



Federated Management for Fleets of Pure Storage® Systems

Managing digital resources and assets effectively while meeting the needs of technology users is a growing challenge for IT infrastructure teams. This is especially true of data storage, where it's not only about equipment and software, but also about data that may be of far greater value than the equipment and must be accessible by users, protected against cyberattack and human error, and compliant with organization and regulatory standards.

Exponential data growth, especially fueled by the recent rise of artificial intelligence, has led organizations to deploy increasing numbers of storage systems.

Historically, storage systems have been discrete *management domains*, with each one's data, client access, performance, protection, and security managed separately. But as users deploy dozens or hundreds of storage systems, tools that consolidate management of multiple systems and integrate with other data center automation facilities become a necessity.

During twelve-plus years of delivering storage systems, Pure Storage has kept pace with users' evolving capacity, security, compliance, and AI readiness needs. Today, with most of its customers deploying multiple systems, the company's "storage simplicity" mantra has been extended to *fleets* of its systems.

This brief describes **Pure Fusion™**—Pure Storage's transformation of the Purity storage operating system into a storage management platform that *federates* users' *fleets* of its products in a cloud-like paradigm that makes managing a hundred Pure Storage systems as straightforward as managing one.

THE PURE STORAGE VIEW OF STORAGE MANAGEMENT

With most enterprises experiencing information technology sprawl, the need for central management of IT resources and assets is increasingly obvious. Infrastructure teams must configure, maintain, repair, upgrade, and replenish—in short, *manage*—their organizations' servers, networks, and storage systems. Of these, storage is the only one that is irreplaceable because the value of the data it holds is an organization's repository of its intellectual property and operating state, which typically far exceeds that of the equipment.

Managing storage can be thought of as involving two types of activities:

OBSERVATION

Monitoring performance, utilization, and “health,” using analytics tools to optimize asset utilization and plan future requirements, tracking inventory and support, and so forth.

ACTION

Creating, resizing, relocating, and destroying storage resources (volumes, file systems, etc.); connecting data to clients; implementing mechanisms for risk mitigation, disaster recovery, cyberattack and human error prevention, and so forth.

RESPONDING TO THE NEED FOR CONSOLIDATED STORAGE MANAGEMENT

From its earliest days, Pure Storage has sought to make data storage easy to own and operate. Arguably, its products have established a new norm for storage management simplicity.

Historically, its customers have managed their systems individually. Administrators would create *storage resources* (e.g., volumes), connect them to clients (hosts), specify quality of service (QoS), schedule snapshots and replication, and so forth, on each system. They would enforce their internal storage standards on each system, incurring repetitive labor cost and “tech debt” for maintaining automation tools.

Because many of its customers operate dozens of its systems, Pure has developed Pure Fusion™ to consolidate management of fleets of its products. Users with multiple systems can manage them in groups, automate common tasks, and control them using other data center automation tools and products.

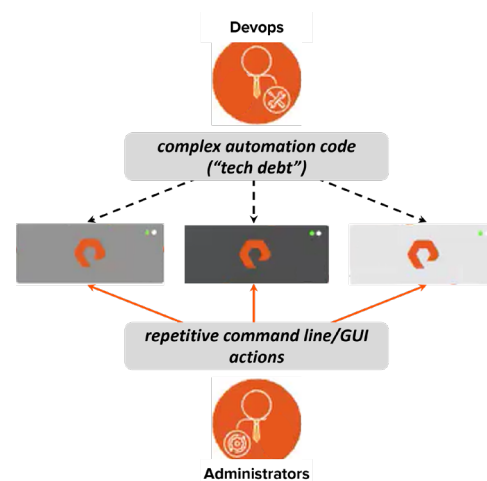


Figure 1: Conventional Storage Management

Pure Fusion™ extends the Purity storage operating system's capabilities by consolidating storage management for fleets of Pure Storage products and assisting with automation of common management tasks.

PURE FUSION™

Pure Fusion™ transforms Purity into a multi-system *storage management platform* whose facilities are accessible via API, CLI, and GUI interfaces. At its core is an *automation engine* that eliminates or simplifies many common storage management tasks. It extends Purity's administrative model by *federating* user-defined fleets of storage systems that can be managed from any fleet member system. Compared to system-by-system management, Pure Fusion provides:

FLEET-WIDE STORAGE PROVISIONING

Storage infrastructure teams can provision and manage storage **on** any system in a fleet **from** any system in the fleet.

COMPATIBILITY WITH EXISTING INFRASTRUCTURE

Pure Fusion uses fleet member resources and organizations' internal networks. No dedicated management server or external connection is needed,¹ so *dark sites* that do not permit external connections can utilize it.

Because it is part of Purity, Pure Fusion does not require separate licensing, installation, or configuration. Its functions are invoked either through application APIs or through GUI or CLI commands. Commands operate on Purity structures, utilizing existing administrative skill sets to simplify existing workflows.

EXTENSIBILITY

As initially released, Pure Fusion supports a subset of the Pure Storage portfolio and provides fleet-wide control and provisioning. Its architecture, however, is designed to encompass Pure's entire product portfolio in a single control plane and to deliver additional capabilities over time.

