

SAS[®] Viya[®] Platform: Backup and Restore

2021.1.6 - 2022.09

This document might apply to additional versions of the software. Open this document in [SAS Help Center](#) and click on the version in the banner to see all available versions.

<i>Backup and Restore: Overview</i>	2
Introduction to Backup and Restore	2
How to Use This Document	3
What Is Backed Up?	3
What Is Not Backed Up?	4
Multi-Tenancy Support	4
<i>Backup and Restore: Concepts</i>	5
Backup and Restore Terms	5
<i>Backup and Restore: Perform an Ad Hoc Backup</i>	7
What Backup Scenarios Are Not Supported?	7
Best Practices for Performing Backups	7
Perform an Ad Hoc Backup	9
Useful Commands	10
List All Backups	10
Get Status of a Backup	10
Get Details of a Backup	10
View Backup Logs	11
<i>Backup and Restore: Schedule a Backup</i>	11
Overview	11
Change the Default Backup Schedule	11
Schedule a Backup	12
List Scheduled Backups	13
Unschedule a Backup Schedule	13
Change a Backup Schedule	14

Backup and Restore: Perform a Restore	14
What Restore Scenarios Are Not Supported?	14
Best Practices for Performing Restores	15
Perform a Restore	16
Post-Restore Validation	20
List Restores	20
Show Details of a Restore	21
View Restore Logs	21
Backup and Restore: Purge Backups	21
Purge Backups	21
Backup and Restore: Recover from a Disaster	22
Overview	22
Disaster Recovery Prerequisites and Planning on the Source Environment	23
Multi-Tenancy Support	24
Perform a Disaster Recovery	24
Post-Restore Validation	29
Backup and Restore: Administrator Tasks	30
Backup and Restore: Reference	31
Backup Details	31
How to Find the Physical Location of the Persistent Volume (PV) in an Azure Environment	33
Examples of Transferring Backup Data between the SAS Viya Platform Environment and the Local Filesystem	34
Copy Backup Data from Azure Environment to Azure Environment	37
Determine the Backup ID of the Last Successful Backup	38
Perform an Inventory Scan	39
Sizing Recommendations for Restore	42
Backup and Restore: Troubleshooting	43
Logs	43
Additional CrunchyData PostgreSQL Jobs and Associated Pods Created After Executing kubectl apply Commands	44
Backup Fails for fileSystem Source When Using Azure Premium Storage	45
Backup Fails When Services Are Not Completely Started	45

Backup and Restore: Overview

Introduction to Backup and Restore

The SAS Viya platform can be deployed with other SAS software at your site to gain insight from analytics and business intelligence. The SAS Viya platform consists of many software components, servers, and services (including SAS Cloud Analytic Services (CAS) Server, SAS Compute Server, SAS Launcher Server, Identities service, and Audit service). You should have a strong backup and restore strategy and validate your backup and restore processes regularly. A backup strategy should include tasks for backing up all software components and data in addition to running the backup job.

How to Use This Document

Here is a strategy for using this document:

See [“Best Practices for Performing Backups” on page 7](#) for information about when to perform backups, backup configuration properties, and other backup settings and properties.

See [“What Is Not Backed Up?” on page 4](#) for your awareness about content that is not backed up.

See [“Disaster Recovery Prerequisites and Planning on the Source Environment” on page 23](#) for recommended ongoing tasks to prepare for a disaster recovery.

See [“Backup and Restore: Administrator Tasks” on page 30](#) for additional administrative tasks.

See [“Multi-Tenancy Support ” on page 4](#) for information about multi-tenancy.

What Is Backed Up?

- Content that users saved in the SAS Infrastructure Data Server is backed up. Content can be reports (including SAS Visual Analytics reports), comments, authorization rules, attachments, audit records, user preferences, and data source definitions.
- Configuration settings and definitions for infrastructure and applications that you defined based on your business requirements are backed up. These settings are saved in SAS Configuration Server (Consul). Application configuration settings include user interface (UI) themes (for example, the SAS Visual Analytics UI theme) and batch run date (for example, batch date of running SAS Visual Analytics reports).
- If CAS is configured for data management and analytics, CAS access controls and global caslibs are backed up. Caslibs help with accessing data from a data source. A caslib consists of a data source definition and can include access controls. Access controls help with controlling user access to the data.

In a clustered environment, caslibs and access controls of the CAS primary node are backed up.
- From the CAS data persistent volume (PV), the subdirectory `/cas/data/caslibs` is backed up. In a non-multi-tenant environment, there is one CAS data PV named `cas-default-data`. In a multi-tenant environment, there is one CAS data PV for every tenant named `cas-tenant_name-data`.

Note:

Multiple persistent volumes (PVs) are used for backup.

Run this command to list the PVCs where backup is stored:

```
kubectl -n namespace get pvc -l "sas.com/backup-role=storage"
```

What Is Not Backed Up?

Backup does not support the backup of several items. You must take alternative steps to back up these items.

- Kubernetes manifest and related items.
- User home directories.
- Other data sources, such as:
 - Third-party databases.
 - Data that is stored outside of the SAS Infrastructure Data Server or PostgreSQL server.
 - SAS Infrastructure Data Server or PostgreSQL server metadata such as database user information, roles, and permissions that are stored in the server.
 - Data that is loaded to CAS. This is in-memory data that is not saved to a persistent volume (PV).
 - Data that is stored on remote file systems (such as path-based data sources including PATH and DNFS).
- Any PVs mounted that are not already explicitly included in the list of backed up items [here on page 3](#).

Multi-Tenancy Support

What Multi-Tenancy Configurations Are Supported?

Backup and restore are supported between environments that are configured for multi-tenancy. In version LTS 2021.2 of the SAS Viya platform, the supported scenario is a backup of a full SAS Viya 4 multi-tenant environment (provider and all customer tenants are backed up simultaneously) and restore of this environment to a SAS Viya 4 multi-tenant environment where the target environment is a matching environment, meaning that the target environment:

- contains all of the same tenants that are in the backup after being restored. The tenants must be onboarded in the target environment before the backup is restored. The provider content is always restored.

Note:

- Do not restore to a target environment that has tenants that are not in the backup.
- You must onboard all of the tenants available in the backup before restoring them to the target environment.
- The tenant names must be the same between the source and target environment.

-
- uses the same cloud provider as in the backup.

- has a matching product set as in the backup. The restore might fail if the set of products in the backup does not match the set of products available in the target environment.
- uses the same database option (schemaPerApplicationTenant or databasePerTenant) as in the backup.

Backup and Restore: Concepts

Backup and Restore Terms

Common Terms

Here is a list of common backup and restore terms:

ad hoc backup

the backup that can be initiated by a Kubernetes administrator on an as-needed basis.

backup

the backup that can be initiated by a Kubernetes administrator. For more information, see [“Backup Details” on page 31](#).

Consul

also known as SAS Configuration Server. An open-source server by Hashicorp. It is used for service discovery and configuration, and is characterized by its high availability and scalability.

job time-out

the number of minutes before a job times out.

persistent volume (PV)

a cluster-wide resource that you can use to store data in a way that it persists beyond the lifetime of a pod.

persistent volume claim (PVC)

a request for storage, which is met by binding the PVC to a persistent volume (PV).

pod

represents a unit of deployment—a single instance of an application in Kubernetes—that can consist of either a single container or a few containers that are tightly coupled and share resources.

PostgreSQL server

a relational database management system (RDBMS) that stores user content such as reports, comments, authorization rules, attachments, audit records, user preferences, and some data source definitions.

pre-restore validation

validation done by the restore job before performing a restore. A pre-restore validation checks the following:

- Does the provided backup exist?
- Is the backup completed?

- Is the backup purged?

recovery point objective (RPO)

the interval of time that might pass during a disruption before the quantity of data lost during that period exceeds the organization's maximum allowable threshold or "tolerance."

restore

the process of restoring a backup.

retention period

the number of days that backups are stored before they are eligible for purge. Default is 30 days.

scan

the process of performing an inventory of a SAS Viya platform deployment. For more information, see the steps under ["Perform an Inventory Scan" on page 39](#).

SAS Infrastructure Data Server

an instance of *PostgreSQL server* that is provided with the SAS Viya platform.

scheduled backup

the backup that can be periodically scheduled by a Kubernetes administrator based on business requirements.

status

status of a backup or restore. The possible values are:

- *Completed* indicates that the backup or restore operation has completed successfully.
- *CompletedWithWarning* indicates that the backup or restore operation has completed with warnings.
- *Failed* indicates that a backup or restore for one or more data sources failed.

Terms for Multi-Tenant Deployment

Here is a list of backup and restore terms that are used in a multi-tenant deployment:

deployment backup

a multi-tenant deployment in which only a Kubernetes administrator can initiate backup for multiple tenants. If an explicit list of tenants is not provided, all onboarded tenants are backed up.

deployment restore

a multi-tenant deployment in which only a Kubernetes administrator can initiate a restore of a deployment backup (with at least one successful tenant backup) for multiple tenants. If an explicit list of tenants is not provided, all onboarded tenants and the provider tenant are restored. If all the tenants in the list are in the backup, then the restore is triggered for each of the tenants in the list. If any of the tenants in the list are not in the backup, then the restore does not proceed and it is marked as failed.

multi-tenant deployment

a SAS Viya platform deployment in which multiple tenants can access the same environment in isolation without impacting the data or processes of other tenants. A multi-tenant deployment has the provider tenant by default.

provider environment

a collection of software and the infrastructure to support one or more tenant environments.

provider tenant

the initial tenant (that is, tenant zero) created when a multi-tenant system is deployed. This tenant has full access to all applications in the deployment, but is intended for Kubernetes administrator access only. Users in this tenant have access to information about the entire deployment, including other tenants.

tenant

one of the customers using a shared SAS Viya platform deployment.

In a multi-tenant deployment, a tenant is said to be onboarded when the SAS Viya platform infrastructure for that tenant is created. This includes the LDAP groups, the LDAP identities, the SAS Infrastructure Data Server databases, the schemas, and the CAS instance.

tenant environment

a collection of software and the infrastructure for use by a single tenant. Customers might have separate tenant environments to support development, staging, and production instances of SAS services.

Backup and Restore: Perform an Ad Hoc Backup

Note: If you are in a multi-tenant environment, see [“Multi-Tenancy Support”](#) on page 4 for information about multi-tenancy.

What Backup Scenarios Are Not Supported?

- Restoring a backup from a different cadence is not supported. For example, you can restore to a SAS Viya platform Stable environment from a backup of a SAS Viya platform Stable environment. You can restore to a SAS Viya platform Long-Term Support environment from a backup of a SAS Viya platform Long-Term Support environment. But you cannot restore to a SAS Viya platform Long-Term Support environment from a backup of a SAS Viya platform Stable environment. See [“Software Releases and Versions”](#) in *Getting Started with SAS Viya Platform Operations* for more information.
- Restoring a backup from a different cadence version is not supported. For example, you can restore to a SAS Viya platform 2021.2.3 environment from a backup of a 2021.2.3 environment. But you cannot restore to a 2021.2.3 environment from a backup of a 2021.2.2 or earlier environment. For example, you cannot restore from a 2021.2.2 environment to a 2021.2.3 or newer environment. See [“Software Releases and Versions”](#) in *Getting Started with SAS Viya Platform Operations* for more information.

