associated with a storage domain. When creating a storage domain, a set of aggregates will be presented in the form of cluster name and aggregate name.

Think of storage domains as resource pools that contain a set of aggregates grouped together by performance (storage type), geography (data centers and so on), or any other criteria. There can be a storage domain consisting of SSD and SAS disks, which can be associated with a provisioning node, and there can be another storage domain consisting of SATA disks, which can be associated with a protection node. This is up to the storage architect. For example, storage domains Dallas-SAS and Dallas-SATA could be created to divide the SAS and SATA storage in Dallas, or a storage domain Ft_Worth could be created to represent the entire Fort Worth data center.

Provisioning Policies

Provisioning policies are used for each node of the protection topology. For example, the primary would have thick provisioning, while the secondary would have thin.

Provisioning policies include these attributes:

- Provisioning policy name
- Controller model
- RAID type
- Disk type
- Space guarantee
- Deduplication
- Compression
- Autogrow
- Autodelete
- Storage domains

At provisioning policy creation, the storage domain is verified to match to the provision policy's characteristics. At least one aggregate in the storage domain must have the characteristics of the provisioning policy to allow the storage domain to be in the provisioning policy. A provisioning policy can include more than one storage domain. For example, a secondary provisioning policy could include two storage domains, Ft_Worth_SATA and Dallas_SATA. Basically, when the disk type is selected, the storage domains that qualify with the specified disk type will be filtered and shown. For example, if the disk type selected is SAS, only those storage domains with SAS disk types will be displayed during provisioning policy creation.

When a provisioning policy is created, a list of storage domains that fit the provisioning policy's service levels (SAS, SATA, SSD, and so on) are shown. The storage domain will be verified that at least one aggregate in the storage domain qualifies for the service level specified.

Protection Policies

There are two types of protection policies, local and remote. Local protection policies determine how primary storage is protected on the local node, while remote protection policies determine how primary storage is protected remotely.

Local Protection Policy

A local protection policy contains the following attributes and one or more local protection rules:

- Name
- Description

Local Protection Rule

A local protection rule contains the following attributes:

- Schedule
- Retention count
- Prefix
- Remote protection label

A local protection rule is a schedule that is assigned to a particular protection policy, and a single policy can have one or multiple schedules associated with it. For example, a local protection policy could have two different schedules: Snapshot copies daily at 8 p.m., and Snapshot copies every 4 hours, with different retention counts for each schedule. The schedules defined in the local protection policy will get instantiated on the storage controllers when storage is provisioned using the defined storage services that include this local protection policy.

The following is an example of a local protection policy with two associated schedules:

Vserver: testwfa

Policy Name	Schedules	Enabled	Comment
primary	2	true	-
Schedule	Count	Prefix	SnapMirror Label
Daily at 8 p.m.	2	Daily at 8 p.m.	-
Every 2 hours	1	Every 2 hours	-

Remote Protection Policy

A remote protection policy determines the protection attributes of remote protection, that is, replication using SnapMirror. Currently the workflows only support mirroring. Vaulting is not supported in this release.

A remote protection policy contains the follow attributes:

- Name
- Description
- Schedule
- Type (mirror, vault)
- Transfer_priority

- Restart
- Tries
- Ignore_access_time

Remote Protection Rule

A remote protection rule is used for vaulting only, and it contains the link to a local protection schedule using the snapmirror_label. Vaulting is currently not supported in this release, so the remote protection rule table is not used by any of the current workflows. The attributes are:

- Snapmirror_label
- Keep
- Preserve
- Warn
- Remote Recovery_Policy

Schedule

Schedules will be instantiated in Data ONTAP at provisioning time. For example, a local recovery schedule will be checked for, and if not present, it will be created.

The schedule is a cron schedule and follows the fields of a Data ONTAP cron schedule; therefore the attributes are:

- Name
- Description
- Days_of_month
- Days_of_week
- Months
- Hours
- Minutes

Schedules are used in local protection policies and remote protection policies.

Storage Services

Storage services consist of provisioning policies and protection policies for each node in the topology (if a secondary exists). This includes the storage domain relationships, the protection relationship type, and schedules for each relationship. Currently only two nodes are supported in a cascade topology. The current implementation does not support tertiary or forked (two different secondaries for the same primary) relationships.

Storage services will include:

- Storage service name
- Provisioning policy for the primary controller node
- Local protection policy (Snapshot copies)
- Provisioning policy for the secondary controller node
- Remote protection policy (only SnapMirror; SnapVault is not supported in this release)

Figure 2 shows a pictorial representation of a storage service with local and remote protection, which means that the service has associated a primary storage domain with provisioning and local protection policies and a secondary storage domain with provisioning and remote protection policies.

Figure 2) Storage services.

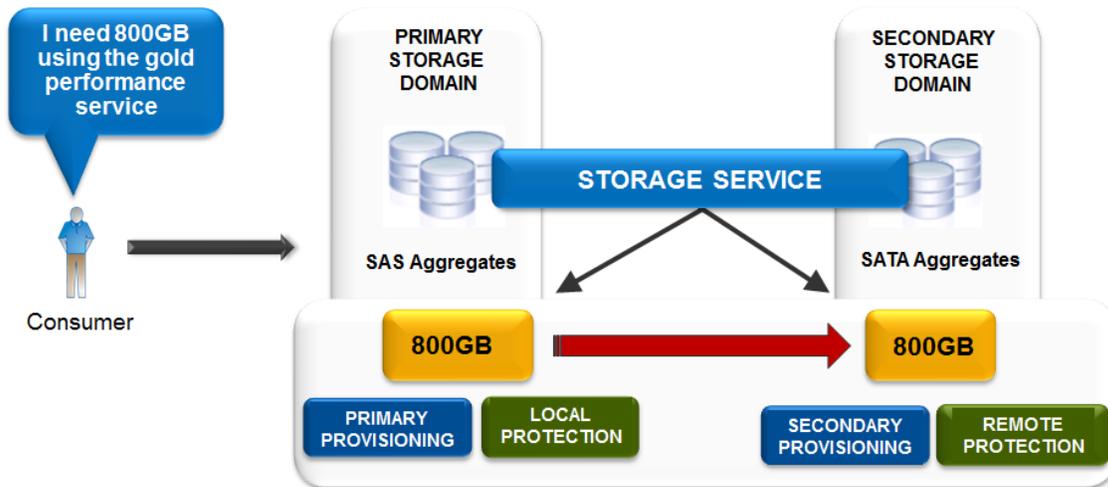
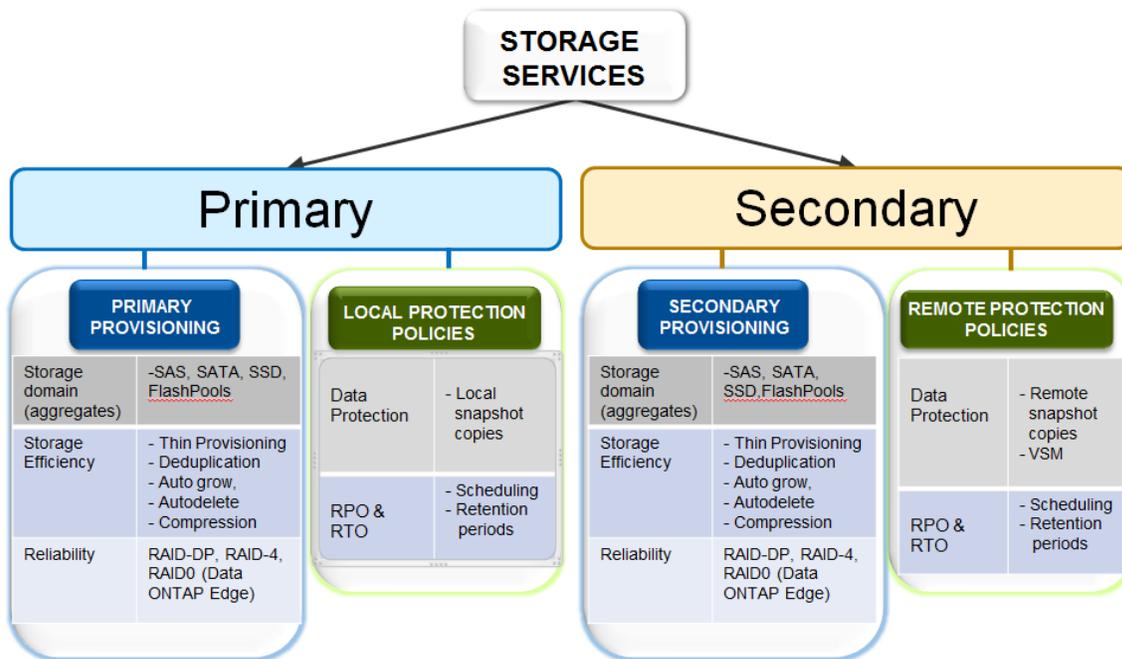


Figure 3 shows the technical attributes that are used for creating differentiated storage services.