With Pure Fusion, IT organizations can meet current and future challenges to data storage management at scale efficiently and without increasing cost. It limits susceptibility to cyberattacks by enforcing compliance with organizations' data protection policies and provides a foundation for artificial intelligence use cases that are fast becoming a mainstay of enterprise information technology. The sections that follow describe Pure Fusion concepts and illustrate how they can be used to manage Pure Storage systems.

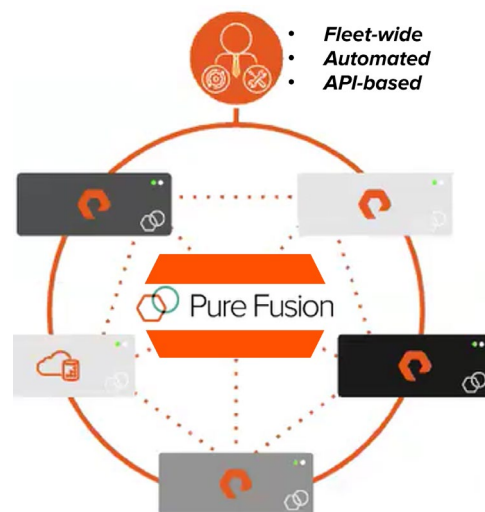


Figure 2: Storage Management with Pure Fusion

¹ Some advanced features, such as workload placement advice, require Pure1 *phonehome* access.

STORAGE MANAGEMENT CONSOLIDATION: FLEETS

With Pure Fusion, storage management domains are user-defined *fleets* of Pure Storage systems. A fleet might consist of the systems in a data center, systems located throughout a campus, or a mix of on-premises systems and Cloud Block Store (CBS) instances in a public cloud.

With Pure Fusion, administrators authenticated by LDAP or AD can view and manage storage on any fleet member system. Locally authenticated administrators have access only to their local arrays.

Pure Fusion *federates* the systems that make up a fleet. A fleet administrator or automation tool can provision storage **on** any fleet member **from** any fleet member, but member systems can also be managed locally.² For example, to provision storage for an application that uses FlashArray volumes, a fleet administrator specifies the volume parameters (name, size) and a *context*³ (system on which to create them). Pure Fusion creates the volumes on the indicated system.

CREATING A FLEET

To create a fleet, an array administrator on any system that will become a member specifies a name for the fleet. The system on which a fleet is created becomes the fleet's first member.

Figure 3 illustrates the use of the Purity GUI to create a fleet.⁴

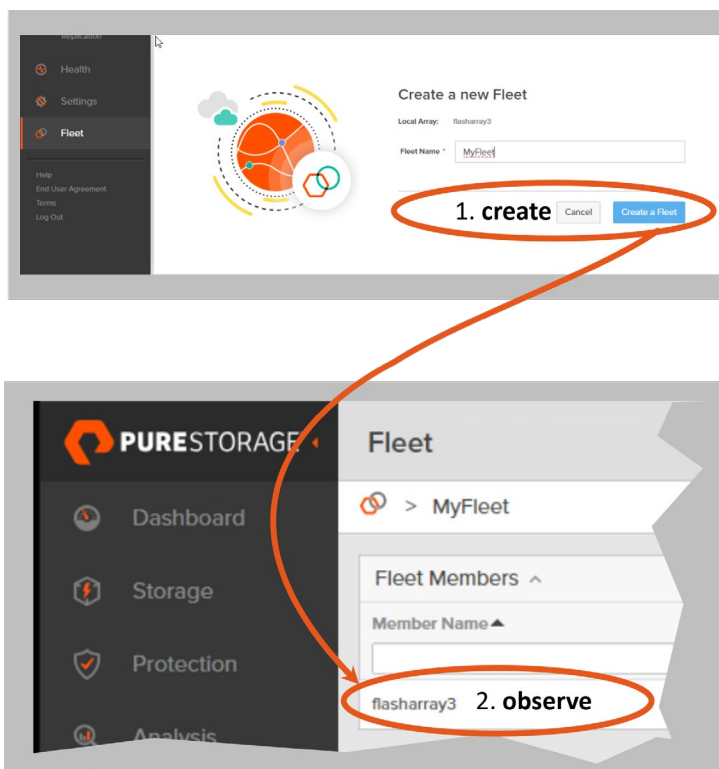


Figure 3: Creating a Fleet

² This brief uses *fleet administrator* to refer to administrative accounts authenticated by an organization's central identity provider, regardless of role. *Local administrator* refers to accounts authenticated by an array. Only fleet administrators are permitted to manage storage on remote fleet members.

³ In action commands, *context* refers to the fleet member on which execution takes place. In informational commands, it refers to the member that holds the requested information. With fleet structures such as *storage presets* (page 9), context refers to the entire fleet.

⁴ The Appendix (page 12) illustrates use of the CLI to perform operations similar to those shown in Figures 3-6.

ADDING MEMBERS TO A FLEET

To add a system to a fleet (called *joining*), an array administrator on any fleet member generates a single-use *fleet key*. Fleet keys are valid for one hour after they are generated.

The member array administrator provides the fleet key to an array administrator of the system to be added. (In many cases, the same individual manages both arrays in separate GUI or CLI sessions.)

The administrator of the system to be added pastes the key into the **Join an Existing Fleet** GUI dialog, supplies the fleet name, and clicks the **Join a Fleet** dialog button as in Figure 4.