Best Practices for Performing Backups

- To perform some elements of this task, you must have elevated Kubernetes permissions.
- Restoring a backup from a different cadence is not supported. For example, you can restore to a SAS Viya platform Stable environment from a backup of a SAS Viya platform Stable environment. You can restore to a SAS Viya platform Long-Term Support environment from a backup of a SAS Viya platform Long-Term Support environment. But you cannot restore to a SAS Viya platform

Long-Term Support environment from a backup of a SAS Viya platform Stable environment. See [“Software Releases and Versions”](#) in *Getting Started with SAS Viya Platform Operations* for more information.

- Restoring a backup from a different cadence version is not supported. For example, you can restore to a SAS Viya platform 2021.2.3 environment from a backup of a 2021.2.3 environment. But you cannot restore to a 2021.2.3 environment from a backup of a 2021.2.2 or earlier environment. For example, you cannot restore from a 2021.2.2 environment to a 2021.2.3 or newer environment. See [“Software Releases and Versions”](#) in *Getting Started with SAS Viya Platform Operations* for more information.
- SAS recommends that you perform backups during off-peak hours. Generally, backups are not disruptive to the environment, but problems can still occur. Here are some examples:
 - It is possible but unlikely that content might not be from a single point in time since different content is backed up with different methodologies.
 - The content might not be as expected based on other ongoing activities.
- For information about using PostgreSQL database options to improve the performance of the backup, see [“Tuning PostgreSQL Database Backup Operations”](#) in *SAS Viya Platform Administration: Tuning*.
- Make sure that the source Kubernetes environment is functioning as expected, and that the frequency of performing [backups on page 5](#) is as specified in the [recovery point objective \(RPO\)](#) set by your organization.
- Perform a [scan on page 6](#) of your source Kubernetes environment before each backup.
- Make sure that the scheduled backups are successful.
- After each successful backup, see [“Determine the Backup ID of the Last Successful Backup” on page 38](#) to obtain and save the backup ID.
- Use the backup job to perform a backup of SAS Viya platform content, configuration, and services. It backs up this data at the same point in time, which is required for a same point-in-time restore.
- If the `$deploy` directory is not stored within a version control system, be sure to back up the `$deploy` directory in addition to performing regular backups.
- Before any infrastructure change, such as a Kubernetes upgrade, perform an ad hoc backup immediately. Do not wait for the next scheduled backup.
- For information about how to change configuration settings for backups, see one of the following files:

File Format	Path
Markdown	<code>\$deploy/sas-bases/examples/ backup/configure/README.md</code>
HTML	<code>\$deploy/sas-bases/docs/ configuration_settings_for_sas_vi ya_backup.htm</code>

- After a configuration change, perform an ad hoc backup immediately. Do not wait for the next scheduled backup.

- After a migration, perform an ad hoc backup immediately. Do not wait for the next scheduled backup to run. For a future restore, use this backup or any backup performed after this backup. Do not use a backup performed before the migration.
- Edit the backup configuration properties immediately after a new installation or any modifications to your SAS Viya platform deployment. Examples include, but are not limited to, the following:
 - deploying the SAS Viya platform
 - installing software updates
 - changing the topology
 - modifying the SAS Viya platform configuration
 - changing configuration properties
- Perform an ad hoc backup immediately before and after a tenant is onboarded or offboarded. Do not wait for the next scheduled backup to run.
- Backups are purged after the retention period (default is 30 days). If you do not want any backups to be deleted after the retention period, you must manually archive the backups before they are purged.
- If a Kubernetes job entry for a purged backup needs to be retained, set the property `RETAIN_BACKUP_JOB_HISTORY_AFTER_PURGE` to `TRUE`. If this property is not set to true, all Kubernetes job entries are deleted by default.
- If the default storage class policy is not set to `retain`, the `sas-common-backup-data-storage-class-transformer.yaml` patch transformer must be used during deployment. As a result, backup PVs are created with a storage class that has the retain reclaim policy set. This ensures that the associated PVC is retained if the namespace is deleted.
- If the environment being backed up or restored was deployed by the SAS Viya Platform Deployment Operator, then note the following:
 - The SAS Viya Platform Deployment Operator does not participate in a backup and restore.
 - An ad hoc backup or restore should be performed manually without changing the custom resource (CR) file used by the SAS Viya Platform Deployment Operator.

See Also

For more information, see [“Backup Details” on page 31](#).

Perform an Ad Hoc Backup

If you are in a multi-tenant environment, see one of the following files for configuration options:

File Format	Path
Markdown	<code>\$deploy/sas-bases/examples/backup/configure/README.md</code>
HTML	<code>\$deploy/sas-bases/docs/configuration_settings_for_sas_viya_backup.htm</code>

To trigger an ad hoc backup from the default backup schedule:

```
kubect1 create job -n name-of-namespace --from=cronjob/sas-scheduled-backup-job backup-job-name
```

Useful Commands

As you perform and manage backups, the following tasks and associated commands are helpful.

List All Backups

To list all backups:

```
kubect1 get jobs -n name-of-namespace -l "sas.com/backup-job-type=scheduled-backup" -L
"sas.com/sas-backup-id,sas.com/sas-backup-job-status,sas.com/sas-backup-persistence-status"
```

Get Status of a Backup

To obtain the status of a backup, get the ID of the applicable backup :

```
kubect1 get jobs -n name-of-namespace -l "sas.com/backup-job-type=scheduled-backup" -L
"sas.com/sas-backup-id"
```

Then to get the status for the backup job, specify the backup ID from the previous command:

```
kubect1 get jobs -n name-of-namespace -l "sas.com/sas-backup-id=backupID" -L "sas.com/sas-
backup-id,sas.com/backup-job-type,sas.com/sas-backup-job-status,sas.com/sas-backup-persistence-
status"
```

For example, suppose the backup ID is 2020-04-01T10_22_33_633_0700:

```
kubect1 get jobs -n viyans -l "sas.com/sas-backup-id=2020-04-01T10_22_33_633_0700" -L "sas.com/
sas-backup-id,sas.com/backup-job-type,sas.com/sas-backup-job-status,sas.com/sas-backup-
persistence-status"
```

Repeat until you get a status of Completed, Failed, or CompletedWithWarning.

IMPORTANT If a data source that contains a volume to be backed up is not running, the Failed status is quickly returned.

Get Details of a Backup

To obtain details for a backup, get the ID of the applicable backup:

```
kubect1 get jobs -n name-of-namespace -l "sas.com/backup-job-type=scheduled-backup" -L
"sas.com/sas-backup-id"
```

Then to get the details for the backup job, specify the backup ID from the previous command:

```
kubect1 describe jobs -n name-of-namespace -l "sas.com/sas-backup-id=backupID"
```

For example, suppose the backup ID is 2020-04-01T10_22_33_633_0700:

```
kubect1 describe jobs -n viyans -l "sas.com/sas-backup-id=2020-04-01T10_22_33_633_0700"
```

To get the details for the backup data sources in a given job, specify the backup ID from the previous command:

```
kubectl get jobs -n name-of-namespace -l "sas.com/sas-backup-id=backupID", "sas.com/backup-job-type=scheduled-backup" -L "sas.com/sas-backup-datasource-types"
```

View Backup Logs

To view the log of a specific backup job, find its pod name:

```
kubectl get pods -n name-of-namespace | grep backup-job-name
```

To see the log:

```
kubectl logs pod-name -n name-of-namespace -c sas-backup-job
```

Backup and Restore: Schedule a Backup

Overview

In the SAS Viya platform, a default backup is scheduled on Sunday at 1:00 a.m. out of the box. You can [change the default scheduled backup](#), or you can [add additional scheduled backups](#) as needed.

Note: The time zone of scheduled jobs in the SAS Viya platform is determined by the time zone of the Kubernetes controller manager. See the [Kubernetes Cronjob documentation](#) for more information.

Here are the other backup tasks:

- [list scheduled backups](#)
- [unschedule a backup schedule](#)
- [change a backup schedule.](#)

Change the Default Backup Schedule

Note: This functionality is available in the 2021.2.5 and later releases of the SAS Viya platform.

See one of the following files for information about changing the default backup schedule:

File Format	Path
Markdown	<code>\$deploy/sas-bases/examples/backup/configure/README.md</code>

To schedule additional backups, see [“Schedule a Backup” on page 12](#).

Schedule a Backup

Note: To perform some elements of this task, you must have elevated Kubernetes permissions.

You can schedule a backup through a CronJob. By default, a backup is scheduled weekly on Sunday at 1:00 a.m. A maximum of 100 backup jobs per day can be scheduled. The CronJob has a backup manifest and creates an instance of a backup using the KUBECTL CREATE command.

To create a custom scheduled backup:

- 1 If it does not exist, create the directory `$deploy/site-config/backup`.¹
See [“Retrieve Required Files” in SAS Viya Platform: Deployment Guide](#) for more information about the `$deploy` directory.
- 2 Navigate to the `$deploy/sas-bases/examples/backup` folder.
- 3 Copy the `$deploy/sas-bases/examples/backup` directory to the `$deploy/site-config/backup` directory.
- 4 Navigate to the `$deploy/site-config/backup` directory.
- 5 Make sure that the Execute permission is set on the `generate-backup-job` script.

```
chmod +x generate-backup-job.sh
```

Make sure that Write permission is set on the `kustomization.yaml` file.

```
chmod +w kustomization.yaml
```

- 6 Execute the `generate-backup-job.sh` script to generate the manifest file for the scheduled backup job.

```
./generate-backup-job.sh scheduled "cron-expr"
```

For example:

```
./generate-backup-job.sh scheduled "0 1 * * 6"
```

This example creates a schedule that runs the job weekly on Saturday at 1:00 a.m. This is a Kubernetes crontab expression that schedules the backup job at whatever time the user specifies.

This script generates a file with the prefix `sas-scheduled-backup-`, followed by a unique identifier, in the same folder.

- 7 Edit the `kustomization.yaml` file. Add the name of the manifest file generated in step 6 to the **resources** section. For example:

```
resources: - sas-scheduled-backup-xxx.yaml
```

There should be only one scheduled backup manifest file in the **resources** section.

1. During deployment, you created a deployment directory that SAS recommended that you name `deploy` or an alternative name that is meaningful to you. The directory is referred to as `$deploy` in this guide. Replace `$deploy` with the directory name that you choose.