Figure 3) Storage services technical attributes.



Consumer Information

Consumers, also known as tenants, provision, access, and decommission the provisioned storage. Each consumer can be associated with one more storage services. This association ties the consumer to a set of storage domains and eventually to a cluster and storage virtual machine (SVM).

The consumer information will include:

- Consumer name
- Storage services
- Primary storage cluster
- Primary SVM
- Secondary storage cluster
- Secondary SVM

Storage Objects

A storage object is an NFS export, a LUN, or a CIFS share on a volume that will be provisioned in Data ONTAP. The storage object will be the primary volume and the secondary volume if the storage service has a mirror.

Each storage object created needs to be associated with the consumer that created it and the storage service used to create it. This allows for a consumer to view the provisioned storage and provide showback or chargeback information.

Each storage object created is associated with the storage service with which it was created. This association allows the storage administrator to assign a cost for the storage object based on the storage service and to see all consumers using a particular service.

Each storage object created contains the primary volume and optional secondary volume where the storage object was created. This association allows the storage administrator to obtain capacity and performance of the storage object directly from Data ONTAP.

The storage objects contain the following:

- Object name
- Object type (export, LUN, share)
- Storage service
- Consumer
- Creation timestamp
- Primary volume (<cluster>://<primary SVM>/<primary volume>)
- Secondary volume (<cluster>://<secondary SVM>/<secondary volume>)

3 Environment Setup and Installation

Day Zero Requirements

Some physical and logical preconfiguration must be in place before the workflows can be used; that is, the day zero configuration must have been completed. The following assumptions are made:

- Clusters will be created and all cluster nodes added and properly configured for cluster membership.
 - All necessary physical cabling between the network and nodes meets best practices.
 - Cluster interconnect switch and ports are properly configured, and the nodes are properly connected.
 - Cluster management LIF is configured and connected to a VLAN that is accessible by WFA.
 - Manage ONTAP® SDKs are executable using the cluster management LIF.
- Clustered Data ONTAP version 8.2 has been installed.
- Cluster HA has been properly configured.
- Flash Cache™ is enabled.
- All required feature licenses have been installed on all nodes.
- 64-bit aggregates have been created on all the relevant clustered nodes (primary/secondary).
- The necessary storage virtual machines (SVMs), network port interface groups (ifgroups), logical interfaces (LIFs), routing groups, and VLANs have been created.
- Underlying network connectivity and configuration between potential cluster/SVM peers have been established for SnapMirror and SnapVault relationship configuration, including intercluster LIFs and routing groups. SVM and cluster peering relationships will be created by WFA if they do not already exist.
- NetApp Workflow Automation (WFA) version 2.1 is installed in the environment and configured.
- OnCommand Unified Manager 6.0 is installed and configured as a data source in WFA. The relevant clustered Data ONTAP clusters (primary, secondary, and so on) should also be discovered and managed in Unified Manager. Make sure that all the credentials for the Data ONTAP clusters are also configured in WFA.
- Because the provided storage service workflows are written in Perl, a Perl distribution package must be installed on the WFA server. Refer to the WFA [install and admin guide](#) for instructions.

Importing Workflows and Creating Playground Schema

Download the workflows and the playground schema from the NetApp Community site:

https://communities.netapp.com/community/products_and_solutions/storage_management_software/workflow-automation

The first step is to populate the WFA playground database with all the metadata/tables used by the workflows.

1. Create the tables used by the workflows in the WFA playground database.
2. Copy/install the custom perl modules:
 - a. ActiveRecord – 0.34
 - b. DBD:mysql – 4.0.022
3. Import the .dar workflows.
4. Run the workflows.

Creating Playground Scheme Tables

The tables in the playground scheme are created by restoring the empty playground scheme table structure using the mysql command. Without these tables, the storage service workflows to be imported will not execute because the workflows expect the tables to exist for reference/updating. Any existing tables in the playground scheme will be not be modified or deleted.

The following tables will get created in the playground scheme as part of this process.

Table 4) Playground scheme tables.

Tables	Description
consumer_storage_service	Consumers and subscribed storage services.
consumers	List of consumers.
cron_schedule	The different schedules that can be used in local and remote protection policies.
local_protection_policy	Local protection policy.
local_protection_rule	Different schedules that are tied to a local protection policy.
provisioning_policies	Provisioning policies and attributes.
provisioning_storage_domain	Provisioning policies and associated storage domains.
remote_protection_policy	Remote protection policy.
remote_protection_rule	Currently not being used because remote protection rules are for vaulting. Will be used in a future release when SnapVault support is added.
storage_domain	Storage domains.
storage_domain_member	Storage domain members: that is, list of aggregates that make up the storage domains.
storage_objects	List of storage objects that are provisioned: exports, LUNs, shares, and associated services and consumers along with the primary and secondary association for these objects.
storage_services	Storage services and associated provisioning and protection policies.

Restore the tables in the playground scheme using the following command:

```
mysql -u wfa -p playground < c:\playground.sql
```

