SAS® Viya® Platform: Deployment Notes

2021.1.1 - 2023.08*

* 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.

SAS® Documentation
September 6, 2023

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2021. *SAS® Viya® Platform: Deployment Notes*. Cary, NC: SAS Institute Inc.

SAS® Viya® Platform: Deployment Notes

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

September 2023

v_014-P1:dplynotes

Contents

1

Introduction

Overview

Note: Beginning in September 2022, the cadence versions are being revised to include the month of the release. For details, see "New Versioning Format for SAS Viya Releases" in *What's New in SAS Viya Platform Operations*.

This guide contains deployment notes for SAS Viya. A deployment note describes a change to the deployment process that supported an earlier version of the SAS Viya software. When performing an update, you must review the deployment notes to determine whether changes to the deployment commands or to specific files are necessary in order to deploy your software. Failure to follow the instructions in the notes could lead to unexpected results.

Note: Before using the deployment notes in the process of updating your SAS Viya software, review the full set of steps for the update procedure in *SAS Viya Platform Operations: Updating Software*.

How to Use the Deployment Notes

The title for each chapter in this guide identifies a version of SAS Viya software. The version number represents the version in which the change was made to the deployment process. The chapters are organized from latest version to earliest.

Each chapter is divided into topics that are organized by SAS Viya product or functionality. The topic title contains the product name or functionality description.

Within the topics, deployment notes are further organized by those that should be acted upon before using the commands to update your software and those that should be acted upon after the deployment commands have been performed. If more than one deployment note is included in a topic, each note title describes the change that pertains to the deployment. Each deployment note describes the deployment change in some detail and then provides the specific steps for addressing the change. Instead of providing specific steps, the deployment note can provide a link to the appropriate README.md file for the change or to the deployment guide, if necessary. Follow the instructions in the deployment notes related to SAS Viya products or functionality that you have deployed to ensure that you address all the changes that can affect your update.

IMPORTANT When performing an update, you should review the chapter for each version between the version that you are updating from and the version that you are updating to. For example, if your current version is 2023.2 and you want to update to 2023.5, review the deployment notes for 2023.03, 2023.04, and 2023.05.

2

Long-Term Support Releases

2023.03 (May 2023)

General Deployment

Before Deployment Commands

SAS Viya Platform Deployment Operator Requires an Update

Note: If you do not use the SAS Viya Platform Deployment Operator to deploy your software, skip this note. Additionally, if you do not use a mirror registry in your deployment, skip this note.

The SAS Viya Platform Deployment Operator has been updated with a mandatory change. If you use the operator to deploy your software, you must update the operator by redeploying it. Before you redeploy it, consider the following best practices:

- Save the existing $deploy-operator directory to a different location so that you can refer to it.
- When editing the site-config/transformers.yaml file, the values for the name and the namespace must match existing values.
- If cluster-wide mode is used, edit the kustomization.yaml and uncomment the specified line as was done previously.

When those tasks are completed, redeploy the operator using the instructions at "Deploy the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*.

Update to Crunchy Data 5 for Internal Instances of PostgreSQL

The version of PostgreSQL that SAS uses, Crunchy Data, is upgrading from 4 to 5. Crunchy Data 5 is more declarative in nature than the previous version of Crunchy Data. It releases a new Custom Resource Definition (CRD) named postgresclusters.postgres-operator.crunchydata.com, and it uses fewer containers. Because of the magnitude of the changes of Crunchy Data 5, internal instances of PostgreSQL must go through a disruptive migration process to the new version.

1 If your deployment is on Red Hat OpenShift, ensure that the SAS Viya Platform Deployment Operator is at version 1.83.2 or higher.

   a Determine the version of the SAS Viya Platform Deployment Operator that you are using:

   ```
   kubectl -n name-of-namespace get deployments/sas-deployment-operator -o
   jsonpath='{.metadata.annotations.sas\.com/version}'
   ```

   If you are using the operator in cluster-wide mode, be sure to use the name of the deployment operator namespace for *name-of-namespace*.

   If the output of the command is 1.83.2 or higher, your version of the operator does not need to be updated. Skip to step 2.

   b Update the SAS Viya Platform Deployment Operator using the instructions at "Update the Deployment Operator" in *SAS Viya Platform Operations: Updating Software*.

2 Perform a search of your base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the strings "postgres", "crunchy", and "pgo-client". Delete any line that includes any of those strings unless the line also contains "site-config".

   In the resources block, delete `- site-config/configure-postgres/internal/pgo-client` despite it not meeting the conditions in the first paragraph. Crunchy Data 5 does not support pgo-client. It was used to schedule pgBackRest backups, but with Crunchy Data 5 backups are performed automatically. For more information about backups, see the "Configuration Settings for PostgreSQL Backups" README located at **`$deploy/sas-bases/examples/crunchydata/backups/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/configuration_settings_for_postgresql_backups.htm`** (for HTML format).

3 Create storage transformers to record and protect the state of your existing storage. Storage settings are usually immutable but the declarative nature of Kubernetes can still result in changes if values are not explicitly declared. To prevent unwanted changes, create a storage transformer for each PostgreSQL cluster in your deployment with the storage settings from the cluster.