8 Navigate to the `$deploy` directory.

- a** Make a copy of the `kustomization.yaml` file to recover after temporary changes are made.

```
cp $deploy/kustomization.yaml $deploy/kustomization.yaml.save
```

- b** Edit the `kustomization.yaml` and add the following to the `resources` section.

```
resources: - site-config/backup
```

9 Build the manifest for the scheduled backup job.

```
kustomize build -o site.yaml
```

10 Apply the manifest.

```
kubectl apply -f site.yaml --server-side --force-conflicts
```

To change the default backup schedule, see [“Change the Default Backup Schedule” on page 11](#).

List Scheduled Backups

To list scheduled backups:

```
kubectl get cronjobs -n name-of-namespace -l "sas.com/backup-job-type=scheduled-backup"
```

Unschedule a Backup Schedule

To permanently unschedule a backup schedule, delete the CronJob.

```
kubectl delete cronjob -n name-of-namespace name-of-cron-job
```

To temporarily unschedule or suspend a backup schedule:

1 Navigate to the `$deploy/site-config/backup` directory.**2** Edit the YAML file for the scheduled job that you want to unschedule. Update the value of the `SUSPEND` field to `true`.**3** Save the file.**4** Edit the `kustomization.yaml` file, and add the name of the YAML file from step 2 to the `resources` section:

```
resources: - sas-scheduled-job-xxxx.yaml
```

5 Navigate to the `$deploy` directory. Edit the `kustomization.yaml` file and add the following line to the `resources` section:

```
resources: - site-config/backup
```

6 Build the manifest:

```
kustomize build -o site.yaml
```

7 Apply the manifest:

```
kubectl apply -f site.yaml --server-side --force-conflicts
```

Note: See “[Managing a SAS Viya Platform Deployment](#)” in *SAS Viya Platform: General Management of Servers and Services* for more information about suspending and resuming jobs.

Change a Backup Schedule

To change a backup schedule:

- 1 Navigate to the `$deploy/site-config/backup` directory.
- 2 Edit the YAML file for the scheduled job that you want to change. Update the value of the `SCHEDULE` field to the cron expression representing the new schedule.
- 3 Save the file.
- 4 Edit the `kustomization.yaml` file, and add the name of the YAML file from step 2 to the resources section:

```
resources: - sas-scheduled-job-xxxx.yaml
```

- 5 Navigate to the `$deploy` directory. Verify that `- site-config/backup` exists in the resources section of the `kustomization.yaml` file. If it does not, add the following line to the resources section:

```
resources: - site-config/backup
```

- 6 Build the manifest:

```
kustomize build -o site.yaml
```

- 7 Apply the manifest:

```
kubectl apply -f site.yaml --server-side --force-conflicts
```

Backup and Restore: Perform a Restore

Note: If you are in a multi-tenant environment, see “[Multi-Tenancy Support](#)” on page 4 for information about multi-tenancy.

What Restore Scenarios Are Not Supported?

- Restoring a backup from a different cadence is not supported. For example, you can restore to a SAS Viya platform Stable environment from a backup of a SAS Viya platform Stable environment. You can restore to a SAS Viya platform Long-Term Support environment from a backup of a SAS Viya platform Long-Term Support environment. But you cannot restore to a SAS Viya platform Long-Term Support environment from a backup of a SAS Viya platform Stable environment. See

“Software Releases and Versions” in *Getting Started with SAS Viya Platform Operations* for more information.

- Restoring a backup from a different cadence version is not supported. For example, you can restore to a SAS Viya platform 2021.2.3 environment from a backup of a 2021.2.3 environment. But you cannot restore to a 2021.2.3 environment from a backup of a 2021.2.2 or earlier environment. For example, you cannot restore from a 2021.2.2 environment to a 2021.2.3 or newer environment. See “Software Releases and Versions” in *Getting Started with SAS Viya Platform Operations* for more information.
- See “Multi-Tenancy Support ” on page 4 for information about support for backup and restore of environments that are configured for multi-tenancy.

Best Practices for Performing Restores

- To perform some elements of this task, you must have Kubernetes administrative permissions on the cluster.
- Restoring a backup from a different cadence is not supported. For example, you can restore to a SAS Viya platform Stable environment from a backup of a SAS Viya platform Stable environment. You can restore to a SAS Viya platform Long-Term Support environment from a backup of a SAS Viya platform Long-Term Support environment. But you cannot restore to a SAS Viya platform Long-Term Support environment from a backup of a SAS Viya platform Stable environment. See “Software Releases and Versions” in *Getting Started with SAS Viya Platform Operations* for more information.
- Restoring a backup from a different cadence version is not supported. For example, you can restore to a SAS Viya platform 2021.2.3 environment from a backup of a 2021.2.3 environment. But you cannot restore to a 2021.2.3 environment from a backup of a 2021.2.2 or earlier environment. For example, you cannot restore from a 2021.2.2 environment to a 2021.2.3 or newer environment. See “Software Releases and Versions” in *Getting Started with SAS Viya Platform Operations* for more information.
- While performing a restore, ensure that the system is not being actively used.
- Always choose the most recent and successful backup to perform a restore.
- See “Sizing Recommendations for Restore” for information about PostgreSQL database size in the restore space.
- For information about using PostgreSQL database options to improve the performance of the restore, see “Tuning PostgreSQL Database Restore Operations” in *SAS Viya Platform Administration: Tuning*.
- For less common configuration changes to restores, such as providing a custom database name or enabling parallel execution, see one of the following files:

File Format	Path
Markdown	<code>\$deploy/sas-bases/examples/ restore/postgresql/README.md</code>
HTML	<code>\$deploy/sas-bases/docs/ uncommon_restore_customizations.h tm</code>

- Restoring to a different cloud provider is supported as long as the backup SAS Viya 4 environment and the target SAS Viya 4 environment use an internal instance of PostgreSQL.
- See [“Support for Migration from SAS Viya 4 to SAS Viya 4 for Offerings”](#) in *SAS Viya Platform: Full-System Migration* for links to additional steps that might be required for some offerings.
- If the environment being backed up or restored was deployed by the SAS Viya Platform Deployment Operator, then note the following:
 - The SAS Viya Platform Deployment Operator does not participate in a backup and restore.
 - An ad hoc backup or restore should be performed manually without changing the custom resource (CR) file used by the SAS Viya Platform Deployment Operator.

Perform a Restore

Note:

- If you are in a multi-tenant environment, you must onboard every tenant that you want to restore before performing the restore. See [“Onboard Tenants”](#) in *SAS Viya Platform: Multi-tenancy* for more information.
- Check for active or scheduled jobs in the environment before performing a restore.
 - Suspend any scheduled jobs in the environment to prevent them from automatically starting after the restore. To check for and disable scheduled jobs, you can do one of the following:

Use the Jobs page in SAS Environment Manager.	See “Job Flow Scheduling Service: Overview” in <i>SAS Viya Platform: Jobs and Flows</i> .
--	---

Use the job plug-in to the SAS Viya platform CLI.	See “Job Flow Scheduling Service: How To (CLI)” in <i>SAS Viya Platform: Jobs and Flows</i> .
---	---

- If your deployment contains SAS Workload Orchestrator, ensure that there are no running or pending SAS Workload Orchestrator jobs in the environment before performing a restore. To check the status of these jobs, you can do one of the following:

Use the SAS Workload Orchestrator graphical user interface (GUI) in SAS Environment Manager.	See “SAS Workload Orchestrator Overview” in <i>SAS Viya Platform: Workload Management</i> .
--	---

Use the workload_orchestrator plug-in to the SAS Viya platform CLI.	Sign in to the sas-viya command-line interface (CLI) and download the workload-orchestrator plug-in. See “Command-Line Interface: Preliminary Instructions” in <i>SAS Viya Platform: Using the Command-Line Interface</i> for more information.
---	---

Use this command to identify active workload-orchestrator jobs:

```
./sas-viya
workload-orchestrator jobs list
```

Do not proceed with the restore if there are any running or pending SAS Workload Orchestrator jobs.

Here are the steps to perform a restore:

- 1 Navigate to the \$deploy folder.¹

See [“Retrieve Required Files” in SAS Viya Platform: Deployment Guide](#) for more information about the \$deploy directory.

- 2 Make a copy of the kustomization.yaml file to recover after temporary changes are made:

```
cp kustomization.yaml kustomization.yaml.save
```

- 3 Add the sas-restore-job-parameters code below to the configMapGenerator section of kustomization.yaml, and remove the configMapGenerator line, if it is already present in the default kustomization.yaml:

```
configMapGenerator:
- name: sas-restore-job-parameters
  behavior: merge
  literals:
  - SAS_BACKUP_ID={{ SAS-BACKUP-ID-VALUE }}
  - SAS_DEPLOYMENT_START_MODE=RESTORE
```

Here are more details about the previous code.

- Replace the value for SAS-BACKUP-ID-VALUE (including the braces) with the ID of the backup that is selected for restore. See [“List All Backups” on page 10](#) for information about how to locate the backup ID.
- To increase the logging levels, add the following line to the `literals` section:


```
- SAS_LOG_LEVEL=DEBUG
```

IMPORTANT Replace each variable that is enclosed in braces (`{{ }}`) and spaces with the value that is appropriate for your deployment. For example, `{{ SAS-BACKUP-ID-VALUE }}` should be replaced with a backup ID, such as `2020-04-01T10_22_33_633_0700`.

If you want to exclude a deprecated configuration from being restored to the target SAS Viya 4 system, see one of the following files:

File Format	Path
Markdown	<code>\$deploy/sas-bases/examples/restore/configure/README.md</code>

1. During deployment, you created a deployment directory that SAS recommended that you name `deploy` or an alternative name that is meaningful to you. The directory is referred to as `$deploy` in this guide. Replace `$deploy` with the directory name that you choose.

HTML	<code>\$deploy/sas-bases/docs/ configuration_settings_for_sas_v ya_restore.htm</code>
------	---

- 4 Generate a temporary manifest to complete the restore. This manifest is applied in a later step.

```
kustomize build -o site-restore.yaml
```

- 5 Modify kustomization.yaml again for the initial phase of restore.

Add the following line to the **resources** section:

```
- sas-bases/overlays/restore
```

- 6 Perform the following steps to start the restore job:

- a Check if the restore job already exists.

```
kubectl get job sas-restore-job -n name-of-namespace
```

If it does exist, delete the job.

```
kubectl delete job sas-restore-job -n name-of-namespace
```

- b From the `$deploy` directory, generate a second temporary manifest for the initial phase of restore, and apply it:

```
kustomize build -o site-restore-initial.yaml
```

```
kubectl apply -f site-restore-initial.yaml --server-side --force-conflicts
```

This starts the restore job.

Note: This causes a surge of pods in the cluster. Therefore, make sure that you have sufficient available resources. In other words, you might increase the max nodes in your autoscaling configuration, if required. See [“Tuning Workload Allocations” in SAS Viya Platform Administration: Tuning](#) for more information.