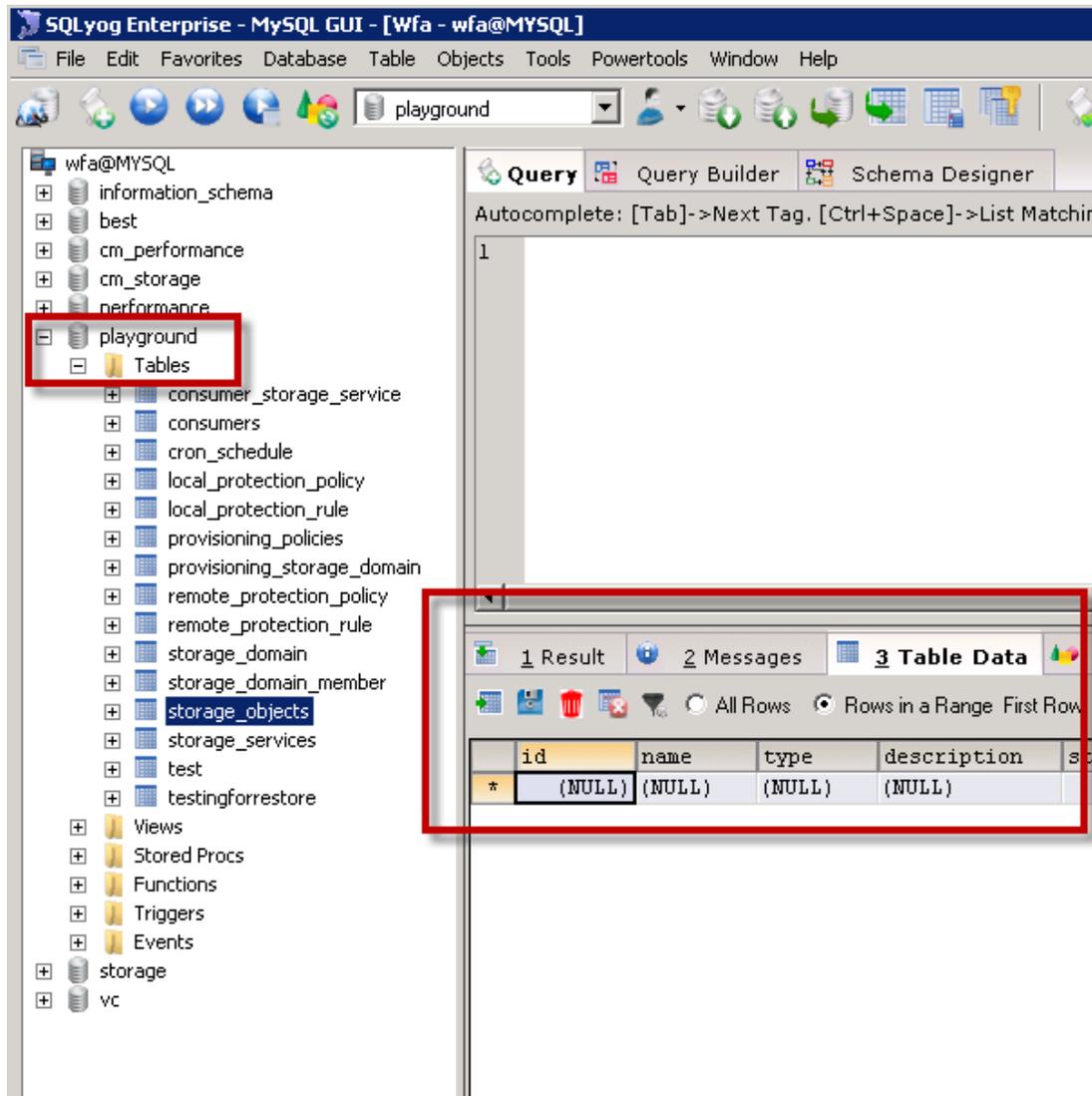
The default password is "Wfa123."

Note: "C:\playground.sql" is the playground schema file that you downloaded from the NetApp Communities site.

The playground db cannot be accessed using 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 WFA database and verify that the tables have been created in the playground scheme.

A screenshot of the playground scheme and verifying that the table structure (no data) has been created, as shown in Figure 5.

Figure 5) Restored playground scheme tables.



Additional Perl Module Requirements for Workflows

There are additional perl modules used by the workflows that need to be installed for the workflows to execute successfully. The specific versions that were tested and used with the workflows are:

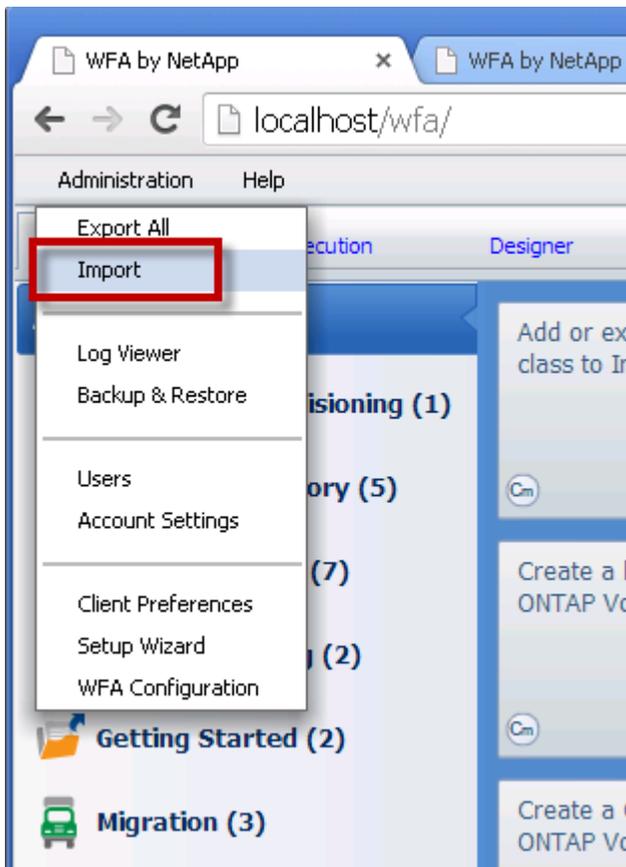
- ActiveRecord::Simple – Version 0.34
- DBD::mysql – Version 4.022

Importing Workflows

Import the .dar file that contains all the workflows. The .dar file is downloaded from the NetApp Community site.

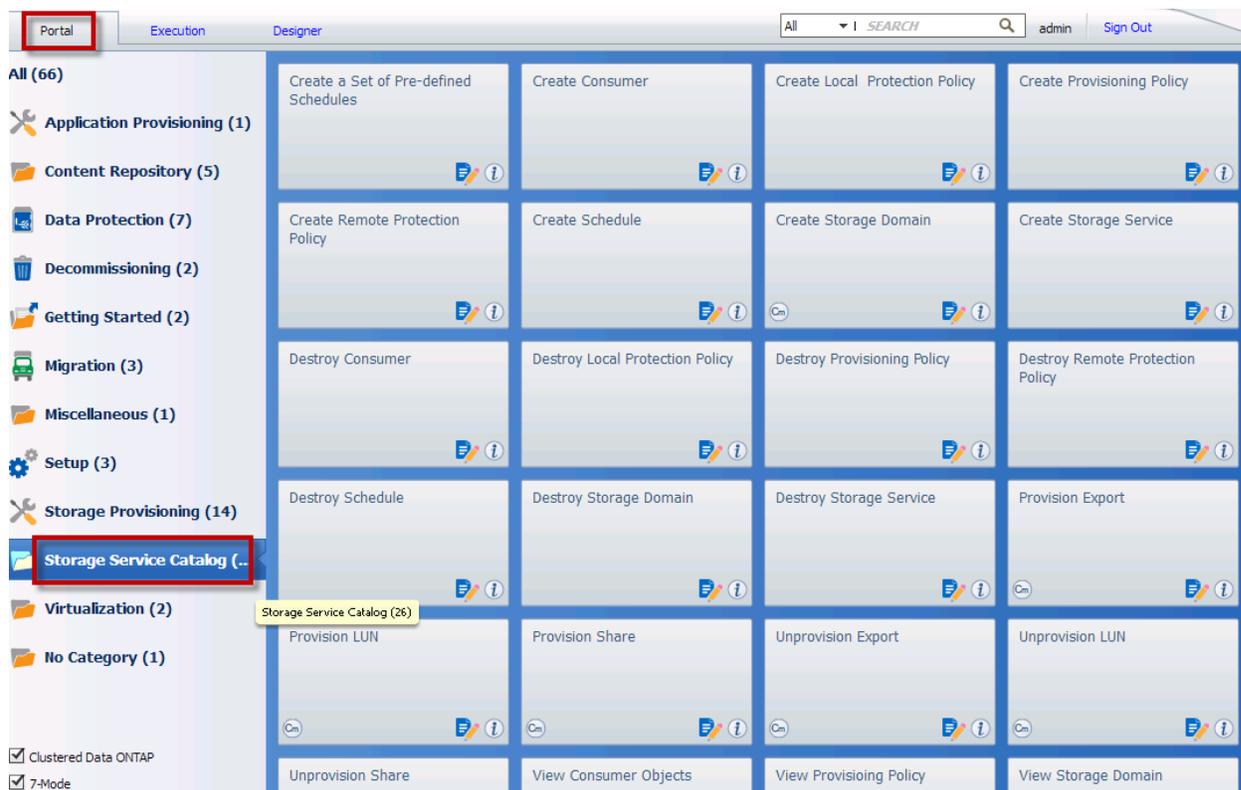
1. Log in to the WFA portal and click Import under the Administration menu.