   IMPORTANT Do not attempt to change the settings while upgrading to Crunchy Data 5. After the upgrade, refer to the comments in the crunchy-storage-transfer.yaml file to determine which settings can be changed and which cannot.

   a Remove any references to old storage transformers (such as `postgres-storage-transformer`) from the transformers block in your base kustomization.yaml.

   b Determine the PostgreSQL clusters in your deployment. Every deployment will have at least one PostgreSQL cluster, sas-crunchy-data-postgres, but some may have a second, sas-crunchy-data-cdspostgres. Run the following command to check:

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

   The output of this command should replace *crunchy-4-cluster-name* in step 3c. If your deployment includes the sas-crunchy-data-cdspostgres cluster, use that name in your second pass through step 3c.

c Follow the steps in the “Configuration Settings for PostgreSQL Storage” README located at **$deploy/sas-bases/examples/crunchydata/storage/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configuration_settings_for_postgresql_storage.htm** (for HTML format), to create a new storage transformer file. In the file, specify the variables as described below.

Note: In the bulleted list below, use the value that is returned by the commands in the storage transformer. Do not replace the returned value with what you feel would be a more appropriate value.

- Find the size of the cluster's PostgreSQL PVC and use the returned value in place of `{{ POSTGRES-STORAGE-SIZE }}` in the file.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name -o
  jsonpath='{.spec.resources.requests.storage}'
  ```

- Find the size of the cluster's pgBackRest PVC and use the returned value in place of `{{ BACKREST-STORAGE-SIZE }}` in the file.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name-pgbr-repo
  -o jsonpath='{.spec.resources.requests.storage}'
  ```

  Note: In the command, the name of the Crunchy Data 4 cluster has `-pgbr-repo` appended to it.

- Find the access mode for the cluster's PostgreSQL PVC and use the returned value in place of `{{ POSTGRES-ACCESS-MODE }}` in the file. Be sure to remove any brackets or quotation marks in the returned value.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name -o
  jsonpath='{.spec.accessModes}'
  ```

- Find the access mode for the cluster's pgBackRest PVC and use the returned value in place of `{{ BACKREST-ACCESS-MODE }}` in the file. Be sure to remove any brackets or quotation marks in the returned value.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name-pgbr-repo
  -o jsonpath='{.spec.accessModes}'
  ```

  Note: In the command, the name of the Crunchy Data 4 cluster has `-pgbr-repo` appended to it.

- Find the storage class for the cluster's PostgreSQL PVC and use the returned value in place of `{{ POSTGRES-STORAGE-CLASS }}` in the file.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name -o
  jsonpath='{.spec.storageClassName}'
  ```

- Find the storage class for the cluster's pgBackRest PVC and use the returned value in place of `{{ BACKREST-STORAGE-CLASS }}` in the file.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name-pgbr-repo
  -o jsonpath='{.spec.storageClassName}'
  ```

  Note: In the command, the name of the Crunchy Data 4 cluster has `-pgbr-repo` appended to it.

- Replace each instance of {{ CLUSTER-NAME }} in the file with the name of the Crunchy Data 5 cluster. For Platform PostgreSQL, use `sas-crunchy-platform-postgres`. For CDS PostgreSQL, use `sas-crunchy-cds-postgres`.

d If your deployment includes the sas-crunchy-data-cdspostgres cluster, create a second storage transformer for CDS PostgreSQL and repeat step 3c for that transformer.

e Include a reference to the newly created storage transformer file or files in the transformer block of the base kustomization.yaml file. For example, if you named the storage transformer platform-postgres-crunchy-storage-transformer.yaml, then the kustomization.yaml file would look like this:

```
transformers:
...
- site-config/crunchydata/platform-postgres-crunchy-storage-transformer.yaml
```

4 If your deployment is managed by the SAS Viya Platform Deployment Operator, skip this step.

Prepare your PostgreSQL database or databases for migration by using the steps described in the README file located at **$deploy/sas-bases/examples/crunchydata/pgo4upgrade/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/upgrade_to_postgresql_operator_v5.htm** (for HTML format). The README includes the direction to add temporary transformers for the migration from Crunchy Data 4 to 5. You will remove those temporary migration transformers later.

5 Modify the base kustomization.yaml file by adding the following lines to the resources and components blocks:

```
resources:
- sas-bases/overlays/postgres/platform-postgres
- sas-bases/overlays/crunchydata/postgres-operator
…
components:
- sas-bases/components/crunchydata/internal-platform-postgres
```

Note: Ensure that any line added to the components block is placed before any lines for TLS components.

If your deployment includes SAS Common Data Store (CDS PostgreSQL), add the following lines to your base kustomization.yaml file also:

```
resources:
- sas-bases/overlays/postgres/cds-postgres
…
components:
- sas-bases/components/crunchydata/internal-cds-postgres
```

6 Refer to the lines in the base kustomization.yaml file that contained "site-config" and which you did not delete from step 2.

a Use this list to find the information to replace the Crunchy Data 4 transformer with the analogous Crunchy Data 5 transformer.