Once a system has joined a fleet, fleet administrators logged in to any fleet member can view its configuration (hosts and host groups, volumes, pods, snapshot history, etc.), and use the Purity GUI, CLI, or Purity's REST APIs embedded in tools and scripts to perform management tasks. Third party tools such as Veeam Backup and VMware vCenter can manage storage on any fleet member via the member to which they are connected.

LOCAL ARRAY VS FLEET-WIDE MANAGEMENT

Pure Fusion simplifies administrator, management application, and custom script workflows. Locally authenticated administrators manage storage on their local systems; fleet administrators can issue management commands to be executed anywhere in the fleet. The fleet member on which a command executes is called the command's *context*.

To manage storage on remote fleet members, fleet administrators either:

- ▶ Select the **Fleet View** button on GUI pages used to manage resources (cf. Figure 5).

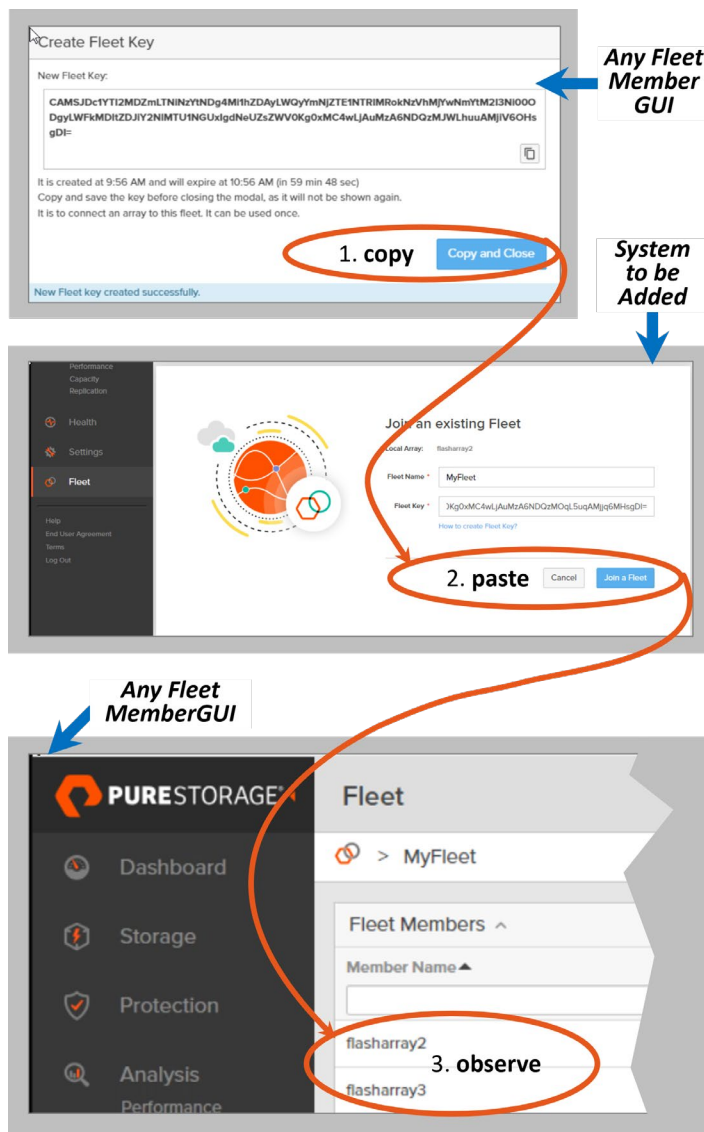


Figure 4: Adding a Member System to a Fleet

- Add `--context` options to CLI commands addressed to remote members.⁵

Purity's REST APIs include a similar context parameter. GUI pages, CLI commands, and REST APIs are otherwise identical to their single-system equivalents.

FLEET MANAGEMENT VIEWS

When **Fleet View** is selected, resource lists include a **Context** column that indicates the location of each listed resource and information about it. For example, Figure 5 is a snippet of the Fleet View **Hosts** page of a FlashArray GUI **Storage** display. The **Context** column lists the fleet member on which each host object is defined.

The GUI **Storage** display includes a page for each type of manageable resource.⁶ For

example, Figure 5 shows the first few lines of a **Hosts** page that displays a list of hosts and host groups and the context (fleet member) that holds information about them.

Fleet View simplifies conceptually simple management tasks that become time-consuming and error-prone in large fleets. For example, to quickly identify the system that hosts a particular volume, a fleet administrator on any fleet member selects the Fleet View **Volumes** page and sorts the list on the serial number column.

PROVISIONING STORAGE RESOURCES IN A FLEET

With Pure Fusion, administrators can provision storage resources on any fleet member from anywhere in a fleet. An administrator simply specifies the target fleet member on which to place the resources. For example, Figure 6 illustrates use of the Purity **Create Multiple Volumes** wizard on **flasharray2** to create five identical volumes on fleet member **flasharray1**. Also noteworthy is that volume creation and addition to a pod are accomplished in a single step. The **flasharray3** GUI **Storage** page on the right shows the created volumes and their contexts (fleet members on which they reside).