Figure 6) Importing the workflows.



2. Select the .dar file and open it. You will see a notification box detailing all the new components that are being imported in green text.
3. Click Import.
4. The import should finish successfully.
5. All the imported workflows will show up as a new category (Storage Service Catalog) under the Portal tab.

Figure 7) Storage service catalog category after import.



The workflows are now ready for use. All the relevant filters, finders, dictionary items, and commands are also imported from the .dar file.

4 WFA Components and Workflows

WFA Components

WFA has a number of types of building blocks to achieve the preceding automation goals. A short description of the same is provided here (more details in WFA product documentation):

- **Data sources.** A data source is a read-only data structure that serves as a connection to the data source object of specific data source type. One such required data source is a connection to an OnCommand Unified Manager 6.0 database. WFA collects resource information from the data sources and formats it for the caching scheme.
- **Cache.** WFA has an internal cache database, which it periodically refreshes from OnCommand Unified Manager. This contains information of the entire storage environment (clusters, SVMs, aggregates, volumes, LUNs, initiator groups, and so on).
- **Filters.** WFA provides a means to describe a specific resource selection criteria based on values of attributes of all supported object types (example: filter aggregates of RAID type RAID-DP).
- **Finders.** 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.
- **Commands.** A step in a workflow is called a command and generally carries out a meaningful granular step (example: create a volume, map a LUN).
- **Templates.** A template of values of various attributes of any supported object type can be created in WFA and used in workflow design (example: space guaranteed settings for a volume; this template can be used during the creation of a volume). A template is used as a blueprint for an object definition.
- **Workflow.** A repeatable process for achieving storage automation, which contains a set of commands, filters, templates, and other conditional execution logic such as loops and approval points.
- **Dictionary entries.** Dictionary entries 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 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.

Storage Service Catalog Package

- The Storage Service Catalog Package contains table definitions, workflows, commands, filters, finders, and functions. All these objects can be viewed under the “Designer” tab. To make life easier, filter on the playground scheme.
- The workflows in the Storage Service Catalog Package consist of provisioning and unprovisioning exports, LUNs, and shares as well as workflows to create and remove the storage service catalog objects discussed previously.
- The storage service catalog objects have commands that operate on the objects. All commands support create, update, and delete.
- Filters and finders allow the commands to find and filter objects in the playground scheme. These allow the commands to find the correct objects. These objects also help in the “view” workflows.
- Functions are provided to assist calculations or string parsing.

Workflows

The imported .dar file has the workflows described in Table 5.

Table 5) Workflow list.

	Table Workflow	Use	Notes
1	Create a set of predefined schedules	The set of predefined schedules can be used as examples to create new schedules or used just as they are.	None.
2	Create schedule	Creates a new schedule.	None.
3	Create storage domain	Groups aggregates for provisioning.	None.
4	Create provisioning policy	Groups storage object attributes and storage domains and creates a provisioning policy.	None.
5	Create local protection policy	Creates local protection attributes.	None.
6	Create remote protection policy	Creates remote protection attributes.	None.
7	Create storage service	Associates provisioning and protections policies to create a storage service.	None.
8	Create consumer	Creates a consumer and assigns storage services, clusters, and SVMs (primary and/or secondary).	A consumer can be associated with a single primary SVM and a single secondary SVM. There is a many:1 mapping between consumers and SVMs. A consumer can be mapped to only one SVM. but the SVM can be shared with other consumers.
9	View consumer objects	View the consumer's associated objects.	None.
10	View provisioning policy	View the provisioning policy's associated objects.	None.
11	View storage domains	View the storage domains and associated members.	None.
12	View storage services	View the storage service's associated objects.	None.
13	View storage objects by consumer	View the storage objects that are associated to a consumer.	None.
14	Destroy schedule	Removes a schedule.	Storage service catalog schedule is removed, but the Data ONTAP schedule is not deleted.
15	Destroy storage domain	Removes the association of aggregates to a storage domain.	None.

	Table Workflow	Use	Notes
16	Destroy provisioning policy	Remove a provisioning policy.	None.
17	Destroy local protection policy	Removes a local protection policy and its associated schedules.	The local protection policy is deleted, but not the Data ONTAP Snapshot policy.
18	Destroy remote protection policy	Removes a remote protection policy and its associated schedules.	The remote protection policy is removed, but not the Data ONTAP SnapMirror policy.
19	Destroy storage service	Removes the storage service and the associated provisioning and protection policies.	None.
20	Destroy consumer	Removes the consumer.	None.
21	Provision export	Provisions an export.	In Data ONTAP, a volume is created with the provisioning policy attributes. An export policy, <consumer>_export_policy is created if it doesn't exist. Export rules are added from user input. Schedules and Snapshot and SnapMirror policies are created if they don't exist. When additional exports are created, the new rules are added to the <consumer>_export_policy. There is no method in the workflow to create a custom export policy.
22	Provision LUN	Provisions one or more LUNs.	In Data ONTAP, a volume is created with the provisioning policy attributes. LUNs are created and mapped using the LUN prefix in a new volume. Schedules and Snapshot and SnapMirror policies are created if they don't exist. An "auto_igroup" is created by default if no igroup is specified during provisioning, and the auto_igroup will be used during future provisioning as well if an igroup is not specified.
23	Provision share	Provisions a share.	In Data ONTAP, a volume is created with the provisioning policy attributes. A share is created for the new volume. Schedules and Snapshot and SnapMirror policies are created if they don't exist.

	Table Workflow	Use	Notes
24	Unprovision export	Removes an export.	In Data ONTAP, the primary and secondary volumes are removed.
25	Unprovision LUN	Removes one or more LUNs starting with the LUN prefix.	In Data ONTAP, the LUNs are unmapped and removed, and the associated primary and secondary volumes are removed.
26	Unprovision share	Removes a share.	In Data ONTAP, the share and associated primary and secondary volumes are removed.

Commands

The commands in the Storage Service Catalog Package operate on the WFA dictionary objects. All the commands have three actions: “create,” “update,” and “delete.”

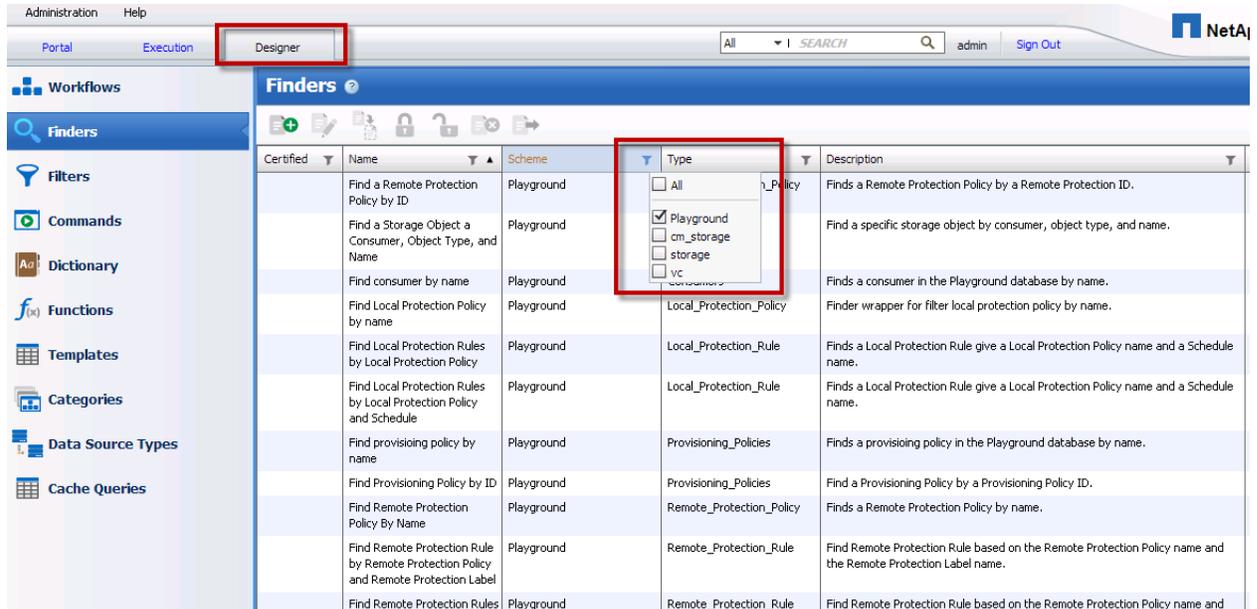
Table 6) Commands list.