- If you have a transformer named postgres-pods-resource-limits-settings-transformer.yaml, see the “Configuration Settings for PostgreSQL Pod

Resources" README located at **$deploy/sas-bases/examples/crunchydata/pod-resources/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configuration_settings_for_postgresql_pod_resources.htm** (for HTML format).

- If you have a transformer named postgres-replicas-transformer.yaml, see the "Configuration Settings for PostgreSQL Replicas Count" README located at **$deploy/sas-bases/examples/crunchydata/replicas/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configuration_settings_for_postgresql_replicas_count.htm** (for HTML format).
- If you have a transformer that contains "postgres-custom-config", "pghba-custom-config", or "patroni-custom-config", see the "Configuration Settings for PostgreSQL Database Tuning" README located at **$deploy/sas-bases/examples/crunchydata/tuning/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configuration_settings_for_postgresql_database_tuning.htm** (for HTML format).

b Ensure that the Crunchy Data 5 version of the transformers that you create are included in the base kustomization.yaml file.

c Delete the Crunchy Data 4 version of the transformers that you are replacing from the base kustomization.yaml file. Remember that the single tuning and configuration Crunchy Data 5 transformer represents as many as six Crunchy Data 4 transformers, so be sure to delete them all.

7 Add transformers to the base kustomization.yaml file to migrate existing data to Crunchy Data 5.

Note: The transformers added in this step will be removed after the update. For the details, see "Finish Upgrading Crunchy Data" on page 14.

a Migrate the platform PostgreSQL data (`sas-crunchy-data-postgres`) by adding the following content to the transformer block:

```
transformers:
...
- sas-bases/examples/crunchydata/pgo4upgrade/crunchy-upgrade-platform-transformer.yaml
```

b If you have CDS PostgreSQL data (`sas-crunchy-data-cdspostgres`), migrate it by add the following content to the transformer block:

```
transformers:
...
- sas-bases/examples/crunchydata/pgo4upgrade/crunchy-upgrade-cdstransformer.
yaml
```

8 Execute the base kustomization.yaml file validation script to determine whether there are any errors in that file for Crunchy Data 5.

a Copy **$deploy/sas-bases/examples/crunchydata/pgo4upgrade/check_base_kustomization_for_crunchy4_5_uip.sh** to the top level of the $deploy directory.

b Modify the permissions of the script so that it can be run, and then run the script:

```
cd $deploy

chmod +x check_base_kustomization_for_crunchy4_5_uip.sh

./check_base_kustomization_for_crunchy4_5_uip.sh name-of-namespace
```

For *name-of-namespace*, use the namespace that contains the PostgreSQL cluster to be upgraded.

c If any messages are generated by the script, review and revise the base kustomization.yaml file and storage transformers according to the messages. Run the script again until no more messages are generated.

Update to Crunchy Data 5 for External Instances of PostgreSQL

The version of PostgreSQL that SAS uses, Crunchy Data, is upgrading from 4 to 5. As a result, the interface to connect to external instances of PostgreSQL is being upgraded as well, requiring deployments with external instances of PostgreSQL to make disruptive changes.

1 Perform a search of your base kustomization.yaml file ($deploy/kustomization.yaml) and delete any line that contains the strings "postgres" or "sql-proxy".

2 Follow the instructions in the “External PostgreSQL Configuration” section of the “Configure PostgreSQL” README file located at **$deploy/sas-bases/examples/postgres/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configure_postgresql.htm** (for HTML format).

Update the mirror.yaml File

Note: If you are not using a mirror registry, skip this note. If you are not performing a manual update, skip this note.

Changes in the software for version 2023.03 require new images for your deployment. To ensure you retrieve those images, you must update your mirror.yaml file. To update the mirror.yaml file, perform steps 1 and 2 of “Add a Mirror Registry to Your SAS Viya Platform Deployment” in *SAS Viya Platform: Deployment Guide*.

Create Pod Templates for Multi-Tenancy

Beginning in version 2023.03, each tenant in a multi-tenant deployment requires SAS Programming Environment pod templates. Each tenant in an existing multi-tenant deployment must have the pod templates added to it. The steps to add the pod templates are described in the "Create Kubernetes Resources" section of the README file located at **$deploy/sas-bases/examples/sas-tenant-job/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/onboard_or_offboard_tenants.htm** (for HTML format).

Modified change-viya-volume-storage-class.yaml File

The change-viya-volume-storage-class.yaml file has been modified. If you used the change-viya-volume-storage-class.yaml file, you must replace the existing version with the new version in your deployment.