- 7 The restore job was started in the previous step. Inspect the status of the job:

```
kubectl get jobs -n name-of-namespace -l "sas.com/backup-job-type=restore" -L "sas.com/sas-backup-id,sas.com/backup-job-type,sas.com/sas-restore-status"
```

Submit the command until you get the Completed status. Possible statuses are Completed, CompletedWithError, and Failed.

- If you get an Error status or need to rerun the job for any reason, delete the job:

```
kubectl delete job sas-restore-job -n name-of-namespace
```

And then perform this previous step again:

```
kubectl apply -f site-restore-initial.yaml --server-side --force-conflicts
```

- If a resource is not available to the pod, the restore job might appear to be running with "0/1" completions, whereas it actually is in a waiting state. You might want to describe the restore pod to confirm that the job has been able to start.

Get the name of the restore pod:

```
kubectl get pods -n name-of-namespace | grep sas-restore-job
```

Describe the pod, substituting the restore pod name that you identified in the last step:

```
kubectl -n name-of-namespace describe pod sas-restore-job-pod-name
```

- 8** Check the log of the restore job to confirm that the restore was successful.

To view the log of the restore job, find its pod name:

```
kubectl get pods -n name-of-namespace | grep sas-restore-job
```

To see the log:

```
kubectl -n name-of-namespace logs sas-restore-job-pod-name -c sas-restore-job
```

- 9** While the restore job is running, all pods get restarted with the exception of the CAS controller pods and CAS worker pods, which are terminated. Before proceeding to the next step, do the following:

- Make sure that the pods that are not dependent on the CAS server are running again after the restart.

Note: At this point, while CAS-dependent pods might be in a **Running** state, requests to those pods might fail until CAS is successfully restarted in a later step.

- Make sure that the CAS controller pods and CAS worker pods have been removed.

```
kubectl get pods -n name-of-namespace
```

- 10** The file system restore requires a clean volume. Otherwise, the restore for the volume is skipped.

Get the list of persistent volume claims for CAS that are used by the file system restore .

```
kubectl -n name-of-namespace get pvc -l 'sas.com/backup-role=provider,app.kubernetes.io/part-of=cas'
```

If you are using dynamic provisioning and restoring all the tenants, clean up the volumes with this command:

```
kubectl -n name-of-namespace delete pvc -l 'sas.com/backup-role=provider,app.kubernetes.io/part-of=cas'
```

If you are using static provisioning, take the necessary steps to manually clean up the volumes participating in the restore.

- 11** In this step, the services are restarted, and the file system restore for CAS begins. Apply the first temporary manifest to complete the restore:

```
kubectl apply -f site-restore.yaml --server-side --force-conflicts
```

- 12** Confirm that the deployment is running. Verify that all pods are running successfully:

```
kubectl get pods -n name-of-namespace
```

Make sure that you are able to sign on to the system.

- 13** Check the CAS pod logs of each participating tenant for these messages to confirm that the restore was successful:

- "message": "Restore of CAS permstore volume content completed successfully."
- "message": "Restore of CAS data volume content completed successfully."
- "message": "Restore of CAS data and permstore volumes completed successfully."

If all of these messages are present in the log, then the CAS restore was successful. If neither or only one of these messages are present in the log along with an error message, then the CAS restore was not successful. Check the CAS pod logs for details.

Here is a sample command to check the CAS pod logs:

```
kubectl -n name-of-namespace logs sas-cas-server-default-controller -c sas-cas-server
```

14 Remove the temporary manifests:

```
rm site-restore-initial.yaml site-restore.yaml
```

15 Recover the saved kustomization.yaml copy to remove any restore modifications:

```
mv kustomization.yaml.save kustomization.yaml
```

The deployment is now ready for use.

Post-Restore Validation

SAS Viya 4 provides the SAS Viya 4 Inventory and SAS Viya 4 Comparison reports that you can use to validate the target SAS Viya 4 environments. These reports, and the associated SASVIYATYPES table are accessible by SAS Environment Manager. To use the reports, you must first generate additional required data by performing an inventory scan.

Prior to the 2022.1.4 release of the SAS Viya platform, the inventory scan was run by creating a Kubernetes job by applying a manifest file. This method is not supported in a multi-tenancy environment. As of the 2022.1.4 release of the SAS Viya platform, there is an additional method of running an inventory scan by creating a Kubernetes job from the CronJob. This method does support the scanning of a multi-tenancy environment.

Note: The inventory scan CronJob does not yet generate the extra data necessary to populate the comparison report.

Inventory Scan	Documentation
Run the inventory scan by creating a Kubernetes job by applying a manifest file.	See “Run the Inventory Scan by Creating a Kubernetes Job by Applying a Manifest File” in <i>SAS Viya Platform: Full-System Migration from SAS Viya 3.x</i>. Note: Running an inventory scan by creating a Kubernetes job by applying a manifest file is not currently supported if you are using SCIM for identity management.
Run the inventory scan by creating a Kubernetes job from the CronJob.	See “Perform an Inventory Scan” on page 39.

List Restores

To list restores:

```
kubectl get jobs -n name-of-namespace -l "sas.com/backup-job-type=restore"
```

Show Details of a Restore

To show details for a restore:

```
kubectl -n name-of-namespace describe jobs sas-restore-job
```

View Restore Logs

To view the log of a specific restore job, find its pod name:

```
kubectl get pods -n name-of-namespace | grep sas-restore-job
```

To see the log:

```
kubectl -n name-of-namespace logs sas-restore-job-pod-name -c sas-restore-job
```

Backup and Restore: Purge Backups

Purge Backups

Note: To perform some elements of this task, you must have elevated Kubernetes permissions.

Backups are purged through a CronJob. The CronJob finds backups past the retention period (the default is 30 days) and purges them from all backup data sources. Here is the workflow:

- 1 Purging is performed through a CronJob that executes daily at 12:15 a.m. To get details about the CronJob, submit the following command:

```
kubectl describe cronjob sas-backup-purge-job
```

- 2 The purge operation lists any Kubernetes resource of type JOBS that has a label related to backup. For example:

```
sas.com/sas-backup-id=2020-05-05T07_34_12_612_0700
```

```
sas.com/sas-backup-persistence-status=available
```

```
sas.com/sas-backup-job-status=completed
```

- 3 Purging a backup job is based on the following conditions:

- The backup job is past the retention period.
- The backup job is not already purged.

IMPORTANT During a purge, all successful backups are considered. The most recent successful backup is set aside. This backup is not purged, even if it is past the retention period.

- 4 If there are backup jobs that need to be purged, then an event is sent to the backup agent.
- 5 The backup agent deletes the backup directory from the PV.
- 6 If a successful status is received from the backup agent, the backup job is deleted.

By default, when a backup is purged, the job that created the backup is also purged. If you want to save the backup job when the backup is purged, set the property `RETAIN_BACKUP_JOB_HISTORY_AFTER_PURGE` to `TRUE` as follows:

- 1 Navigate to the `$deploy` directory.
- 2 Edit the `kustomization.yaml` file, and add `RETAIN_BACKUP_JOB_HISTORY_AFTER_PURGE=TRUE` in the `literals` section as follows:

```
configMapGenerator:
- name: sas-backup-job-parameters
  behavior: merge
  literals:
  - RETAIN_BACKUP_JOB_HISTORY_AFTER_PURGE=TRUE
```

- 3 Build the manifest to save the job for the purged backup.

```
kustomize build -o site.yaml
```

- 4 Apply the manifest.

```
kubectl apply -f site.yaml --server-side --force-conflicts
```

If the `RETAIN_BACKUP_JOB_HISTORY_AFTER_PURGE` property is `TRUE`, then the status of the jobs for which the backups are purged is set to `purged`. To view the jobs with a status of `purged`, issue this command:

```
kubectl -n name-of-namespace get jobs -l "sas.com/sas-backup-persistence-status=purged" -L "sas.com/sas-backup-persistence-status"
```

Backup and Restore: Recover from a Disaster

Note: To perform some elements of this task, you must have elevated Kubernetes permissions.

Overview

Disaster recovery in SAS Viya 4 is the process for recovering in the event of a disaster. Recovery in this context refers to regaining access to data, hardware, and software.

Here are the steps.

- Prepare the target environment by installing SAS Viya 4 using the deployment files from the source environment.
- Make sure that the persistent volumes (PVs) that contain backup data of the source environment are accessible from the target environment.
- Perform a restore using the most recent successful backup of the source environment.

The steps above restore to a SAS Viya 4 target environment that is the same as the source environment with respect to the following items:

- the name of the namespace
- the cloud infrastructure
- the set of SAS Viya 4 products and the version

Disaster Recovery Prerequisites and Planning on the Source Environment

- Develop a plan for disaster recovery specific for your organization. This disaster recovery plan should contain a documented process for preparing your organization to recover in the shortest amount of time in the event of a disaster.
- Make sure that the \$deploy directory is backed up initially and after every configuration change on the source system.

Note: During deployment, you created a deployment directory that SAS recommended that you name `deploy` or an alternative name that is meaningful to you. The directory is referred to as `$deploy` in this guide. Replace `$deploy` with the directory name that you choose.

- Make sure to set the Kubernetes reclaim policy for PVs to Retain instead of Delete, in order to prevent accidentally losing backups stored in backup PVs if the namespace is deleted.
- Restoring a backup from a different cadence is not supported. For example, you can restore to a SAS Viya platform Stable environment from a backup of a SAS Viya platform Stable environment. You can restore to a SAS Viya platform Long-Term Support environment from a backup of a SAS Viya platform Long-Term Support environment. But you cannot restore to a SAS Viya platform Long-Term Support environment from a backup of a SAS Viya platform Stable environment. See [“Software Releases and Versions” in *Getting Started with SAS Viya Platform Operations*](#) for more information.
- Restoring a backup from a different cadence version is not supported. For example, you can restore to a SAS Viya platform 2021.2.3 environment from a backup of a 2021.2.3 environment. But you cannot restore to a 2021.2.3 environment from a backup of a 2021.2.2 or earlier environment. For example, you cannot restore from a 2021.2.2 environment to a 2021.2.3 or newer environment. See [“Software Releases and Versions” in *Getting Started with SAS Viya Platform Operations*](#) for more information.
- For information about using PostgreSQL database options to improve the performance of the restore, see [“Tuning PostgreSQL Database Restore Operations” in *SAS Viya Platform Administration: Tuning*](#).
- In order to maximize your ability to recover from a disaster, you should do the following on an ongoing basis:

- ❑ Make sure that the source Kubernetes environment is functioning as expected, and that the frequency of performing [backups](#) is as specified in the [recovery point objective \(RPO\)](#) set by your organization. Make sure that the scheduled backups are successful.
- ❑ Perform a [scan](#) of the source Kubernetes environment before each backup.
- ❑ Make sure that the data is backed up correctly by the backup job, and that the status of the most recent backup is `Completed`.
- ❑ Make sure that the persistent volumes (PVs) that contain backup data are secure by verifying that the necessary replication is configured.
- ❑ Make sure that the backups are stored safely.
- ❑ Make sure that you know the physical location of the PVs that contain backup data. See [“How to Find the Physical Location of the Persistent Volume \(PV\) in an Azure Environment” on page 33](#) for examples of finding the physical location of PVs. Make a note of the PV location information and save it in a file. It is recommended that you save this file in the same location where the `$deploy` directory is stored.
- ❑ Practice the restore process using a successful backup of the source system.