	Commands	Use	Side Effects
1	Consumer operations	Creates, updates, and deletes consumers.	Operates on the Consumer_Storage_Service table.
2	Cron schedule operations	Creates, updates, and deletes schedules.	None.
3	Local protection policy operations	Creates, updates, and deletes local protection policies.	None.
4	Local protection rule operations	Creates, updates, and deletes local protection rules.	Requires a local protection policy database ID for creation.
5	No-op storage service	Allows for finders to operate on storage service catalog objects for use in workflows.	None.
6	Provisioning policy operations	Creates, updates, and deletes provisioning policies.	Operates on the provisioning storage domain table.
7	Remote protection policy operations	Creates, updates, and deletes local protection policies.	None.
8	Remote protection rule operations	Creates, updates, and deletes local protection rules.	Requires a remote protection policy database ID for creation.
9	Storage domain operations	Creates, updates, and deletes storage domains.	Operates on the Storage_Domain_Members table.
10	Storage object operations	Creates and deletes storage objects.	None.
11	Storage service operations	Creates, updates, and deletes storage services.	None.

Filters and Finders

WFA finders are a collection of one or more filters that specify return attributes. You can sort all the finders and filters used by the imported workflows by filtering on the playground scheme.

Figure 8) Finders and filters filtered by playground scheme.



The following filters are in the Storage Service Catalog Package:

- Filter consumers by name
- Filter cron schedules by local protection policy
- Filter cron schedules by remote protection policy
- Filter local protection policy by name
- Filter local protection rules by local protection policy
- Filter local protection rules by remote protection label
- Filter local protection rules by schedule
- Filter provisioning policies by name
- Filter provisioning policy by ID
- Filter remote protection policy by name
- Filter remote protection policy by ID
- Filter remote protection policy by remote protection label
- Filter remote protection rules by remote protection policy
- Filter storage domains by name
- Filter storage objects by consumer
- Filter storage objects by name
- Filter storage objects by type
- Filter storage objects similar to name
- Find cron schedule by name

- Find storage domains by disk type
- Find storage domains by RAID type
- Filter storage domains by technical SLCs
 - This includes disk type, RAID type, and node controller model.

The following finders are part of the Storage Service Catalog Package:

- Find a remote protection policy by ID
- Find a storage object by consumer, object type, and name
- Find consumer by name
- Find local protection policy by name
- Find local protection rules by local protection policy
- Find local protection rules by local protection policy and schedule
- Find provisioning policy by name
- Find provisioning policy by ID
- Find remote protection policy by name
- Find remote protection rule by remote protection policy and remote protection label
- Find remote protection rules by remote protection policy
- Find schedule by name
- Find storage domain by name
- Find storage domains by provisioning policy SLCs
- Find storage objects by consumer and object type
- Find storage service by name
- Find storage services from attached consumer name
- Return cron schedules in local protection policy
- Return cron schedules in remote protection policy

Functions

Functions were added to aid in creating parameters for WFA commands. The follow functions were added:

- **booleanToString(i)**. Converts a zero (0) to “false” and a one (1) to “true.”
- **forge_export(string)**. Verifies the export name starts with a slash (/).
- **Forge_vol_path(string)**. Creates a full path consisting of cluster://vserver/volume for logging.
- **get_space_guarantee(string)**. Returns the Data ONTAP volume space guarantee from the provisioning policy space guarantee specification.
- **getAutoDeleteOptions(string)**. Returns the Data ONTAP autodelete options from the autoshrink option in a provisioning policy.
- **getMirrorType(string)**. Returns the Data ONTAP SnapMirror type from the remote protection policy.
- **getWfaUser(i)**. Returns the WFA database user name. If the WFA database user name is changed, then this function must be modified to return the new WFA database user name. Any integer can be passed as an input parameter, because it is not used.
- **getWfaPassword(i)**. Returns the WFA database password. If the WFA database password is changed, then this function must be modified to return the new WFA database password. Any integer can be passed as an input parameter, because it is not used.

- **notNull(data)**. Returns zero (0) if the input parameter is null and a one (1) if the input parameter is not null. Used to skip a WFA workflow row in the loop construct.

Note: When creating a consumer, one or more storage services must be assigned to the consumer. The storage architect will be able to choose from a set of SVMs that are available to operate on the respective provisioning policy's storage domain. Basically SVMs will be chosen from a set of SVMs that can operate on the storage domain, or if none is found, then the aggregates in the storage domain should be added to the SVM list of allowed aggregates, or a new SVM should be created that can operate on the storage domains that have been created.

Note: A consumer can be mapped to a single SVM, but SVMs can be shared across multiple consumers.

5 Building Storage Services

After the workflows have been imported, the methodology to build the storage services will be:

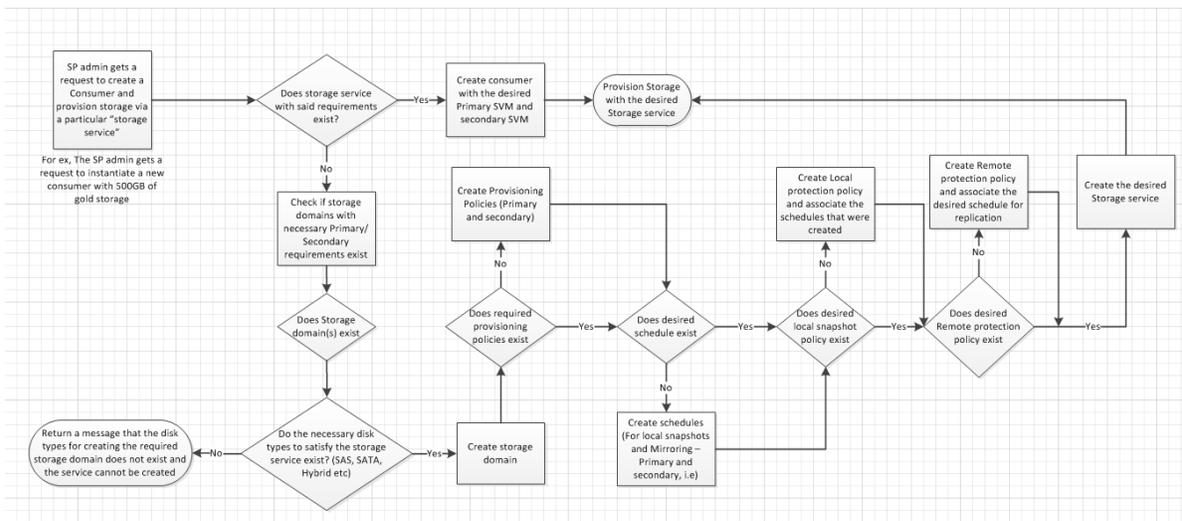
- Create storage domains
- Create one or more provisioning policies and associate them to the appropriate storage domains
- Create one or more schedules
- Create one or more local protection policies and associate with the appropriate schedules
- Create one or more remote protection policies (mirror) and associate with the appropriate schedules
- Create storage services with the appropriate primary/secondary provisioning and protection policies
- Create consumers and associate with storage services
- Create/provision exports/LUNs/shares for the consumers using the storage services

There are also additional workflows to view the storage objects and to deprovision the storage objects.