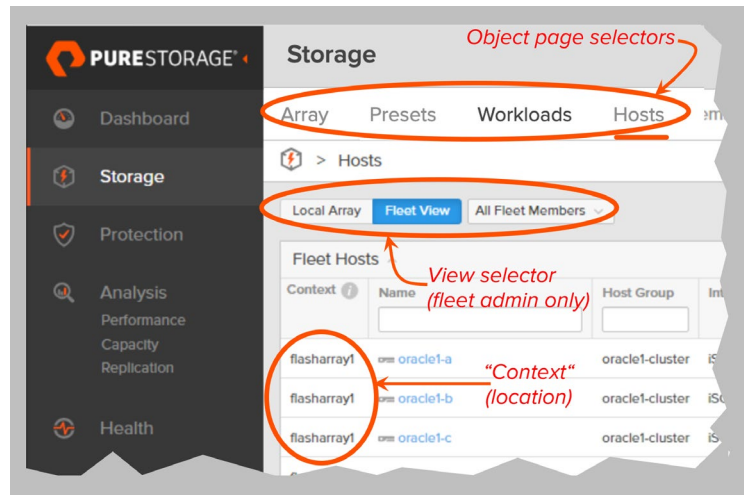


Figure 5: Fleet View of Hosts

⁵ Pages displayed to local administrators do not include the `Local Array` `Fleet View` buttons. CLI commands issued by local administrators do not allow the `--context` option.

⁶ The initial Pure Fusion release does not support fleet-wide management of all resources (e.g., **object buckets**).

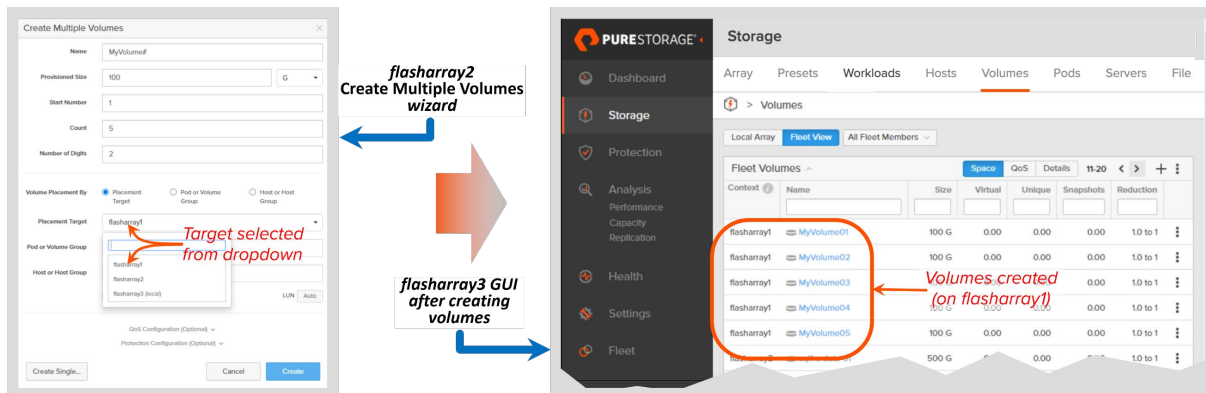


Figure 6: Creating Volumes on a Remote Fleet Member

Pure Fusion takes the guesswork out of configuring storage resources. For example, as Figure 6 illustrates, the GUI dialog for creating volumes and placing them in pods only displays eligible hosts and pods, eliminating potential sources of frustration and error, especially in large fleets. The utility and time savings with this and similar features are exponentially greater with larger fleets.

WORKLOADS AND STORAGE PRESETS

The capabilities described in the preceding section are useful for “one-off” and nascent applications. But many organizations frequently deploy instances of common applications (e.g., databases) whose storage needs are well-understood. Two additional Pure Fusion concepts—*workloads* and *storage presets*—to be introduced soon after initial rollout, simplify storage provisioning and ensure consistent adherence to organization policies for such applications.

WORKLOADS

A Pure Fusion *workload* is the set of storage resources (e.g., volumes) that an application requires to function. It necessarily includes one or more resources whose properties (e.g., size, QoS parameters, ...) are specified in the storage preset used to create it.

Workload Provisioners (typically in the storage administrator role⁷) specify *storage presets* (discussed below) —templates of workload specifications—for use with the **Configure New Workload** wizard to create workloads for frequently-deployed applications. Figure 7 shows the **Configure New Workload** wizard used to create an Oracle database workload from a simple preset. The rightmost pane of the wizard displays the resources and properties specified in the preset.

⁷ Locally authenticated administrators are limited to configuring workloads on their local arrays.