Multi-Tenancy Support

Disaster recovery of a SAS Viya 4 environment to a SAS Viya 4 environment is supported between environments that are configured for multi-tenancy. In SAS Viya 4 version LTS 2021.2, the supported scenario is the recovery of a full SAS Viya 4 multi-tenant environment (provider and all customer tenants are backed up simultaneously) to a SAS Viya 4 multi-tenant environment where the target environment is a matching environment, meaning that the target environment:

- contains all of the same tenants that are in the backup after being restored. The tenants must be onboarded in the target environment before the backup is restored. The provider content is always restored.

Note:

- Do not restore to a target environment that has tenants that are not in the backup.
 - You must onboard all of the tenants available in the backup before restoring them to the target environment.
 - The tenant names must be the same between the source and target environment.
-
- uses the same cloud provider as in the backup.
 - has a matching product set as in the backup. The restore might fail if the set of products in the backup does not match the set of products available in the target environment.
 - uses the same database option (`schemaPerApplicationTenant` or `databasePerTenant`) as in the backup.

Perform a Disaster Recovery

- 1 See [“Determine the Backup ID of the Last Successful Backup” on page 38](#) to get the backup ID.

2 Prepare the target environment.

Note: In the following steps, perform a deployment using Kubernetes commands of SAS Viya 4.

a See [“Support for Migration from SAS Viya 4 to SAS Viya 4 for Offerings” in SAS Viya Platform: Full-System Migration](#) for links to additional steps that might be required for some offerings.

b A virtual infrastructure and certain hardware and resources are required for the deployment. Review [“Virtual Infrastructure Requirements” in System Requirements for the SAS Viya Platform](#) and [“Storage Requirements” in System Requirements for the SAS Viya Platform](#) to ensure that all requirements are met.

For example, one requirement is that you must create a dedicated namespace for the deployment. Create the namespace that is mentioned in the root kustomization.yaml file in the \$deploy directory.

```
kubectl create -n name-of-namespace
```

c See [“Sizing Recommendations for Restore”](#) for information about the PostgreSQL database size in the restore space.

d Copy the \$deploy directory saved from the source environment to the \$deploy directory on the target environment.

e Perform any necessary [pre-installation tasks](#).

f If you need to make any changes to your manifest, see [“Common Customizations” in SAS Viya Platform: Deployment Guide](#) and the README files that this section describes.

g Deploy the software by following the steps [here](#).

Note: If the backup that you are using for the source SAS Viya 4 environment was created by a migration or has undergone a backup or restore, make sure that the kustomization.yaml file does not contain any references to migration or backup jobs from the source environment

h Perform any necessary post-installation steps. See [“Post-Installation Tasks” in SAS Viya Platform: Deployment Guide](#) for more information.

i After completing the installation of SAS Viya 4, ensure that the software is deployed correctly and works as expected by logging in to the SAS Logon application.

j Run an [ad hoc backup on page 7](#) to ensure that deployment is working correctly. This backup can be helpful if you want to roll back to the original state for the target environment prior to restore.

3 Ensure that the backup contents are accessible from the target environment.

Note:

Multiple persistent volumes (PVs) are used for backup.

Run this command to list the PVCs where backup is stored:

```
kubectl -n namespace get pvc -l "sas.com/backup-role=storage"
```

■ If you are reusing the PVs from the source environment (static provisioning), then verify that the backup contents are already available.

- If you are not reusing the PVs from the source environment (dynamic provisioning), then you need to copy the backup content from the source environment PV to the respective PV on the target environment. Make sure to preserve the permissions and date timestamp while copying.

Copy the backup data from the source environment to the target environment. See [“Examples of Transferring Backup Data between the SAS Viya Platform Environment and the Local Filesystem”](#) for a generic example for copying data. See [“Copy Backup Data from Azure Environment to Azure Environment” on page 37](#) for examples of copying backup PV data from one Azure environment to another Azure environment.

- 4 Perform a restore of the last successful backup using the backup ID identified in the planning step. Use the same steps as if you are doing a normal restore.

Note: If you are in a multi-tenant environment, you must onboard every tenant that you want to restore before performing the restore. See [“Onboard Tenants” in SAS Viya Platform: Multi-tenancy](#) for more information.

- a Navigate to the \$deploy folder.¹

See [“Retrieve Required Files” in SAS Viya Platform: Deployment Guide](#) for more information about the \$deploy directory.

- b Make a copy of the kustomization.yaml file to recover after temporary changes are made:

```
cp kustomization.yaml kustomization.yaml.save
```

- c Add the sas-restore-job-parameters code below to the configMapGenerator section of kustomization.yaml, and remove the configMapGenerator line, if it is already present in the default kustomization.yaml:

```
configMapGenerator:
- name: sas-restore-job-parameters
  behavior: merge
  literals:
  - SAS_BACKUP_ID={{ SAS-BACKUP-ID-VALUE }}
  - SAS_DEPLOYMENT_START_MODE=RESTORE
```

Here are more details about the previous code.

- Replace the value for SAS-BACKUP-ID-VALUE (including the braces) with the ID of the backup that is selected for restore. See [“List All Backups” on page 10](#) for information about how to locate the backup ID.
- To increase the logging levels, add the following line to the `literals` section:

```
- SAS_LOG_LEVEL=DEBUG
```

IMPORTANT Replace each variable that is enclosed in braces ({ }) and spaces with the value that is appropriate for your deployment. For example, {{ SAS-BACKUP-ID-VALUE }} should be replaced with a backup ID, such as 2020-04-01T10_22_33_633_0700.

If you want to exclude a deprecated configuration from being restored to the target SAS Viya 4 system, see one of the following files:

1. During deployment, you created a deployment directory that SAS recommended that you name deploy or an alternative name that is meaningful to you. The directory is referred to as \$deploy in this guide. Replace \$deploy with the directory name that you choose.

File Format	Path
Markdown	<code>\$deploy/sas-bases/examples/restore/configure/README.md</code>
HTML	<code>\$deploy/sas-bases/docs/configuration_settings_for_sas_viya_restore.htm</code>

- d Generate a temporary manifest to complete the restore. This manifest is applied in a later step.

```
kustomize build -o site-restore.yaml
```

- e Modify `kustomization.yaml` again for the initial phase of restore.

Add the following line to the `resources` section:

```
- sas-bases/overlays/restore
```

- f Perform the following steps to start the restore job:

- 1 Check if the restore job already exists.

```
kubectl get job sas-restore-job -n name-of-namespace
```

If it does exist, delete the job.

```
kubectl delete job sas-restore-job -n name-of-namespace
```

- 2 From the `$deploy` directory, generate a second temporary manifest for the initial phase of restore, and apply it:

```
kustomize build -o site-restore-initial.yaml
```

```
kubectl apply -f site-restore-initial.yaml --server-side --force-conflicts
```

This starts the restore job.

Note: This causes a surge of pods in the cluster. Therefore, make sure that you have sufficient available resources. In other words, you might increase the max nodes in your autoscaling configuration, if required. See [“Tuning Workload Allocations” in SAS Viya Platform Administration: Tuning](#) for more information.

- g The restore job was started in the previous step. Inspect the status of the job:

```
kubectl get jobs -n name-of-namespace -l "sas.com/backup-job-type=restore" -L "sas.com/sas-backup-id,sas.com/backup-job-type,sas.com/sas-restore-status"
```

Submit the command until you get the Completed status. Possible statuses are Completed, CompletedWithError, and Failed.

- If you get an Error status or need to rerun the job for any reason, delete the job:

```
kubectl delete job sas-restore-job -n name-of-namespace
```

And then perform this previous step again:

```
kubectl apply -f site-restore-initial.yaml --server-side --force-conflicts
```

- If a resource is not available to the pod, the restore job might appear to be running with "0/1" completions, whereas it actually is in a waiting state. You might want to describe the restore pod to confirm that the job has been able to start.

Get the name of the restore pod:

```
kubectl get pods -n name-of-namespace | grep sas-restore-job
```

Describe the pod, substituting the restore pod name that you identified in the last step:

```
kubectl -n name-of-namespace describe pod sas-restore-job-pod-name
```

- h Check the log of the restore job to confirm that the restore was successful.

To view the log of the restore job, find its pod name:

```
kubectl get pods -n name-of-namespace | grep sas-restore-job
```

To see the log:

```
kubectl -n name-of-namespace logs sas-restore-job-pod-name -c sas-restore-job
```

- i While the restore job is running, all pods get restarted with the exception of the CAS controller pods and CAS worker pods, which are terminated. Before proceeding to the next step, do the following:

- Make sure that the pods that are not dependent on the CAS server are running again after the restart.

Note: At this point, while CAS-dependent pods might be in a **Running** state, requests to those pods might fail until CAS is successfully restarted in a later step.

- Make sure that the CAS controller pods and CAS worker pods have been removed.

```
kubectl get pods -n name-of-namespace
```

- j The file system restore requires a clean volume. Otherwise, the restore for the volume is skipped.

Get the list of persistent volume claims for CAS that are used by the file system restore .

```
kubectl -n name-of-namespace get pvc -l 'sas.com/backup-role=provider,app.kubernetes.io/part-of=cas'
```

If you are using dynamic provisioning and restoring all the tenants, clean up the volumes with this command:

```
kubectl -n name-of-namespace delete pvc -l 'sas.com/backup-role=provider,app.kubernetes.io/part-of=cas'
```

If you are using static provisioning, take the necessary steps to manually clean up the volumes participating in the restore.

- k In this step, the services are restarted, and the file system restore for CAS begins. Apply the first temporary manifest to complete the restore:

```
kubectl apply -f site-restore.yaml --server-side --force-conflicts
```

- l Confirm that the deployment is running. Verify that all pods are running successfully:

```
kubectl get pods -n name-of-namespace
```

Make sure that you are able to sign on to the system.

- m Check the CAS pod logs of each participating tenant for these messages to confirm that the restore was successful:

- "message": "Restore of CAS permstore volume content completed successfully."
- "message": "Restore of CAS data volume content completed successfully."

- "message": "Restore of CAS data and permstore volumes completed successfully."

If all of these messages are present in the log, then the CAS restore was successful. If neither or only one of these messages are present in the log along with an error message, then the CAS restore was not successful. Check the CAS pod logs for details.

Here is a sample command to check the CAS pod logs:

```
kubectl -n name-of-namespace logs sas-cas-server-default-controller -c sas-cas-server
```

- Remove the temporary manifests:

```
rm site-restore-initial.yaml site-restore.yaml
```

- Recover the saved kustomization.yaml copy to remove any restore modifications:

```
mv kustomization.yaml.save kustomization.yaml
```

The deployment is now ready for use.