Instantiating Consumer and Provisioning Storage with Storage Service

Figure 9 shows the flowchart for instantiating a consumer along with a storage service. The exact steps along with the screenshots are also provided in the subsequent section. After the consumer is instantiated, storage can then be provisioned for the consumer with a desired storage service.

Figure 9) Instantiating a consumer with a storage service.

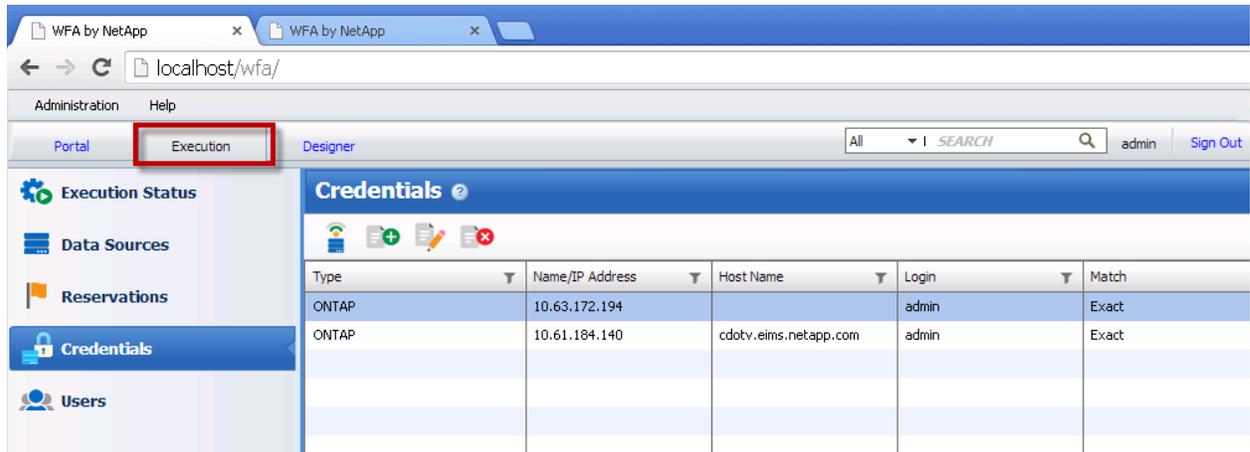


Creating Storage Service

Prerequisites

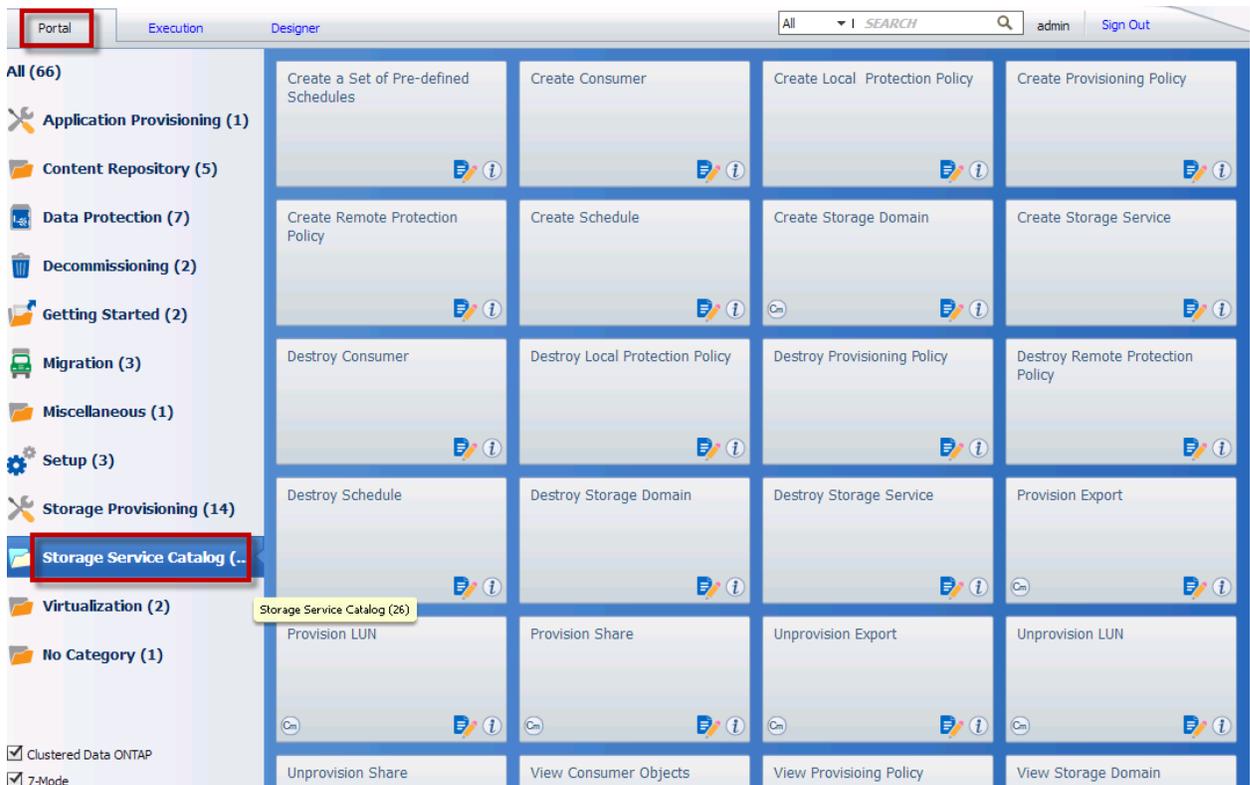
1. Confirm that the Data ONTAP cluster credentials are configured in WFA as shown in Figure 10.

Figure 10) Data ONTAP cluster credentials in WFA.



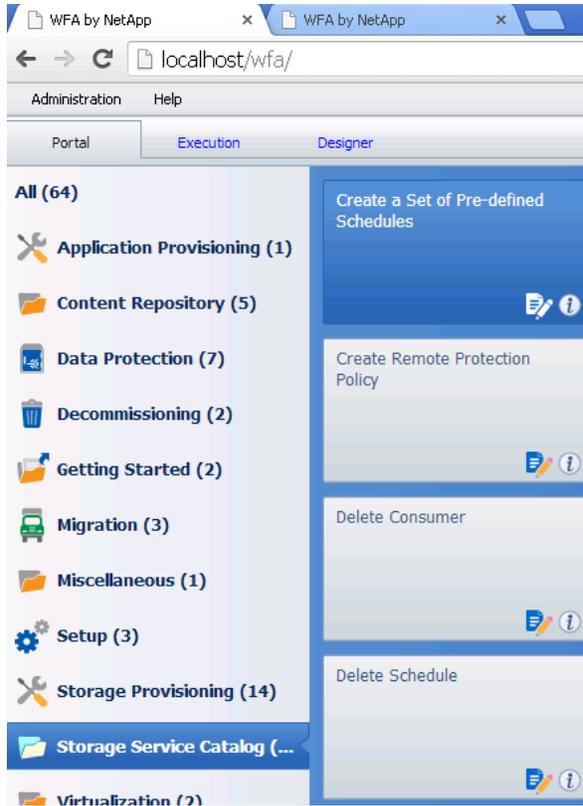
2. The workflows can be filtered by clicking the storage service catalog category under the Portal tab to list all the custom workflows that have been imported.

Figure 11) Storage service catalog.



- Execute the “Create a set of Predefined schedules” workflow by clicking the workflow. This workflow creates a set of predefined schedules that should suit most protection needs. If a custom schedule is desired, it can be created using the “Create local protection policy” workflow.

Figure 12) Executing the workflow.



Building Storage Service

- The first step in building a storage service is to start with creation of storage domains. Execute the “Create Storage Domain” workflow.

Figure 13) Creating a storage domain (aggregates).