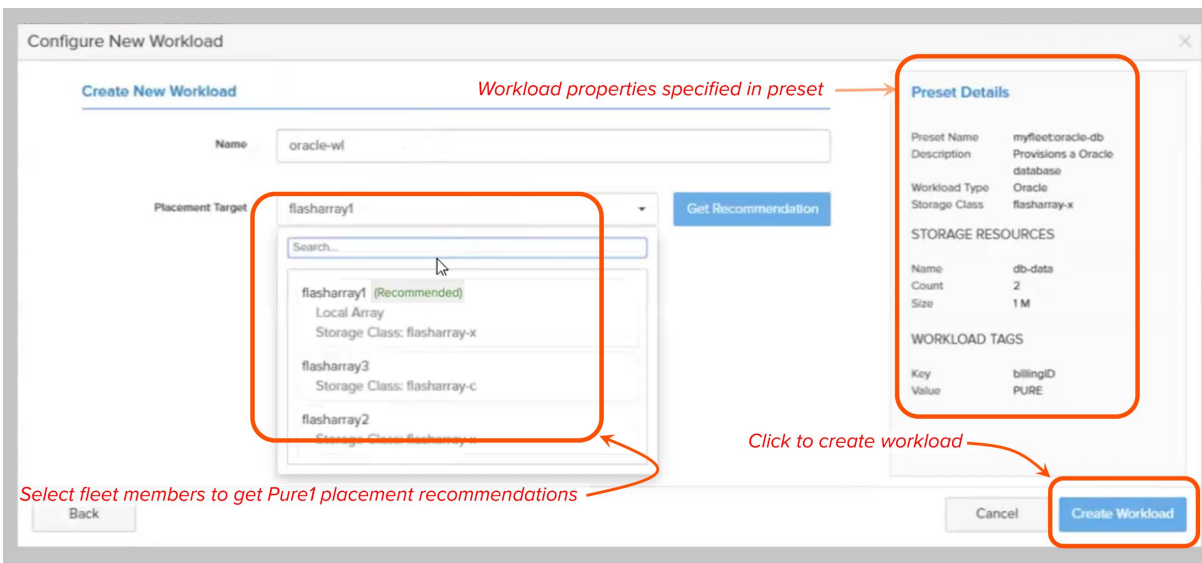


Figure 7: Configure New Workload Wizard

WORKLOAD PLACEMENT

In large fleets with hundreds of storage resources, identifying the optimal location for a new workload can be challenging. In systems connected to the Pure1 cloud, the Pure1 **Workload Planner** can assist with placement decisions by analyzing fleet members designated by the Workload Provisioner and making placement recommendations based on projected impact of the new workload on each one. Provisioners click the **Get Recommendation** button to select one or more fleet members from a list and request recommendations from the Pure1 cloud. Pure1 responds by flagging the most suitable fleet members for placement as **Recommended** and others that satisfy the workload's criteria (e.g., storage classes, utilization, ...) as **Acceptable**. Members that do not satisfy the criteria are not flagged.

The Provisioner selects a placement target from the list returned by Pure1 as shown in Figure 7.

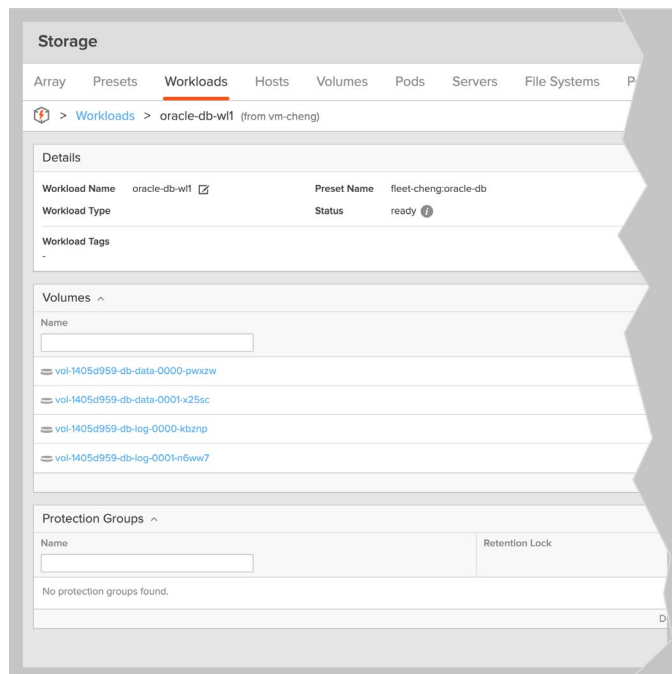


Figure 8: Viewing Workload After Creation

When a fleet member has been selected for placement of the new workload, the Workload Provisioner clicks the  button (Figure 7.) to cause Purity on the selected member to create the workload's storage resources. After provisioning, the workload can be viewed in the GUI **Storage** page's **Workloads** view as illustrated in the snippet in Figure 8. Storage administrators can add resources to workloads as needed. Adding a resource to an existing workload does not change the storage preset used to create it.

The **Configure New Workload** wizard simplifies workload deployment and promotes adherence to organizations' storage resource policies, including size, type, QoS, and data protection. In addition, it supports user-defined *workload tags* with which users can classify workloads for their own internal purposes such as chargeback and billing.

STORAGE PRESETS

Storage presets are fleet-wide persistent templates defined by **Preset Authors** (in the storage administrator role) and used by the **Configure New Workload** wizard to guide Workload Provisioners through workload provisioning. Presets express resources and properties as values or as ranges, for example minimum and maximum number of volumes of a given type, upper and lower limits on provisioned capacity, etc.