1. In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "change-viya-volume-storage-class.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the yaml file. Go to that location and change the file name to `change-viya-volume-storage-class2.yaml`. For example, if the file is located at **site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml**, change the file name to **site-config/sas-programming-environment/storage/change-viya-volume-storage-class2.yaml**.
3. Copy **sas-bases/examples/sas-programming-environment/storage/change-viya-volume-storage-class.yaml** to the directory in step 2. You should now have a change-viya-volume-storage-class.yaml file and a change-viya-volume-storage-class2.yaml file in the same directory.
4. Using the change-viya-volume-storage-class2.yaml file as a guide, revise the change-viya-volume-storage-class file so that it includes the same values you used in the earlier version of the file. The change requires you to replace `{{ VOLUME-STORAGE-CLASS }}` with the value you used in the original file. For example, assume change-viya-volume-storage-class2.yaml looks like this:

```
kind: PodTemplate
  metadata:
    name: change-viya-volume-storage-class
  template:
   spec:
    volumes:
    - $patch: delete
      name: viya
    - name: viya
      nfs:
        server: myserver.mycompany.com
        path: /path/to/my/location
```

The highlighted code shows the value that was used to replace `{{ VOLUME-STORAGE-CLASS }}` in the original file.Therefore, in change-viya-volume-storage-class.yaml, you would replace `{{ VOLUME-STORAGE-CLASS }}` with the same content.

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      name: viya
      nfs:
        server: myserver.mycompany.com
        path: /path/to/my/location
```

5 After your revisions are complete, delete the change-viya-volume-storage-class2.yaml file.

6 In the base kustomization.yaml file, remove the reference to the change-viya-volume-storage-class.yaml file from the patches block. If the change-viya-volume-storage-class.yaml file was saved to **$deploy/site-config**, the reference would look like this:

```
patches:
- path: site-config/change-viya-volume-storage-class.yaml
  target:
    kind: PodTemplate
    labelSelector: "sas.com/template-intent=sas-launcher"
```

Everything that is highlighted in the example should be removed.

Note: If there are no other patches in the block after the change-viya-volume-storage-class.yaml patch is removed, remove the entire patches block.

7 In the transformers block of the base kustomization.yaml file, add a reference to the change-viya-volume-storage-class.yaml file. The reference must be before the `sas-bases/overlays/required/transformers.yaml` entry. If the change-viya-volume-storage-class.yaml file was saved to **$deploy/site-config**, the reference would look like this:

```
transformers:
...
- site-config/change-viya-volume-storage-class.yaml
- sas-bases/overlays/required/transformers.yaml
```

Changes in GID and UID Management for Identities Microservice

The update to version 2023.03 includes default assignments for new configuration properties for the Identities service. The new default settings might contradict how your existing deployment is configured, resulting in users being unable to access content.

In your SAS Viya platform environment, if your end-users interact with the file system from the SAS Compute Server, CAS server host-launched sessions, or SAS/CONNECT sessions, the UID and GID that are provided by the identities service are crucial to the implementation of file system permissions. The Launcher Service launches pods using user attribute information, including the UID and sometimes the GID, from the identities service. The default option for the SAS/CONNECT Spawner and for host-launched sessions of CAS also uses user attribute information that is returned from the identities service. The changes to the configuration properties affect the way that the identities service obtains UID and GID information.

Before making the updated deployment available to your end users, review the new Identities settings at "Identities Service" in *SAS Viya Platform: Configuration Properties* and revise the configuration to best suit your needs. More details can be found in "Identity Management: Overview" in *SAS Viya Platform: Identity Management*.

CAS and seccomp Mode Changes

The yaml files that configure seccomp mode for CAS have been revised for version 2023.03. If you used these files in earlier versions, you must replace the original files with the new ones.

For the cas-disable-seccomp.yaml file:

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "cas-disable-seccomp.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `cas-disable-seccomp2.yaml`. For example, if the file is located at **`site-config/cas/configure/cas-disable-seccomp.yaml`**, change the file name to **`site-config/cas/configure/cas-disable-seccomp2.yaml`**.
3. Copy **`sas-bases/examples/cas/configure/cas-disable-seccomp.yaml`** to the directory in step 2. You should now have a cas-disable-seccomp.yaml file and a cas-disable-seccomp2.yaml file in the same directory.
4. Using the cas-disable-seccomp2.yaml file as a guide, revise the cas-disable-seccomp.yaml file so that it includes the same values you used in the earlier version of the file.
5. After your revisions are complete, delete the cas-disable-seccomp2.yaml file.

For the cas-seccomp-profile.yaml file:

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "cas-seccomp-profile.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and delete the cas-seccomp-profile.yaml file.
3. Copy **`sas-bases/examples/cas/configure/cas-seccomp-profile.yaml`** to the directory in step 2.
4. Using the comments at the beginning of the cas-seccomp-profile.yaml file as a guide, revise the cas-seccomp-profile.yaml file.

SAS Configurator for Open Source Configuration Change

SAS Configurator for Open Source has revised one of the configuration files used to deploy. As a result, you must replace your old configuration file in order to update successfully.

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.

2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-configuration2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-pyconfig/change-configuration.yaml`**, change the file name to **`$deploy/site-config/sas-pyconfig/change-configuration2.yaml`**.
3. Copy the file in **`$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml`** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.
4. Using the change-configuration2.yaml file as a guide, revise the change-configuration.yaml file so that it includes the same values you used in the earlier version of the file.
5. After your revisions are complete, delete the change-configuration2.yaml file.

Updates to cachelocator and cacheserver Services

Note: This deployment note is an exact copy of the deployment note with the same title for version Long-Term Support 2022.09. The issue effects both the Long-Term Support 2022.09 and Long-Term Support 2023.03 versions, but if you are performing an update that includes both versions you should only perform this task once. The 2023.03 version of the deployment note exists for users whose initial SAS Viya platform deployment is 2022.09 and they are updating to 2023.03.

Because of changes to sas-cachelocator and sas-cacheserver, in order to deploy the new versions, you must scale their replicas to 0 when you update.