Execute Workflow 'Create Storage Domain'

User Inputs

Storage Domain Name*: primary_sas

Storage Domain Description :

Cluster : rtp-cse-cl01

Aggregates :

- rtp_cse_cl01_n01_sas450_aggr02
- rtp_cse_cl01_n02_sas450_aggr01
- rtp_cse_cl01_n02_sas450_aggr02
- rtp_cse_cl01_n03_sas450_aggr01

Options

Execute now:

Choose date and time for execution:

Execution Comment

Preview Execute Cancel

2. Create a provisioning policy and associate it with a storage domain. Storage efficiency parameters (thin provisioning, deduplication, compression, and so on) are specified here.

Figure 14) Creating a provisioning policy.

Execute Workflow 'Create Provisioning Policy' ?

User Inputs

Policy Name*: gold_primary

Policy Description :

RAID Type : raid_dp

Space Guarantee : thick

Controller model : FAS

Disk Type : sas

Enable Deduplication :

Enable Compression :

Enable Auto Grow :

Enable Auto Shrink :

Storage Domains :

Search

- aggr1node1
- aggr1node12
- aggr1node2

Options

Execute now:

Preview Execute Cancel

Note: The workflow also supports Data ONTAP Edge. If you want to create a storage domain with Data ONTAP Edge, select the RAID type to be Raid_0 in the drop-down menu for RAID type. One of the use cases would be to use Data ONTAP Edge as a mirror destination.

3. Create a schedule if the preexisting schedules do not match the requirements.

Figure 15) Creating a schedule.

Execute Workflow 'Create Schedule'

User Inputs

Schedule

Schedule Name*:

Schedule Description :

Months :

Days of Month :

Days of the Week :

Hours :

Minutes :

Options

Execute now:

Choose date and time for execution:

Execution Comment

Preview Execute Cancel

4. Create a local protection policy and associate the desired schedule.

Figure 16) Creating a local protection policy.

Execute Workflow 'Create Local Protection Policy'

User Inputs

Local Protection Policy*: local

Local Protection Policy Description :

[-] Schedule

Schedule Name*: Daily at 8 pm

[-] Schedule Attributes

Retention Count*: 4

Prefix :

Remote Protection Label :

Options

Execute now:

Choose date and time for execution:

Execution Comment

Preview Execute Cancel

5. Create a remote protection policy that defines the replication characteristics.

Figure 17) Creating a remote protection policy (mirror).

Execute Workflow 'Create Remote Protection Policy'

User Inputs

Remote Protection Policy*: mirror every 2 hours

Remote Protection Policy Description :

Relationship Type*: mirror

Transfer Priority : normal

Schedule : Every 2 hours

Tries : 8

Ignore Access Time :

Restart : default

+ Vault Rule

Options

Execute now:

Choose date and time for execution:

Execution Comment

Preview Execute Cancel

6. Create a storage service and associate the previously created components to build the desired service. Associate the primary provisioning policy and the local protection policy to the storage service. If the storage service needs to have a secondary, then associate secondary provisioning policy and the remote protection policy to the service as well. This will also determine what the primary and the secondary SVMs/clusters are when creating a “consumer” in the subsequent steps.

Figure 18) Creating a storage service.

Execute Workflow 'Create Storage Service'

User Inputs

Storage Service Name*: gold_primary

Storage Service Description : gold service with mirroring

Primary Information

Primary Provisioning Policy : aggr1node1

Local Protection Policy : local

Secondary Information

Secondary Provisioning Policy : cdotv

First Remote Protection Policy : remote

Options

Execute now:

Choose date and time for execution:

Execution Comment

Preview Execute Cancel

Creating Consumer

1. Create a consumer with the desired service levels. Select the storage service to associate with the consumer and the primary and the secondary clusters/SVMs. The secondary cluster/SVM will be active only if the storage service that is selected has an associated secondary provisioning policy/storage domain.

Figure 19) Creating a consumer (tenant).

The screenshot shows a dialog box titled "Execute Workflow 'Create Consumer'". It is divided into two main sections: "User Inputs" and "Options".

User Inputs:

- Consumer Name*: Ford
- Consumer Description : (empty)
- Storage Service 1*: Primary with Mirror
- Storage Service 2 : NONE
- Storage Service 3 : NONE
- Primary Cluster*: ccr-cmode-02
- Primary SVM*: testwfa
- Secondary Cluster : cdotv
- Secondary SVM : PublicCloud

Options:

- Execute now:
- Choose date and time for execution:

This is a critical step, because it defines the primary cluster and SVM and optionally the secondary cluster and SVM.

Note: An important point to note here is that a consumer can ONLY be associated with one SVM: the SVM that is selected when creating the consumer. The consumer cannot be associated with multiple SVMs, but one SVM can share multiple consumers. You have an option to select up to three storage services when creating a consumer, and all the storage services will provision on the same primary SVM and the same secondary SVM. You have to make sure that the selected SVM is allowed to operate on the aggregates in the storage domain. The aggregates should be in the "list of aggregates" on which the SVM is allowed to operate; use the Data ONTAP CLI command "vserver show -vserver <vserver name> -instance" to verify.

2. Provision storage for the consumer using a storage service. A consumer can have multiple storage services associated: gold, bronze, with protection, without protection, and so on. The provisioned storage will match the criteria specified in the storage service selected in this step.

Figure 20) Provisioning storage for a consumer (tenant).

Execute Workflow 'Provision Export'

User Inputs

Consumer Name*: Ford

Storage Service Name*: Primary with Mirror

Export Name*: test

Export Size (GB)*: 1

Export Description :

Export Details

client specification	read-only rule	read-write rule

Options

Execute now:

Choose date and time for execution:

Note: If a storage service is chosen that includes secondary protection, the mirror relationship will also be initialized. The SVM and cluster peering will be done by WFA, but it is recommended to manually check beforehand that network connectivity is properly configured (intercluster LIFs, routing groups, reachability between clusters, and so on).

6 Viewing Storage Services Components

There are workflows that can be utilized for viewing the components that are created. These workflows help in verifying the storage services to which a consumer subscribes and the storage objects that are created for each consumer. You can also view the provisioning policies, storage domains, and storage services.

Because WFA does not provide a way to list its objects, viewing workflows are provided. The viewing workflows allow the user to select a field and view the relevant information. For example, when viewing the consumer objects, the consumer is selected, and the list of objects will be displayed for the specific consumer. If the workflow is executed, nothing will be modified.

Table 7) View workflows.

	View Workflows	Description
1	View consumer objects	Views the different storage services to which a consumer subscribes and the components that make up the storage service.
2	View provisioning policies	Views the provisioning policies and storage domains associated with the provisioning policy.
3	View storage domains	Views the storage domains and members.
4	View storage objects by consumer	Views the storage objects created by consumers (primary volume, secondary volume, type of object: LUN/export/share).
5	View storage services	Views the storage services and local and remote provisioning and protection policies and the consumers subscribing to that storage service.

View Consumer Objects

Figure 21 shows a screenshot of a view consumer objects workflow.

View the storage service to which the consumer is subscribing, along with the primary cluster/SVM and the local and remote provisioning and protection policies.

Figure 21) View consumer objects workflow.

Execute Workflow 'View Consumer Objects' ?