Preset Authors create storage presets on one fleet member and Pure Fusion propagates them throughout the fleet. They specify values and/or allowable ranges for properties including:

- ▶ **Workload type**
(e.g., database, analytics, ...) Used by Pure1 to make placement recommendations.
- ▶ **Storage resource type(s)**
Names, numbers, and capacity ranges for the workload's storage resources (e.g., volumes, file systems, ...).
- ▶ **Storage class(es)**
System type(s) (e.g., FlashArray//X, FlashArray//C, ...) on which a workload's storage resource may be provisioned.
- ▶ **QoS limits**
Allowable IOPS and/or bandwidth ranges for each type of storage resource.
- ▶ **(Optional) data protection**
Snapshot and/or replication schedules for each type of storage resource.
- ▶ **(Optional) workload tags**
User-defined tags (described below) used to assist in managing workloads.

Authors use Pure Fusion's **Preset Authoring Wizard**, illustrated in Figure 9, to develop storage presets for Workload Provisioners to use in creating workloads that meet their organizations' storage needs for frequently-deployed applications.

The screenshot shows the 'Create New Preset in Fleet' wizard. It is divided into three main sections:

- 1. Preset Details:** Includes fields for 'Preset Name' (oracle-db) and 'Description (optional)' (Provisions a Oracle database). Below this are expandable sections for 'DEFINE PARAMETERS AND CONFIGS'.
- 2. Workload Details:** Includes 'Workload Type' (Oracle), 'Storage Class' (flasharray-x), and a 'Storage Resources' section. The 'Storage Resources' section has fields for 'Name' (Letters, Numbers, -), 'Count', 'Provisioned Size' (Positive numbers, G), and 'Snapshot Configurations'. A red arrow points to this section with the text 'Specify properties for storage resource'.
- Review:** A summary of the preset configuration, including 'Preset Name', 'Description', 'Workload Type', 'Storage Class', and a list of 'STORAGE RESOURCES' (db-data, 2, 1 M). A red arrow points to this section with the text 'Running summary of creation process'.

At the bottom, there are two buttons: 'Add to Preset' and 'Create'. A red arrow points to the 'Add to Preset' button with the text 'Add storage resource to preset'. Another red arrow points to the 'Create' button with the text 'Create preset'.

Figure 9: The Storage Preset Authoring Wizard

The **Preset Authoring Wizard** contains three sections:

PRESET DETAILS

Used by Preset Authors to specify preset-wide options, including snapshot and replication enabling, and reusable (multi-valued) parameters.

WORKLOAD DETAILS

Used by Preset Authors to specify type, number, and storage class for the workload's storage resources. Authors enter details for each type of resource and click the **Add to Preset** button to add it to the preset.

REVIEW

A running display of preset properties during the authoring process.

When all resources and parameters have been specified, the author clicks the **Create** button to cause Purity to persist the preset and propagate it to all fleet members.

WORKLOAD TAGS

Workload Provisioners can use *resource tags* to classify workloads. For example, tags can identify workload criticality, assist with intra-organizational chargeback, etc.

Managed service providers can use them to implement usage-based customer billing.

Workload tags are key-value pairs. The value associated with a tag's key may be:

- ▶ A single value, e.g., **billingID=acme-sales** (cf. Figure 10).
- ▶ An enumerated list, e.g., **tier=("Mission Critical", "Dev-Test")**
- ▶ A free-form numeric or alphanumeric value to be entered by Workload Provisioners.

Figure 10 illustrates using the Authoring wizard to add a single-value tag with key **billingID** and value **acme-sales** to a storage preset. All workloads created using the preset will be tagged **acme-sales**. Administrators can sort the **Workload** view of the GUI **Storage** page to quickly identify workloads associated with a given tag.

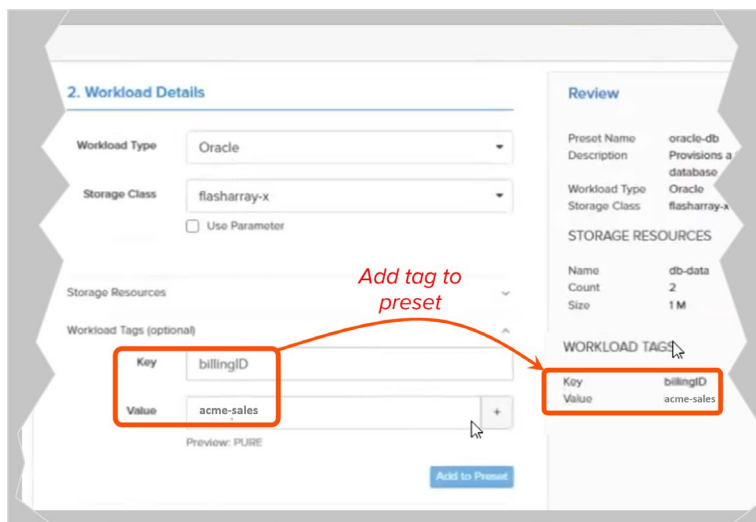


Figure 10: Adding a Workload Tag to a Preset

WORKLOAD PARAMETERIZATION

Preset Authors can *parameterize* most workload properties during preset creation to permit flexible workload configurations without proliferating the number of presets. For example, the number and sizes of storage resources, storage classes on which each type of resource can be allocated, and workload tags can all be parameterized to allow Workload Provisioners to select from numeric ranges or lists of values specified in the preset.

Parameterization is optional. When Preset Authors do not specify it, all workload instances created from the preset have identical properties.

Figure 9 illustrates a sample pane from the preset authoring wizard. Authors can specify several types of flexibility such as:

- ▶ One or more storage resource types, each with ranges for number of resources, resource capacity, and QoS limits.
- ▶ *Storage classes* (system types) on which resources in workloads based on the preset can be located.