Post-Restore Validation

SAS Viya 4 provides the SAS Viya 4 Inventory and SAS Viya 4 Comparison reports that you can use to validate the target SAS Viya 4 environments. These reports, and the associated SASVIYATYPES table are accessible by SAS Environment Manager. To use the reports, you must first generate additional required data by performing an inventory scan.

Prior to the 2022.1.4 release of the SAS Viya platform, the inventory scan was run by creating a Kubernetes job by applying a manifest file. This method is not supported in a multi-tenancy environment. As of the 2022.1.4 release of the SAS Viya platform, there is an additional method of running an inventory scan by creating a Kubernetes job from the CronJob. This method does support the scanning of a multi-tenancy environment.

Note: The inventory scan CronJob does not yet generate the extra data necessary to populate the comparison report.

Inventory Scan	Documentation
Run the inventory scan by creating a Kubernetes job by applying a manifest file.	See “Run the Inventory Scan by Creating a Kubernetes Job by Applying a Manifest File” in <i>SAS Viya Platform: Full-System Migration from SAS Viya 3.x</i>. Note: Running an inventory scan by creating a Kubernetes job by applying a manifest file is not currently supported if you are using SCIM for identity management.
Run the inventory scan by creating a Kubernetes job from the CronJob.	See “Perform an Inventory Scan” on page 39.

Backup and Restore: Administrator Tasks

Here are tasks that an administrator can perform as needed:

- Configure a backup. You can change a backup configuration through ConfigMap. Configuration settings include:
 - backup purge information, such as retention period (default is 30 days)
 - backup job time-out value
 - additional SAS Configuration Server (Consul) properties to back up

For information about how to change the configuration settings for backups, see one of the following files:

File Format	Path
Markdown	<code>\$deploy/sas-bases/examples/ backup/configure/README.md</code>
HTML	<code>\$deploy/sas-bases/docs/ configuration_settings_for_sas_vi ya_backup.htm</code>

- List the persistent volume claims that store the backup content for products that participate in backup and restore. For example:


```
kubectl -n name-of-namespace get pvc -l "sas.com/backup-role=provider"
```
- Determine the persistent volume claim (PVC) that each backup source is stored on. For example:


```
kubectl -n name-of-namespace get pvc -l "sas.com/backup-role=storage"
```
- Determine the persistent volume claim (PVC) that corresponds to the persistent volume that each backup source is stored on. See [“Backup Details” on page 31](#) for more information.

Backup and Restore: Reference

Backup Details

Overview

A backup consists of several files contained in a time-stamped directory and distributed among multiple persistent volumes (PV). The timestamp is the backup ID. Different PVs are used for different backup sources, such as CAS and PostgreSQL. Identical timestamp directories on different PVs indicate that the files belong to the same backup. The following table shows the persistent volume claim (PVC) that corresponds to the PV that each backup source is stored on.

Backup Source	PVC
SAS Infrastructure Data Server or PostgreSQL server	sas-common-backup-data
Configuration	sas-common-backup-data
CAS	sas-cas-backup-data

IMPORTANT The PVs that contain backups are part of the cluster that they are backing up. Therefore, it is recommended that backups are routinely copied to storage outside of the cluster, such as NFS, in the event that the PV or entire cluster fails. For best practices for backups and disaster recovery, see [“Best Practices for Performing Backups” on page 7](#) and [“Disaster Recovery Prerequisites and Planning on the Source Environment” on page 23](#).

How to Determine the Persistent Volume Claim (PVC) That Each Backup Source Is Stored On

There might be additional persistent volumes. To list all backup persistent volume claims corresponding to these persistent volumes, use this command:

```
kubectl -n name-of-namespace get pvc -l "sas.com/backup-role=storage"
```

Here is an example of the output:

NAME	STATUS	VOLUME	CAPACITY
ACCESS MODES	STORAGECLASS	AGE	
sas-cas-backup-data	Bound	pvc-ce04c7d5-0dd6-4dad-99c6-bbc6e0dd00e1	8Gi RWX
nfs-client	34d		
sas-common-backup-data	Bound	pvc-724d74f1-9d79-4639-9c77-399b3c3ad10a	25Gi RWX
nfs-client	34d		

Backup Directory Structure

The backup directory structure differs for a non-multi-tenancy environment and a multi-tenancy environment.

- Here is the backup directory structure for a non-multi-tenancy environment:

backup PVC/backup ID/__default__/source/file name

For example, suppose you backed up the PostgreSQL server in a non-multi-tenancy environment. The backup PVC is sas-common-backup-data. The backup ID is 2019-10-28T05_33_47_326-0400, which is the timestamp. In this scenario, the backup creates the following files:

- sas-common-backup-data/2019-10-28T05_33_47_326-0400/__default__/postgres/blobs
- sas-common-backup-data/2019-10-28T05_33_47_326-0400/__default__/postgres/SharedServices_pg_dump.dmp
- sas-common-backup-data/2019-10-28T05_33_47_326-0400/__default__/postgres/SharedServices_pg_dump.log
- sas-common-backup-data/2019-10-28T05_33_47_326-0400/__default__/postgres/status.json

- Here is the backup directory structure for the provider in a multi-tenancy environment:

backup PVC/backup ID/provider/source/file name

Here is the backup directory structure for a tenant in a multi-tenancy environment:

backup PVC/backup ID/tenant/source/file name

The file names of the backup files created in a multi-tenancy environment are similar to those in a non-multi-tenancy environment, except the file names of the tenant backup contain the tenant name.

Where to Copy Backup Data for Recovery after a Disaster Recovery

To restore data from a backup in a disaster recovery scenario, you must copy the contents of the desired backup onto the appropriate volumes in the target environment. This content should be stored inside the directory whose name contains the backup ID of the desired backup.

For example, suppose that the backup ID of the desired backup is 2019-10-28T05_33_47_326-0400. You are backing up PostgreSQL data, so the backup PVC is sas-common-backup-data. In this scenario, the directory that contains the backup data that you must copy to the appropriate volumes in the target environment is `sas-common-backup-data-pvc/2019-10-28T05_33_47_326-0400/__default__/postgres`.

How to Find the Physical Location of the Persistent Volume (PV) in an Azure Environment

Note: The following example uses Azure. The steps might be different for you depending on your specific deployment or configuration

In this example, the storage class is azurefile.

- 1 Run the following command on the Kubernetes controller node:

```
kubectl -n name-of-namespace get pv $(kubectl get pvc -n name-of-namespace | grep
persistent_volume_claim | awk '{print $3}') -o json | jq '.spec'
```

Here is an example of the resulting output when running the command for the sas-common-backup-data PVC:

```
{
  "accessModes": [
    "ReadWriteMany"
  ],
  "azureFile": {
    "secretName": "azure-storage-account-fbc3e3ecf1289451c913303-secret",
    "secretNamespace": "d25366",
    "shareName": "kubernetes-dynamic-pvc-23a787c9-9488-4c80-a86e-f9b263d623ce"
  },
  "capacity": {
    "storage": "25Gi"
  },
  "claimRef": {
    "apiVersion": "v1",
    "kind": "PersistentVolumeClaim",
    "name": "sas-common-backup-data",
    "namespace": "d25366",
    "resourceVersion": "4666",
    "uid": "23a787c9-9488-4c80-a86e-f9b263d623ce"
  },
  "persistentVolumeReclaimPolicy": "Delete",
  "storageClassName": "azurefile",
  "volumeMode": "Filesystem"
}
```

- 2 Locate the azureFile section in the above JSON file to find the secretName and shareName to use for mounting the volume.

Here are these values using the sas-common-backup-data PVC output from the prior step:

secretName	azure-storage-account-fbc3e3ecf1289451c913303-secret
shareName	kubernetes-dynamic-pvc-23a787c9-9488-4c80-a86e-f9b263d623ce

- 3 Get the Azure storage account name from the value of `secretName`. Here is an example using the sample values from the prior step:

```
STORAGEACCOUNT=fbc3e3ecf1289451c913303
```

Examples of Transferring Backup Data between the SAS Viya Platform Environment and the Local Filesystem

The following examples use a generic method (`tar` and `kubectl exec`) and a customized pod to copy data between environments. The steps in this generic method can be used for different cloud providers and are not specific to any one cloud provider. You might follow a slightly different set of steps depending on what type of storage you are using for your data.

Note: Make sure that the user that issues the `tar` command to copy files has the appropriate permissions (root level) to copy backup data.

How to Copy Common Backup Data between the SAS Viya Platform Environment and the Local Filesystem Using `tar` and `kubectl exec`

- 1 Determine the location of the data that you wish to copy from or to.
- 2 Navigate to the machine where `kubeconfig` is set to run `kubectl` commands.
- 3 To copy common backup data from the `sas-common-backup-data` PVC to the local filesystem or from the local filesystem to the `sas-common-backup-data` PVC, you must create a Kubernetes job. Create a manifest as follows:
 - a From the base directory of your deployment, create a `backup` subdirectory within the `$deploy/site-config` directory.¹
 - b Navigate to the `$deploy/site-config/backup` directory.
 - c Get an example for creating the Kubernetes job from one of the following files:

File Format	Path
Markdown	<code>\$deploy/sas-bases/examples/backup/README.md</code>
HTML	<code>\$deploy/sas-bases/docs/sas_viya_backup.htm</code>

1. During deployment, you created a deployment directory that SAS recommended that you name `deploy` or an alternative name that is meaningful to you. The directory is referred to as `$deploy` in this guide. Replace `$deploy` with the directory name that you chose.

- d Copy the example to the `$deploy/site-config/backup` directory that you created in the prior step.

- e By default, the example file only mounts the `sas-common-backup-data` PVC location. If you wish to copy directly to another PVC or NFS location that is known to the target environment, you should edit this file now.

- f Create a `kustomization.yaml` file inside the `$deploy/site-config/backup` directory. The `kustomization.yaml` file should have only the following content:

```
resources: - sas-backup-copy-job.yaml
```

- g Navigate to the `$deploy` directory. Make a copy of the `kustomization.yaml` file to recover after temporary changes are made:

```
cp kustomization.yaml kustomization.yaml.save
```

- h Edit the `kustomization.yaml` file in the `$deploy` directory, and add the following line to the `resources` section:

```
resources: - site-config/backup
```

- i Generate and apply the manifest to create the Kubernetes job named `sas-backup-copy-job`.

```
kustomize build -o site-sas-backup-copy-job.yaml
```

```
kubectl -n name-of-namespace apply -f site-sas-backup-copy-job.yaml -l sas.com/job-type=sas-backup-copy-job --server-side --force-conflicts
```

This step runs the `sas-backup-copy-job`.