User Inputs

Consumers : Ford

Storage Services : Primary with Mirror

Primary Policies

Primary Provisioning Policy : Standard Primary

Primary Cluster: Vserver : ccr-cmode-02:testwfa

Local Protection Policy : 30 minutes and daily

Local Protection Rules :

	name	retention_count
<input type="checkbox"/>	Daily at 8 pm	7
<input type="checkbox"/>	Every 30 minutes	48

Secondary Policies

Secondary Provisioning Policy : Low cost mirror

Secondary Cluster:Vserver : cdotv:PublicCloud

Remote Protection Policy : Hourly Mirror

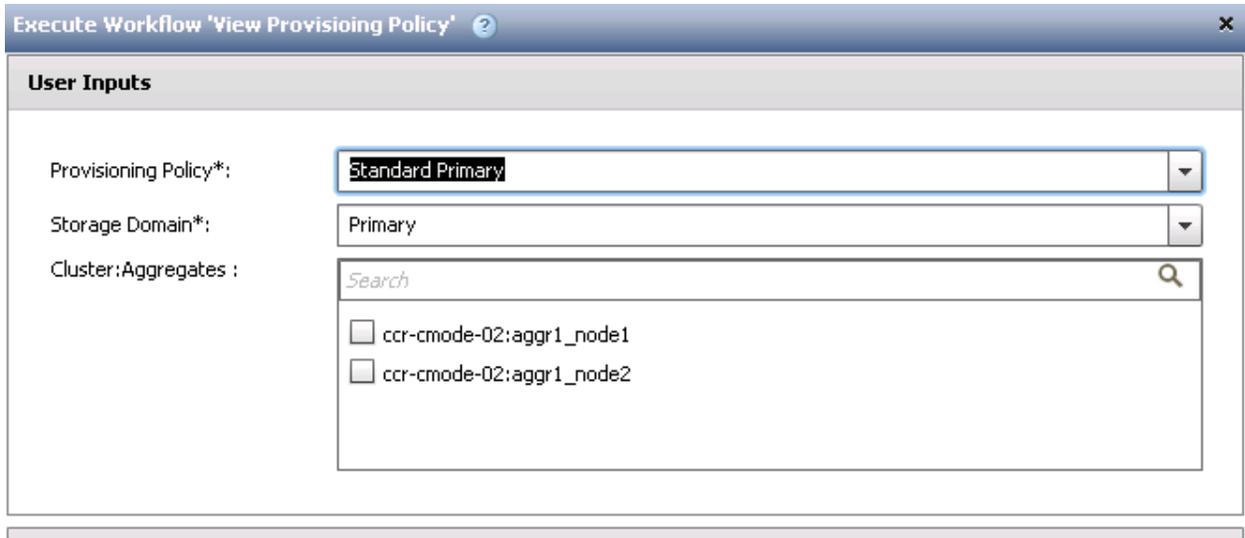
Schedule Info

Schedules : Daily at 8 pm

View Provisioning Policy

Figure 22 shows a screenshot of viewing provisioning policies and the storage domains with which each provisioning policy is associated.

Figure 22) View provisioning policy workflow.



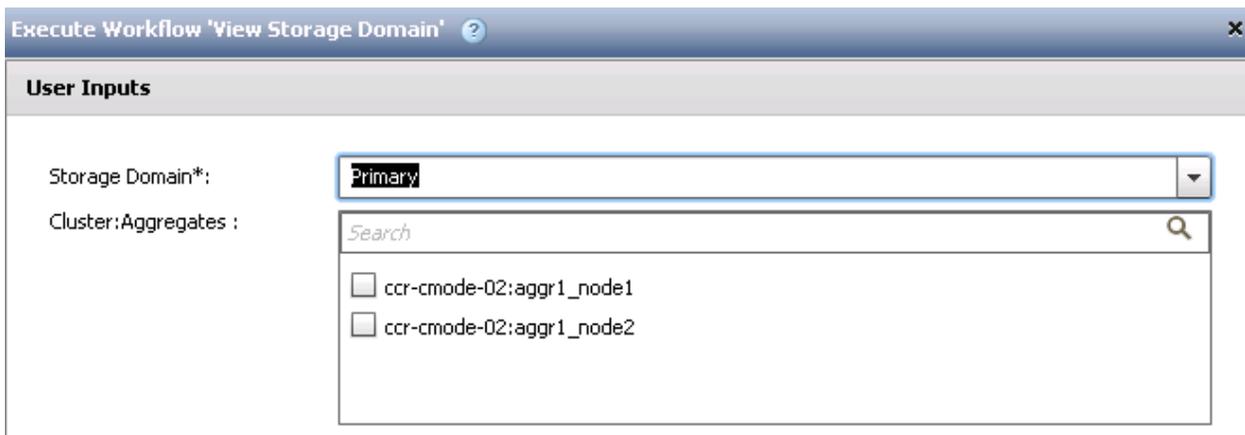
The screenshot shows a window titled "Execute Workflow 'View Provisioning Policy'". Under the "User Inputs" section, there are three fields:

- Provisioning Policy*:** A dropdown menu with "Standard Primary" selected.
- Storage Domain*:** A dropdown menu with "Primary" selected.
- Cluster:Aggregates :** A search box containing "Search" and a magnifying glass icon. Below it, two items are listed with checkboxes:
 - ccr-cmode-02:aggr1_node1
 - ccr-cmode-02:aggr1_node2

View Storage Domains

Figure 23 shows a screenshot of storage domains and its associated members using the view storage domain workflow.

Figure 23) View storage domain workflow.



The screenshot shows a window titled "Execute Workflow 'View Storage Domain'". Under the "User Inputs" section, there are two fields:

- Storage Domain*:** A dropdown menu with "Primary" selected.
- Cluster:Aggregates :** A search box containing "Search" and a magnifying glass icon. Below it, two items are listed with checkboxes:
 - ccr-cmode-02:aggr1_node1
 - ccr-cmode-02:aggr1_node2

View Storage Objects by Consumer

Figure 24 shows a screenshot of viewing the storage objects for a particular consumer. It shows the primary and secondary volumes for the consumer and the type of object (LUN/export/share) along with the storage service used to create the objects.

Figure 24) View storage objects by consumer workflow.

	name	type	description	timestamp	storage service	primary_volume	secondary_volume
<input type="checkbox"/>	/bill_demo	export	bill demo export	2014-01-23 ...	Primary with Mirror	ccr-cmode-02://te...	cdrv://PublicCloud/bill_demo_secondary
<input type="checkbox"/>	/tesla_mkt	export	Marketing	2014-01-23 ...	Primary Only	ccr-cmode-02://te...	

View Storage Service

Figure 25 is a screenshot of a view storage services workflow displaying the components that make up the storage service and the consumers subscribing to that storage service.

Figure 25) View storage service workflow.

Storage Service*: Primary with Mirror

Primary Node

Primary Provisioning Policy : Standard Primary

Local Protection Policy : 30 minutes and daily

Primary Storage Domain : Primary

Secondary Node

Secondary Provisioning Policy : Low cost mirror

Remote Protection Policy : Hourly Mirror

Secondary Storage Domain : Secondary low cost

Consumers

Consumers :

Search

Dodge

Ford

Tesla

Appendix

Procedure to Call Workflows from External Orchestrator

For detailed information about REST APIs, see the WFA web services primer for REST API on the Workflow Automation space within the OnCommand community:

<https://communities.netapp.com/docs/DOC-15285>

You can use the REST APIs provided by Workflow Automation (WFA) to invoke workflows from external portals and data center orchestration software.

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

Refer to the [Interoperability Matrix Tool \(IMT\)](#) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer's installation in accordance with published specifications.

NetApp provides no representations or warranties regarding the accuracy, reliability, or serviceability of any information or recommendations provided in this publication, or with respect to any results that may be obtained by the use of the information or observance of any recommendations provided herein. The information in this document is distributed AS IS, and the use of this information or the implementation of any recommendations or techniques herein is a customer's responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. This document and the information contained herein may be used solely in connection with the NetApp products discussed in this document.

Go further, faster®



www.netapp.com

© 2014 NetApp, Inc. All rights reserved. No portions of this document may be reproduced without prior written consent of NetApp, Inc. Specifications are subject to change without notice. NetApp, the NetApp logo, Go further, faster, Data ONTAP, Flash Cache, Flash Pool, FlexVol, Manage ONTAP, OnCommand, RAID-DP, SnapMirror, Snapshot, and SnapVault are trademarks or registered trademarks of NetApp, Inc. in the United States and/or other countries. SQL Server is a registered trademark of Microsoft Corporation. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such. TR-4272-0214