- ▶ (Optional) snapshot and/or replication schedules and retention for workloads based on the preset.
- ▶ (Optional) *workload tags* with which to classify workloads.

Using the **Configure New Workload** wizard to create workloads defined by storage presets simplifies storage deployment while ensuring that organizational standards are met.

Presets are especially useful for deploying instances of common applications in large fleets. For example, as Figure 7 suggests, an organization's Oracle databases might utilize a data volume and a log volume. Production deployments would typically include snapshots and/or replication, whereas development and testing might forego those, Workload Provisioners can adjust preset parameters to meet each deployment's requirements.

AUDITING

A fleet administrator on any fleet member can inspect the audit logs of any member system. Figure 11 shows views logs for three fleet members (**flasharray1**, **flasharray2**, and **flasharray3**) viewed from a **flasharray1** CLI console by specifying values for the `--context` parameter.

```
arrayadmin@flasharray1> pureaudit list --context flasharray3
```

ID	Time	User	Origin	Command	Subcommand
396	2024-06-23 23:58:35 UTC	jjones	flasharray3	purefleet	create
397	2024-06-23 23:59:08 UTC	jjones	flasharray3	purefleet	create

```
arrayadmin@flasharray1> pureaudit list --context flasharray2
```

ID	Time	User	Origin	Command	Subcommand
406	2024-06-24 00:00:15 UTC	jjones	flasharray2	purefleet	member add
410	2024-06-24 00:02:50 UTC	jjones	flasharray2	purefleet	create

```
arrayadmin@flasharray1> pureaudit list --context flasharray1
```

ID	Time	User	Origin	Command	Subcommand
475	2024-06-24 00:03:31 UTC	jjones	flasharray1	purefleet	member add

```
arrayadmin@flasharray1>
```

Figure 11: Viewing Local and Remote Audit Logs

The events in the logs shown in Figure 11 are:

FLASHARRAY3

Creation of fleet **MyFleet** on **flasharray3**.

Generation of a fleet key to be used to add **flasharray2** to **MyFleet**.

FLASHARRAY2

Addition of member **flasharray2** to **MyFleet** using the fleet key generated on **flasharray3**.

Generation of a fleet key on **flasharray2** to be used to add **flasharray1** to **MyFleet**.

FLASHARRAY1

Addition of member **flasharray1** to **MyFleet** using the fleet key generated on **flasharray2**.

An administrator can specify multiple contexts in a single command. For example:

```
--context flasharray1,flasharray2,flasharray3
```

For security reasons, log entries do not include the single-use fleet keys.

PURE FUSION: THE MODERN APPROACH TO STORAGE MANAGEMENT

Pure Fusion streamlines management workflows that create storage resources, both for frequently-deployed applications and for ad hoc resource provisioning. Working **from** any of a fleet's member systems, Workload Provisioners can direct Pure Fusion to provision storage **on** any of the fleet's members.

Pure Fusion's mechanism for enforcing organizational workload standards is *storage presets*—customizable templates that specify sets of storage resources and their properties. Presets specify properties—allowable numbers, sizes, system types, QoS, and so forth—for the storage resources that make up workloads. To provision storage for an application, Workload Provisioners invoke the **Configure New Workload** wizard, select a storage preset and supply values for any adjustable parameters. Pure Fusion creates the resources and, if the Pure1 Workload Planner is accessible, advises on optimal placement in the fleet. When created with storage presets, workloads comply with the organizational standards expressed in the presets.

As with all Pure Storage innovations, Pure Fusion provides significant user value at introduction, but its underlying structure enables the company to deliver additional capabilities over time. For example, Pure Fusion initially provides fleet management and block volume provisioning for FlashArray systems and Cloud Block Store (CBS) virtual arrays. Storage preset and workload support are expected to be available shortly thereafter. Over time, the company intends to deliver Pure Fusion support for its entire portfolio of FlashArray and FlashBlade systems, and to add data mobility, compliance verification, further integration with Pure1, and other capabilities that enhance and simplify consolidated storage management for multiple arrays.

Pure Fusion greatly reduces the complexity and number of ad hoc decisions required to deploy storage for both common application workloads and ad hoc requirements. It increases administrator productivity, eliminates most human errors, and greatly reduces the code complexity (so-called “tech debt”) required to integrate storage provisioning with data center workflow automation tools.

APPENDIX:

USING THE PURITY CLI WITH PURE FUSION

The Purity CLI is functionally equivalent to the GUI. It is often used to create scripts that perform frequently-repeated administrative operations. Some administrators simply prefer working with command line interfaces. This appendix illustrates use of the CLI to perform some basic fleet administrative operations similar to those shown in Figures 3 through 6.

CREATING A FLEET (CF. FIGURE 3 ON PAGE 3)

Figure 12 illustrates creation of **MyFleet**. The system on which an array administrator issues the command becomes the first fleet member.

```
arrayadmin@flasharray2> purefleet create MyFleet
Name
MyFleet
```

Figure 12: Creating a Fleet

ADDING A MEMBER TO A FLEET (CF. FIGURE 4 ON PAGE 4)

Figure 13 illustrates adding member system **flasharray1** to **MyFleet**. Prior to the addition the fleet consists of members **flasharray2** and **flasharray3**.