The job creates the following mount paths for the target environment:

Pod	PVC	Mount path
scheduled backup	sas-common-backup-data	/backup-common-data

- j Get the name of the running pod.

```
kubectl -n name-of-namespace get pods -l sas.com/job-type=sas-backup-copy-job
```

- k Copy the backup data to the target location using the `tar` and `kubectl exec` command.

- 1 Here is an example of the command to copy `sas-common` backup data from the SAS Viya platform environment to the local filesystem:

```
kubectl -n name-of-namespace exec sas-backup-copy-job-pod-name -c sas-backup-agent -- tar cpf - -C /sasviyabackup backup_id | tar xf - -C .
```

Note: In a SAS Viya platform environment, multiple backup data directories exist with the same name. You must keep each backup data directory in a different directory and make a record of the location.

- 2 Here is an example of the command to copy `sas-common` backup data from the local filesystem to the SAS Viya platform environment:

```
tar cpf - backup_id | kubectl -n name-of-namespace exec -i sas-backup-copy-job-pod-name -c sas-backup-agent -- tar xf - -C /sasviyabackup
```

Note: The time required to copy the backup data contents varies based on the size of the backup directory, the disk I/O rate of the system, and the type of file system that you are using.

- I When you have completed copying the data, do the following:

- 1 Remove the temporary manifest.

```
rm site-sas-backup-copy-job.yaml
```

- 2 Remove the sas-backup-copy-job from the target environment.

```
kubectl -n name-of-namespace delete job -l sas.com/job-type=sas-backup-copy-job
```

- 3 Recover the saved copy of kustomization.yaml to remove any previously added modifications:

```
mv kustomization.yaml.save kustomization.yaml
```

How to Copy Other Backup Data between the SAS Viya Platform Environment and the Local Filesystem Using tar and kubectl exec

- 1 Determine the location of the data that you wish to copy from or to.
- 2 Navigate to the machine where kubeconfig is set to run kubectl commands.
- 3 Use the following command to get a list of running pods using a backup PVC from the SAS Viya platform environment:

```
kubectl -n name-of-namespace describe pvc -l "sas.com/backup-role=storage" | grep "Used By\|Mounted By" | awk '{print $3}'
```

Here is an example of the list of running pods might that result.

```
sas-cas-server-default-controller
```

If you have a CAS backup controller configured then the resulting pod name might be sas-cas-server-default-backup which can also be used as the name of the running pod in the next step.

- 4 Repeat this step for each running pod in the list from the previous step.

Copy the backup data to the target location using the the tar and kubectl exec command.

- a Here is an example of the command to copy backup data from the SAS Viya platform environment to the local filesystem:

```
kubectl -n name-of-namespace exec sas-pod_name -c sas-backup-agent -- tar cpf - -C /sasviyabackup backup_id | tar xf - -C .
```

Note: In a SAS Viya platform environment, multiple backup data directories exist with the same name. You must keep each backup data directory in a different directory and make a record of the location.

- b Here is an example of the command to copy backup data from the local filesystem to the SAS Viya platform environment:

```
tar cpf - backup_id | kubectl -n name-of-namespace exec -i sas-pod_name -c sas-backup-agent -- tar xf - -C /sasviyabackup
```

Note: The time required to copy the backup data contents varies based on the size of the backup directory, the disk I/O rate of the system, and the type of file system that you are using.

Copy Backup Data from Azure Environment to Azure Environment

Note: The following example uses Azure file share. The steps might be different for you depending on your specific deployment or configuration

- 1 Determine the location of the PVs used in the source environment as specified in [“Disaster Recovery Prerequisites and Planning on the Source Environment” on page 23](#).
- 2 Go to the target environment machine where the target persistent volumes are mounted or from where they can be accessed. Mount the source environment path identified in the previous step or go to the actual location if it is already available. If target persistent volumes are not mounted, you can use the same steps as used for mounting the source environment persistent volumes.

If you mount the path locally, make sure that you have the appropriate permissions to mount the file system.

Mount the Azure file share/volume using the standard steps contained in the Azure documentation [here](#).

Here is the command to mount the Azure file share to the local directory.

```
sudo mount -t cifs //STORAGEACCT.file.core.windows.net/shareName /mnt/MyAzureFileShare -o vers=3.0,username=STORAGEACCT,password=STORAGEKEY,dir_mode=0777,file_mode=0777,serverino
```

Here is an example of the command when using the sas-common-backup-data PVC output identified in this [output on page 33](#):

```
sudo mount -t cifs //fbc3e3ecf1289451c913303.file.core.windows.net/kubernetes-dynamic-pvc-23a787c9-9488-4c80-a86e-f9b263d623ce /mnt/MyAzureFileShare -o vers=3.0,username=fbc3e3ecf1289451c913303,password=71He1K2Z5j5Bv8hCb5tb+m4zGgfJFXsBSiUejsV/+5x6eWHrFEUPREzHaQpMKF26IjB59qnKWN7Ex8cPP8954Q==,dir_mode=0777,file_mode=0777,serverino
```

- 3 Copy the required backup ID directory contents using the backup ID identified in [step 1 on page 24](#).

```
cp -rp source_mounted_directory/backup_id target_environment_pv_path
```

Here is an example of the command when using the sas-common-backup-data PVC output from the source environment in the previous step:

```
cp -rp /mnt/MyAzureFileShare/backup_id /mnt/TargetAzureFileShare
```

In the above example, `/mnt/TargetAzureFileShare` is the target environment PV location.

Note: The time required to copy the backup data contents varies based on the size of the backup directory, the disk I/O rate of the system, and the type of file system that you are using.

- 4 Repeat steps 1 through 3 to copy the backup ID directory contents for all PVs.
- 5 Make sure that the directory and file permissions are the same as on the source environment.

Determine the Backup ID of the Last Successful Backup

To determine the backup ID of the last successful backup, do the following:

- 1 Determine the PVCs that are associated with backups.

```
kubectl -n namespace get pvc -l "sas.com/backup-role=storage"
```

- 2 Get details about the volume that is named for the sas-common-backup-data PVC.

```
kubectl -n namespace describe pv volume
```

For example, suppose the volume is pvc-5e4c6052-a7c8-49a8-a7b6-866062b63ac2.

```
kubectl -n namespace describe pv pvc-5e4c6052-a7c8-49a8-a7b6-866062b63ac2
```

- 3 Navigate to the sas-common-backup-data PVC directory named in the results of the previous command.
- 4 Validate that the status.json file has {"Status": "Completed"} after the BACKUPID line.

Here is an example of these contents from the status.json file:

```
{
  "BackupID": "2020-10-07T09_23_07_607_0700",
  "Status": "Completed",
  "StartTime": "2020-10-07T09:23:07.645373321Z",
  "EndTime": "2020-10-07T09:24:27.67487685Z",
  -
  -
}
```

Alternatively, you can list all of the successful backups by running the following command from a directory that points to the sas-common-backup-data PVC:

```
grep -r -n -h --include "status.json" -A 3 -i "BackupID" | grep -A 3 "2:[[:space:]]" | grep -B 1 -A 2 -i "completed"
```

Here is an example of the output from the above command when running it from a directory that points to the sas-common-backup-data PVC:

```
2: "BackupID": "2020-10-05T11_32_39_639_0700",
3- "Status": "Completed",
4- "StartTime": "2020-10-05T11:32:39.679649848Z",
5- "EndTime": "2020-10-05T11:34:19.711633701Z",
--
2: "BackupID": "2020-09-28T16_15_25_625_0700",
3- "Status": "Completed
4- "StartTime": "2020-09-28T16:15:25.492563992Z",
5- "EndTime": "2020-09-28T17:20:45.975722312Z",
```

Note: Each backup in the output above is separated by a line containing "--". You can select the BackupID of the backup with the latest timestamp from the above list.

- 5 Make a note of the backup ID and save it in a file in a known location.

Perform an Inventory Scan

Note: If the environment on which you are running the inventory scan was deployed by the SAS Viya Platform Deployment operator, then note the following:

- The SAS Viya Platform Deployment Operator does not participate in the inventory scan.
- The inventory scan should be performed manually without changing the custom resource (CR) file used by the SAS Viya Platform Deployment Operator.

Prior to the 2022.1.4 release of the SAS Viya platform, the inventory scan was run by creating a Kubernetes job by applying a manifest file. This method is not supported in a multi-tenancy environment. As of the 2022.1.4 release of the SAS Viya platform, there is an additional method of running an inventory scan by creating a Kubernetes job from the CronJob. This method does support the scanning of a multi-tenancy environment.

Inventory Scan	Documentation
Run the inventory scan by creating a Kubernetes job by applying a manifest file.	See “Run the Inventory Scan by Creating a Kubernetes Job by Applying a Manifest File” in SAS Viya Platform: Full-System Migration from SAS Viya 3.x. Note: Running an inventory scan by creating a Kubernetes job by applying a manifest file is not currently supported if you are using SCIM for identity management.
Run the inventory scan by creating a Kubernetes job from the CronJob.	Proceed to the steps following this table.

Here are the steps to run an inventory scan by creating a Kubernetes job from the CronJob:

Note: Running the inventory scan by creating a Kubernetes job from the CronJob is supported in a multi-tenancy environment.

As a best practice, in the following steps name the inventory job accordingly when you are running it to generate a comparison report. For example, you might name the job *sas-inventory-comparison-job*. Also, name the inventory job accordingly when you are running the job simply to produce an inventory scan without generating a comparison report. For example, you might name the job *sas-inventory-collector-job*.

- 1 To run the inventory scan in a non-multi-tenancy environment, perform the following steps.

If you wish to generate the inventory comparison report, run an inventory scan on the SAS Viya 4 environment. This scan will be compared to the inventory scan of the source SAS Viya 4 environment.

Note: If the source environment is SAS Viya 4, an initial scan of the SAS Viya 4 source environment must already exist.

```
kubectl -n name-of-namespace create job --from=cronjob/sas-inventory-collector sas-inventory-target-job
```

Delete the prior inventory job, if one exists.

```
kubectl -n name-of-namespace delete job sas-inventory-job-name
```

If you wish to generate the inventory comparison report, run the following commands:

- a** Set the COMPARISON environment variable to true.

```
kubectl -n name-of-namespace set env cronjob/sas-inventory-collector COMPARISON=true
```

- b** Create and run the inventory comparison job.

```
kubectl -n name-of-namespace create job --from=cronjob/sas-inventory-collector sas-inventory-comparison-job
```

If you wish to create and run the inventory job without generating a comparison report, run the following commands:

- a** If the COMPARISON environment variable is set to run a comparison report, use the following command to reset the environment variable.

```
kubectl -n name-of-namespace set env cronjob/sas-inventory-collector COMPARISON-
```

- b** Create and run the inventory job.

```
kubectl -n name-of-namespace create job --from=cronjob/sas-inventory-collector sas-inventory-collector-job
```

To view the logs, use the following command:

```
kubectl -n name-of-namespace logs -l job-name=sas-inventory-job-name
```

To view the job status, use the following command:

```
kubectl -n name-of-namespace get pods -l job-name=sas-inventory-job-name
```

- 2** To run the inventory scan on all tenants, perform the following steps.

If the environment variable is set to run for a single tenant, use the following command to set it for all tenants:

```
kubectl -n name-of-namespace set env cronjob/sas-inventory-collector TENANT-
```

If you wish to generate the inventory comparison report, run an inventory scan on the SAS Viya 4 environment. This scan will be compared to the inventory scan of the source SAS Viya 4 environment.

Note: If the source environment is SAS Viya 4, an initial scan of the SAS Viya 4 source environment must already exist.

```
kubectl -n name-of-namespace create job --from=cronjob/sas-inventory-collector sas-inventory-target-job
```

Delete the prior inventory job, if one exists.

```
kubectl -n name-of-namespace delete job sas-inventory-job-name
```

If you wish to generate the inventory comparison report, run the following commands:

- a** Set the COMPARISON environment variable to true.

```
kubectl -n name-of-namespace set env cronjob/sas-inventory-collector COMPARISON=true
```

- b** Create and run the inventory comparison job.

```
kubectl -n name-of-namespace create job --from=cronjob/sas-inventory-collector sas-  
inventory-comparison-job
```

If you wish to create and run the inventory job without generating a comparison report, run the following commands:

- a** If the COMPARISON environment variable is set to run a comparison report, use the following command to reset the environment variable.

```
kubectl -n name-of-namespace set env cronjob/sas-inventory-collector COMPARISON-
```

- b** Create and run the inventory job.

```
kubectl -n name-of-namespace create job --from=cronjob/sas-inventory-collector sas-  
inventory-collector-job
```

- 3** To run the inventory scan on one tenant, perform the following steps.

If the environment variable is set to run for all tenants, use the following command to set it for one tenant:

```
kubectl -n name-of-namespace set env cronjob/sas-inventory-collector TENANT=tenant_name
```

For example, use the following command to set the environment to run for the acme tenant:

```
kubectl -n name-of-namespace set env cronjob/sas-inventory-collector TENANT=acme
```

If you wish to generate the inventory comparison report, run an inventory scan on the SAS Viya 4 environment. This scan will be compared to the inventory scan of the source SAS Viya 4 environment.

Note: If the source environment is SAS Viya 4, an initial scan of the SAS Viya 4 source environment must already exist.

```
kubectl -n name-of-namespace create job --from=cronjob/sas-inventory-collector sas-  
inventory-target-job
```

Delete the prior inventory job, if one exists.

```
kubectl -n name-of-namespace delete job sas-inventory-job-name
```

If you wish to generate the inventory comparison report, run the following commands:

- a** Set the COMPARISON environment variable to true.

```
kubectl -n name-of-namespace set env cronjob/sas-inventory-collector COMPARISON=true
```

- b** Create and run the inventory comparison job.

```
kubectl -n name-of-namespace create job --from=cronjob/sas-inventory-collector sas-  
inventory-comparison-job
```

If you wish to create and run the inventory job without generating a comparison report, run the following commands:

- a** If the COMPARISON environment variable is set to run a comparison report, use the following command to reset the environment variable.

```
kubectl -n name-of-namespace set env cronjob/sas-inventory-collector COMPARISON-
```

- b** Create and run the inventory job.

```
kubectl -n name-of-namespace create job --from=cronjob/sas-inventory-collector sas-inventory-collector-job
```

To check the status of a tenant job, use the following command:

```
kubectl -n name-of-namespace get pods -l job-name=sas-inventory-job-name
```

To view the log for a tenant job, use the following command:

```
kubectl -n name-of-namespace logs -l job-name=sas-inventory-job-name
```

- 4** For information about scheduling the inventory CronJob, see one of the following files:

File Format	Path
Markdown	<code>\$deploy/sas-bases/examples/sas-inventory-collector/README.md</code>
HTML	<code>\$deploy/sas-bases/docs/using_the_inventory_collector.htm</code>

Note: If any manifest that is in the \$deploy directory is reapplied after the schedule is changed, then you must rerun the command to schedule the inventory Cronjob. Reapplying the manifest resets the schedule to the default schedule.

Sizing Recommendations for Restore

The restore of a PostgreSQL database requires more space than the size of the database in the backup. SAS recommends that you allot 2.5 times the size of the source PostgreSQL database in the space that you are restoring to. For example, if your source PostgreSQL database is 50GB then you should allot 125GB of space for the restore.

The status.json file for each backup contains a sourceSize field for each instance of a PostgreSQL database in the source SAS Viya 4 environment. The sourceSize field contains the size of the source PostgreSQL database. For example, if you are restoring a SAS Viya 4 deployment that contains only the default instance of the PostgreSQL database, the path location of the status.json file and sourceSize field contents will be as follows:

Table 1 Path Location of *status.json* File and *sourceSize* Field Contents

SAS Viya 4 Environment	Path Location of <i>status.json</i> File	Contents of <i>sourceSize</i> field
non-multi-tenant environment	<i>backupID</i> /__default__/postgres/ <i>status.json</i>	size of the SharedServices database
multi-tenant environment with databasePerTenant database mode	<i>backupID</i> / <i>tenantID</i> /postgres/ <i>status.json</i>	size of the tenant database
multi-tenant environment with schemaPerApplicationTenant database mode	<i>backupID</i> / <i>tenantID</i> /postgres/ <i>status.json</i>	aggregated size of all the tenant schemas (those suffixed with <i>_tenantID</i>)

Additionally, you can view the *sourceSize* for all of the instances of a PostgreSQL database in these path locations:

Table 2 Path Location for *sourceSize* of all PostgreSQL Databases

head	head
non-multi-tenant environment	<i>backupID</i> /__default__/ <i>status.json</i>
multi-tenant environment	<i>backupID</i> / <i>tenantID</i> / <i>status.json</i>

Backup and Restore: Troubleshooting

Logs

Overview

Logs for a data source backup are stored with the backup data in the same folder. To view logs in the backup PV for the data source, go to the file system of the backup PV, and navigate to the *backup ID*/restore/*restore ID* folder where the logs are stored.

Find the Pod Name for a Backup Job

List all of the backups to find the name of the backup job:

```
kubectl get jobs -n name-of-namespace -l "sas.com/backup-job-type in (backup,scheduled-backup)"
-L "sas.com/sas-backup-id,sas.com/sas-backup-job-status,sas.com/sas-backup-persistence-status"
```

To find the pod name for the backup job, substituting the backup job name from the last step:

```
kubectl get pods -n name-of-namespace | grep backup-job-name
```

View the Backup Job Log for the Pod

To view the backup job log for the pod:

```
kubectl logs backup-job-pod-name -n name-of-namespace -c sas-backup-job
```

View a Backup Job Log by Backup ID

To view a backup job log by backup ID:

```
kubectl logs -l backup-ID -n name-of-namespace
```

View a Backup Agent's Log

To view a backup agent's log:

```
kubectl logs data-source-pod-name -n name-of-namespace -c sas-backup-agent
```

The data source pod name can be the name of any pod where the backup agent is added as a sidecar. For example, it might be the backup-job-pod-name or the name of any CAS controller pod.

Additional CrunchyData PostgreSQL Jobs and Associated Pods Created After Executing kubectl apply Commands

Jobs or pods that include `postgres-fix` in the name might appear after performing a restore. These jobs and associated pods are only applicable to certain scenarios during an upgrade and have no effect on the system at this time. For example, you might see jobs such as the following:

```
sas-crunchy-cds-postgres-fix-v4-backrest-job
sas-crunchy-cds-postgres-fix-v4-postgres-job
sas-crunchy-cds-postgres-fix-v5-postgres-job
sas-crunchy-platform-postgres-fix-v4-backrest-job
sas-crunchy-platform-postgres-fix-v4-postgres-job
sas-crunchy-platform-postgres-fix-v5-postgres-job
```

To resolve, you can safely ignore these jobs. If you wish to clean up any of these jobs, run the following commands:

```
kubectl -n name-of-namespace get jobs -l "sas.com/admin=none"
kubectl -n name-of-namespace delete jobs -l "sas.com/admin=none"
```

After deletion of the jobs, the associated pods will also automatically be removed.

Backup Fails for fileSystem Source When Using Azure Premium Storage

If the following error results when backing up the fileSystem source and you are using Azure premium storage, then the settings for Azure premium storage might not be set appropriately:

```
08:54:15 {"level":"error","version":1,"source":"sas-backupjob",
"messageKey":"backupjob-log-icu.backup.occurred.error.log","messageParameters":
{"err":"The job failed for data sources: (DataSource: sas-planning SourceType:
fileSystem), (DataSource: cas-shared-default SourceType: fileSystem)"},"properties":
{"logger":"backupjob","caller":"backupjob/main.go:231"},"timeStamp":
"2021-09-06T12:50:49.501779+00:00","message":"An error occurred. Details:
The job failed for data sources: (DataSource: sas-planning SourceType: fileSystem),
(DataSource: cas-shared-default SourceType: fileSystem)"}
```

To resolve, ensure that the Azure premium storage settings are correct. See [“Additional PVC Requirements for Microsoft Azure” in System Requirements for the SAS Viya Platform](#) for more information.

Backup Fails When Services Are Not Completely Started

If the following errors result after performing a backup, then the services might not be started or might still be in the process of starting.

- You might see the following error in the status.json file located in the backup directory here: *backup PV/backup ID/status.json*:

Note: For SAS Viya platform 2022.10 and later environments, *sas-crunchy-platform-postgres* replaces *sas-crunchy-data-postgres* in the following message.

```
{
  "ServiceID": "sas-crunchy-data-postgres",
  "Status": "Failed",
  "StatusCode": 3,
  "Size": "",
  "Logs": [
    "pg_dump --format directory --host sas-crunchy-data-postgres --lock-wait-timeout 300000
--no-blobs --verbose -U dbmsowner -d SharedServices -f
/sasviyabackup/2022-02-22T07_09_07_607_0700/___default___/postgres/SharedServices_pg_dump.dmp -p 5432 ",
    "exit status 1"
  ],
  "StartTime": "2022-02-22T07:09:19.066361261Z",
  "EndTime": "2022-02-22T07:09:28.185899152Z"
}
```

- You might see content such as the following in the PostgreSQL log in the backup directory here: *backup PV/backup ID/___default___/postgres/SharedServices.log*

```
pg_dump: last built-in OID is 16383
pg_dump: reading extensions
pg_dump: identifying extension members
pg_dump: reading schemas
pg_dump: reading user-defined tables
pg_dump: error: query failed: ERROR:  relation "risketl.databasexchangeloglock" does not exist
pg_dump: error: query was: LOCK TABLE risketl.databasexchangeloglock IN ACCESS SHARE MODE
```

To resolve, check the backup logs to ensure that all of the services are completely started before performing a backup. See [“Overview” on page 43](#) for information about the backup logs.