The **--fleet-key** option in the command indicates that the administrator is to supply the key when prompted by the CLI.

The administrator obtains a one-time fleet key from one of the fleet's member administrators and pastes it into the CLI console when prompted. Pure Fusion adds **flasharray1** to the fleet and lists the fleet's expanded membership.

Array administrators can also remove systems from a fleet and can delete a fleet when (a) it contains only a single member and (b) all of the fleet's storage presets have been deleted.

```
arrayadmin@flasharray1> purefleet member list
Fleet      Name      Status
MyFleet    flasharray2  joined
MyFleet    flasharray3  joined

arrayadmin@flasharray1> purefleet member add --fleet MyFleet
--fleet-key
Enter the fleet key of the target fleet:
CAQSDQxMwMwMWE1LTBiNmMtNDYwOC1hOGYyLTNjZDUzYzUwNDdLOBokNDEx
YzAxYTUtMGt2Y0NjA4LWE4ZjItM2NkNTNjNTA0N2U4IgdNeUZsZWV0Kg0xMC
4wLjAuMzA6NDQzMmGduryEMjjB+pW+hDI=
Fleet      Name      Status
MyFleet    flasharray3  joined
MyFleet    flasharray2  joined
MyFleet    flasharray1  joined
arrayadmin@flasharray1>
```

Figure 13: Adding a System to a Fleet

LISTING HOSTS ON A REMOTE SYSTEM (CF. FIGURE 5 ON PAGE 5)

Figure 14 illustrates use of the **flasharray2** CLI to display a list of the host objects on fleet member **flasharray1**.

Except for the **--context** option, which allows administrators to specify a list of arrays whose hosts or volumes are to be listed, the command is identical to that used to list the host objects on a local array.

```
# List hosts on flasharray1
arrayadmin@flasharray2> purehost list --context flasharray1
```

Context NQN	Name Host Group	WWN	IQN
flasharray1	oracle1-a	-	iqn.1994-05.com.redhat:2989f3e00967
		-	oracle1-cluster
flasharray1	oracle1-b	-	iqn.1994-05.com.redhat:2989f3e00968
		-	oracle1-cluster
flasharray1	oracle1-c	-	iqn.1994-05.com.redhat:2989f3e00969
		-	oracle1-cluster

```
arrayadmin@flasharray2>
```

Figure 14: Listing a Remote Fleet Member's Hosts

CREATING VOLUMES ON A REMOTE SYSTEM (CF. FIGURE 6 ON PAGE 6)

Figure 15 illustrates use of the **flasharray2** CLI to create volumes **CLIVolume-003** and **CLIVolume-004** on target system (context) **flasharray1**. Except for the **--context** option, the commands are identical to those used to create volumes on a local array.

A separate command following volume creation lists the newly created volumes.

```
# Create CLIVolume-003
arrayadmin@flasharray2> purevol create -context flasharray1
--size 50g CLIVolume-003
```

Name	Size	Source	Created	Serial
CLIVolume-003	50G	-	2024-07-17 22:40:26 UTC	
C2698D47EC20410900013F49		pgroup-auto		

```
# Create CLIVolume-004
arrayadmin@flasharray2> purevol create --context flasharray1
--size 50g CLIVolume-004
```

Name	Size	Source	Created	Serial
CLIVolume-004	50G	-	2024-07-17 22:40:46 UTC	
C2698D47EC20410900013F4A		pgroup-auto		

```
# List volumes on flasharray1
arrayadmin@flasharray2> purevol list --context flasharray1
```

Location Created	Name Serial	Size	Source
{'name': 'flasharray1'} 2024-07-17 22:40:26 UTC	CLIVolume-003 C2698D47EC20410900013F49	50G	-
{'name': 'flasharray1'} 2024-07-17 22:40:46 UTC	CLIVolume-004 C2698D47EC20410900013F4A	50G	-

Figure 15: Creating Volumes on a Remote Fleet Member



Laying the Groundwork for the Future of Storage Management

© 2024 Pure Storage

Pure Storage, the Pure Storage P Logo, Pure Fusion, and the marks on the Pure Storage Trademark List are trademarks or registered trademarks of Pure Storage, Inc. in the U.S. and other countries. The Pure Storage Trademark List can be found at [purestorage.com/trademarks](https://www.purestorage.com/trademarks).

Use of Pure Storage Products and Programs are covered by End User Agreements, IP, and other terms, available at: <https://www.purestorage.com/legal/productenduserinfo.html> and <https://www.purestorage.com/patents>

The Pure Storage products described in this documentation are distributed under a license agreement restricting the use, copying, distribution, and decompilation/reverse engineering of the products. The Pure Storage products described in this documentation may only be used in accordance with the terms of the license agreement. No part of this documentation may be reproduced in any form by any means without prior written authorization from Pure Storage, Inc. and its licensors, if any. Pure Storage may make improvements and/or changes in the Pure Storage products and/or the programs described in this documentation at any time without notice.

The development, release, and timing of any features or functionality described for Pure products remains at Pure's sole discretion. The information provided is for informational purposes only and is not a commitment, promise, or legal obligation to deliver any material, code, or functionality.

THIS DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. PURE STORAGE SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.