```
kubectl -n name-of-namespace scale statefulsets sas-cacheserver --replicas=0
kubectl -n name-of-namespace scale statefulsets sas-cachelocator --replicas=0
```

The update will set the replica count back to 2 if your software contains retail components.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or with sas-orchestration deploy and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2023.03 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Finish Upgrading Crunchy Data

Note: If your deployment uses an external instance of PostgreSQL, skip this note.

Note: The steps described in this deployment note can only be performed after using the deployment note "Update to Crunchy Data 5 for Internal Instances of PostgreSQL" on page 4. If you have not already, go to that note, perform the steps described there, and perform the deployment commands again.

When the Crunchy upgrade has completed, you should see pods that resemble the following:

```
$ kubectl -n name-of-namespace get pod | grep postgres
sas-crunchy-platform-postgres-00-dtbg-0                          5/5     Running   # Three data pods (1 primary + 2 replicas)
sas-crunchy-platform-postgres-00-gkmw-0                          5/5     Running
sas-crunchy-platform-postgres-00-glkn-0                          5/5     Running
sas-crunchy-platform-postgres-backup-95xj-6x74d                  0/1     Completed # Initial backup job
sas-crunchy-platform-postgres-move-pgbackrest-repo-dir-1-td6t5   0/1     Completed # Crunchy 4 to 5 upgrade job
sas-crunchy-platform-postgres-move-pgdata-dir-1-6sjtv            0/1     Completed # Crunchy 4 to 5 upgrade job
sas-crunchy-platform-postgres-repo-host-0                        2/2     Running   # pgBackRest backup/archive repo
sas-crunchy-platform-postgres-repo1-full-27896040-2v7rk          0/1     Completed
sas-crunchy5-postgres-operator-f98f56f96-q25vr                   1/1     Running   # Crunchy Operator
sas-crunchy5-postgres-operator-upgrade-6cd7c58569-7l6tv          1/1     Running   # Crunchy Operator Upgrader
```

Finish the upgrade to Crunchy Data 5:

1. Remove the temporary migration transformers you added to the base kustomization.yaml file ($deploy/kustomization.yaml) before you ran the deployment commands. In the transformers block, remove the following line:

   ```
   transformers:
   - sas-bases/examples/crunchydata/pgo4upgrade/crunchy-upgrade-platform-transformer.yaml
   ```

   If your deployment includes SAS Common Data Store (CDS PostgreSQL), remove this line also:

   ```
   transformers:
   - sas-bases/examples/crunchydata/pgo4upgrade/crunchy-upgrade-cds-transformer.yaml
   ```

2. Patch the newly created PostgresCluster resources.

   ```
   kubectl -n name-of-namespace patch postgrescluster sas-crunchy-platform-
   postgres --type json -p '[{"op": "remove", "path": "/spec/dataSource"}]'
   ```

   If your deployment includes SAS Common Data Store (CDS PostgreSQL), patch the CDS PostgreSQL resource as well.

   ```
   kubectl -n name-of-namespace patch postgrescluster sas-crunchy-cds-postgres
   --type json -p '[{"op": "remove", "path": "/spec/dataSource"}]'
   ```

3. Because names have been changed in Crunchy Data 5, if you have any tools with dependencies on the names of PostgreSQL deployment objects (such as pods, secrets, and configmaps), remove those dependencies.

Remove cachelocator and cacheserver PVCs

With the 2023.03 release of the SAS Viya platform, Redis fully replaces Apache Geode and SAS Cache Server. However, two PVCs from the previous caching software should be removed manually from your deployment.

Perform the following commands to remove the PVCs:

```
kubectl -n name-of-namespace delete pvc -l app.kubernetes.io/name=sas-
cacheserver
kubectl -n name-of-namespace delete pvc -l app.kubernetes.io/name=sas-
cachelocator
```

SAS Dynamic Actuarial Modeling

Before Deployment Commands

Add CDS PostgreSQL to SAS Dynamic Actuarial Modeling

In version 2023.03, an instance of CDS PostgreSQL is being added to SAS Dynamic Actuarial Modeling. A CDS PostgreSQL instance is used because the character of the data it will contain is hierarchically different than the data generally stored in the platform PostgreSQL database. Adding the CDS PostgreSQL instance requires modifying YAML files in the deployment. For the specific steps required, see the "Common Data Store (CDS) PostgreSQL" section of the README file located at `$deploy/sas-bases/examples/postgres/README.md` (for Markdown format) or `$deploy/sas-bases/docs/configure_postgresql.htm` (for HTML format).

Note: For the requirements for an instance of CDS PostgreSQL, see "PostgreSQL Server Requirements" in *System Requirements for the SAS Viya Platform*.

SAS Event Stream Processing

Before Deployment Commands

Required PersistentVolumeClaim with RWX Permissions

Beginning with this version of the SAS Viya platform, SAS Event Stream Processing requires a PVC that is set to RWX. In a multi-tenant deployment, one persistent volume is required per tenant. SAS Event Stream Processing Studio must be configured to use the PV. For more information, see "Managing Persistent Volumes" in *SAS Event Stream Processing: Using SAS Event Stream Processing in a Kubernetes Environment*.

Delete espserver before the Update

When you update to version 2023.03, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.03, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

Configuration File Changes

Files that are used to configure SAS Event Stream Processing have been modified. If you included the transformer files in the base kustomization.yaml file, you must replace the earlier versions of them with updated versions to ensure a successful update.

Note: In previous releases of SAS Event Stream Processing, the espconfig directory was named ESPConfig.

1. In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "espconfig-env-variables.yaml". If that string is not present, go to step 6. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `espconfig-env-variables2.yaml`. For example, if the file is located at **$deploy/site-config/sas-esp-operator/espconfig/espconfig-env-variables.yaml**, change the file name to **$deploy/site-config/sas-esp-operator/espconfig/espconfig-env-variables2.yaml**.
3. Copy **$deploy/sas-bases/examples/sas-esp-operator/espconfig/espconfig-env-variables.yaml** to the directory in step 2. You should now have an espconfig-env-variables.yaml file and an espconfig-env-variables2.yaml file in the same directory.
4. Using the espconfig-env-variables2.yaml file as a guide, revise the espconfig-env-variables.yaml file so that it includes the same values you used in the earlier version of the file.
5. After your revisions are complete, delete the espconfig-env-variables2.yaml file.
6. In your base kustomization.yaml file, search for the string "espconfig-properties.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 7.
7. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `espconfig-properties2.yaml`. For example, if the file is located at **$deploy/site-config/**

`sas-esp-operator/espconfig/espconfig-properties.yaml`, change the file name to `$deploy/site-config/sas-esp-operator/espconfig/espconfig-properties2.yaml`.

8 Copy `$deploy/sas-bases/examples/sas-esp-operator/espconfig/espconfig-properties.yaml` to the directory in step 2. You should now have an espconfig-properties.yaml file and an espconfig-properties2.yaml file in the same directory.

9 Using the espconfig-properties2.yaml file as a guide, revise the espconfig-properties.yaml file so that it includes the same values you used in the earlier version of the file.

10 After your revisions are complete, delete the espconfig-properties2.yaml file.

SAS Inventory Optimization

Before Deployment Commands

Update Warning

Do not update SAS Inventory Optimization to version 2023.03. The latest update contains changes that will break your software deployment. Instead, you should uninstall any deployment you have from before version 2023.03 and perform a new deployment using version 2023.03.

SAS Model Risk Management

Before Deployment Commands

Add CDS PostgreSQL to SAS Model Risk Management

In version 2023.03, an instance of CDS PostgreSQL is being added to SAS Model Risk Management. A CDS PostgreSQL instance is used because the character of the data it will contain is hierarchically different than the data generally stored in the platform PostgreSQL database. Adding the CDS PostgreSQL instance requires modifying YAML files in the deployment. For the specific steps required, see the "Common Data Store (CDS) PostgreSQL" section of the README file located at

$deploy/sas-bases/examples/postgres/README.md (for Markdown format) or **$deploy/sas-bases/docs/configure_postgresql.htm** (for HTML format).

Note: For the requirements for an instance of CDS PostgreSQL, see "PostgreSQL Server Requirements" in *System Requirements for the SAS Viya Platform*.

Update Transformer File

In 2023.03, one of the SAS Model Risk Management transformer files has been modified, requiring you to change a key and the associated value. Prior to 2023.03, the **$deploy/site-config/sas-risk-cirrus-mrm/resources/mrm_transfer.yaml** file contained an `mrm_builder_repo_token` variable for which you used the encoded version of the personal access token. In 2023.03, that variable has been replaced with `SAS_RISK_CIRRUS_SOLUTION_BUILDER_REPO_TOKEN`.

To replace the variable:

1. Find the sas-risk-cirrus-mrm-secret:

   ```
   kubectl -n name-of-namespace get secrets |grep sas-risk-cirrus-mrm-secret
   ```

   The output should look something like this:

   ```
   sas-risk-cirrus-mrm-secret-97725ckf4b        Opaque        4      12d
   ```

   In this example, the name of the secret is `sas-risk-cirrus-mrm-secret-97725ckf4b`.

2. Determine if `mrm_builder_repo_token` was used in the existing deployment:

   ```
   kubectl -n name-of-namespace describe secret secret-name
   ```

   The *secret-name* is the output you got in step 1. If a key is returned, then you must modify the existing deployment. If a key is not returned, no further steps are necessary.

3. Go to **$deploy/site-config/sas-risk-cirrus-mrm/resources/mrm_transfer.yaml**.

4. Edit the file by replacing `mrm_builder_repo_token` with `SAS_RISK_CIRRUS_SOLUTION_BUILDER_REPO_TOKEN`. The `value` line that follows should not be revised unless your Git personal access token has changed. If it has changed, use the new value in this line.

5. Delete the old secret:

   ```
   kubectl -n name-of-namespace patch secret secret-name --type=json -
   p='[{"op": "remove", "path": "/data/mrm_builder_repo_token"}]'
   ```

6. Verify the secret has been removed:

   ```
   kubectl -n name-of-namespace describe secret secret-name
   ```

   If `mrm_builder_repo_token` is not included in the output, then the secret has been successfully deleted.

7. Save and close the **$deploy/site-config/sas-risk-cirrus-mrm/resources/mrm_transfer.yaml** file.

SAS Risk Engine

Before Deployment Commands

Delete sas-risk-samples Job

Before updating to version 2023.03, delete the sas-risk-samples job:

```
kubectl -n name-of-namespace delete job sas-risk-samples
```

If there is no job to delete, the command responds with a message that the job was not found. This is an acceptable result.

Add CDS PostgreSQL to SAS Risk Engine

In version 2023.03, an instance of CDS PostgreSQL is being added to SAS Risk Engine. A CDS PostgreSQL instance is used because the character of the data it will contain is hierarchically different than the data generally stored in the platform PostgreSQL database. Adding the CDS PostgreSQL instance requires modifying YAML files in the deployment. For the specific steps required, see the "Common Data Store (CDS) PostgreSQL" section of the README file located at **`$deploy/sas-bases/examples/postgres/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/configure_postgresql.htm`** (for HTML format).

Note: For the requirements for an instance of CDS PostgreSQL, see “PostgreSQL Server Requirements” in *System Requirements for the SAS Viya Platform*.

2022.09 (November 2022)

General Deployment

Before Deployment Commands

Update the SAS Viya Platform Deployment Operator

Note: If you do not use the SAS Viya Platform Deployment Operator to deploy your software, skip this note.

The SAS Viya Platform Deployment Operator has been updated with a mandatory change. If you use the operator to deploy your software, you should determine if the operator should be updated and, if so, update the operator before performing any other deployment tasks.

1 Run the following command:

```
kubectl -n name-of-namespace get deployments/sas-deployment-operator -
o jsonpath='{.metadata.annotations.sas\.com/version}'
```

If you are using the operator in cluster-wide mode, be sure to use the name of the operator namespace for *name-of-namespace*.

If the output of the command is 1.71.3 or higher, your version of the operator does not need to be updated and you can skip the rest of this note. If the output is less than 1.71.3, continue with step 2.

2 Update the SAS Viya Platform Deployment Operator by following the steps at "Deploy the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*. Here are some best practices to follow while updating the operator:

- Save the existing $deploy-operator directory to a different location so that you can refer to it.
- When editing the site-config/transformers.yaml file, the values for the name and the namespace must match existing values.
- If you are using the operator in cluster-wide mode, edit the kustomization.yaml file and uncomment the specified line as in your previous deployment.

Kubernetes Server-Side Apply

The SAS Viya platform now uses Kubernetes server-side apply (SSA) for select resources deployed to the Kubernetes cluster. SSA is a Kubernetes feature that provides improved field management for applied resources and addresses issues with increasingly large resource specifications.

The SAS Deployment Operator now uses SSA where appropriate. Also, the manual deployment steps for the SAS Viya platform have been modified to enable SSA. For the revised manual deployment steps, see "Deployment Using Kubernetes Commands" in *SAS Viya Platform: Deployment Guide*, especially step 2.

For more information about SSA, see the official Kubernetes SSA documentation.

Default Certificate Generator Change

Beginning in this version, the SAS Viya platform uses openssl as the default certificate generator. The openssl certificate generator is proprietary SAS software that uses the OpenSSL open-source project. SAS recommends using the openssl certificate generator because it is provided to meet the needs of the SAS Viya platform software and is tested for compatibility with each release of the SAS Viya platform.

Note: For more information about certificate generators, see "Certificate Generators" in *SAS Viya Platform Encryption: Data in Motion*. To determine which certificate generator your deployment uses, see "Determine Which Certificate Generator Is Being Used" in "Manage Certificates and Secrets" in *SAS Viya Platform Encryption: Data in Motion*.

If you are updating to this version and had been using cert-manager as your certificate generator but want to use openssl as your certificate generator going forward, perform the following steps:

1. In the resources block of your base kustomization.yaml file, remove the `- sas-bases/overlays/cert-manager-issuer` line.
2. In the transformers block of your base kustomization.yaml file, remove the `- site-config/security/cert-manager-provided-ingress-certificate.yaml` line.
3. If your deployment does not use a customer-provided ingress certificate:
   a. Copy the **`$deploy/sas-bases/examples/security/openssl-generated-ingress-certificate.yaml`** file to **`$deploy/site-config/security/openssl-generated-ingress-certificate.yaml`**.
   b. Add a reference to the copied file to the resources block of the base kustomization.yaml file. Here is an example:

```
resources:
...
- site-config/security/openssl-generated-ingress-certificate.yaml
...
```

4 To complete the conversion to openssl and disassociate the deployment from cert-manager so that cert-manager can safely be removed from your deployment, see "Remove Cert-manager from a SAS Viya Platform Deployment" in *SAS Viya Platform Encryption: Data in Motion*.

To override the default and continue using cert-manager as the certificate generator for your deployment, see the "Using the cert-manager Certificate Generator" section of the security README file located at **$deploy/sas-bases/examples/security/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configure_network_security_and_encryption_using_sas_security_certificate_framework.htm** (for HTML format).

Changes for cert-manager in kustomization.yaml

Note: If you are not using cert-manager as the certificate generator in your SAS Viya platform deployment, skip this note.

The YAML files used in selecting cert-manager as the certificate generator for your SAS Viya platform deployment have been revised.

If you are using cert-manager as your certificate generator:

1 In the base kustomization.yaml file, remove the `- site-config/security/cert-manager-provided-ingress-certificate.yaml` line from the transformers block.

2 In the base kustomization.yaml file, add a reference to the location of the new transformer for the cert-manager-provided-ingress-certificate in the transformers block:

```
transformers:
...

- sas-bases/overlays/cert-manager-provided-ingress-certificate/
ingress-annotation-transformer.yaml
```

Changes to Cross-Origin Resource Sharing Configuration

Changes to third-party software included in the SAS Viya platform may force changes to how you have configured Cross-Origin Resource Sharing (CORS). The `sas.commons.web.security.cors` property can no longer use the asterisk (`*`) wildcard to replace a list of URIs. If you have used the asterisk wildcard, you must replace it with a full list and then restart your services.

For the details to find the `sas.commons.web.security.cors` property and revise it, see "Configure Cross-Origin Resource Sharing" in *SAS Viya Platform: Authentication*. For the instructions to restart your services, see "Managing a SAS Viya Platform Deployment" in *SAS Viya Platform Operations: Servers and Services*.

Removal: $deploy/sas-bases/overlays/network/ingress Directory

The **$deploy/sas-bases/overlays/network/ingress** directory has been removed. Therefore, you must revise the reference to the directory in the base kustomization.yaml file if it has not been revised already.

In the resources block, replace `- sas-bases/overlays/network/ingress` with `- sas-bases/overlays/network/networking.k8s.io`:

```
resources:
- sas-bases/base
- sas-bases/overlays/cert-manager-issuer
- sas-bases/overlays/network/networking.k8s.io
```

The example kustomization.yaml file located at "Create the File" in *SAS Viya Platform: Deployment Guide* reflects this revision.

Same-Site Transformer Change

With the removal of the **$deploy/sas-bases/overlays/network/ingress** directory (see "Removal: $deploy/sas-bases/overlays/network/ingress Directory" on page 24), the content of the same-site transformer from previous versions of the SAS Viya platform has been moved to a different file and location.

To determine if your deployment is affected and what actions to take:

1. Examine the base kustomization.yaml file (**$deploy/kustomization.yaml**). In the transformers block, look for the following line:

   ```
   - sas-bases/overlays/network/ingress/security/transformers/sas-ingress-cookie-samesite-transformer.yaml
   ```

   If the line is present, continue with these steps. Otherwise, skip the rest of this note.

2. Remove the `- sas-bases/overlays/network/ingress/security/transformers/sas-ingress-cookie-samesite-transformer.yaml` line.

3. In the components block of the base kustomization.yaml file, add the following line.

   ```
   components:
   ...
   - sas-bases/components/security/web/samesite-none
   ```

Configure Internal Instances of PostgreSQL

IMPORTANT If you are performing an update up to or past version 2022.10 (October 2022), you should skip this deployment note since the version of PostgreSQL is being upgraded with that update.

Note: If your deployment uses an external instance of PostgreSQL, skip this note.

Changes to the deployment of internal instances of PostgreSQL require revisions to the base kustomization.yaml file when you update to version 2022.09. Those changes also require evaluating the space available for backups and potentially cleaning up the space used for backups in order to prevent Disk Full errors. For the full set of tasks to be performed, see Perform Required PostgreSQL Server Maintenance.

Note: There is a post-deployment validation step associated with the changes described in this deployment note. See "Verify pgBackRest Backup Schedule" on page 123 for the details.

Connect Workload Class Changes

SAS/CONNECT provides the ability for clients to launch SAS compute processes within a local or remote SAS environment. The SAS/CONNECT Spawner is a SAS Viya platform service that launches these processes on behalf of SAS/CONNECT clients. The processes can be launched in their own pods (referred to as "dynamically launched pods") or in the SAS/CONNECT Spawner pod.

Beginning in this version, SAS/CONNECT Spawner is deployed in the stateless work class by default. The connect workload class is only required if you are not using dynamically launched pods. If you choose not to use dynamically launched pods, you must add a transformer to your base kustomization.yaml file.

Note: For more information about dynamically launched pods, see "Workload Classes" in *SAS Viya Platform: Deployment Guide*.

To add the use-connect-workload-class.yaml transformer, revise the transformers block of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) to include the path to the transformer:

```
transformers:
...
- sas-bases/overlays/sas-connect-spawner/enable-spawned-servers/use-
connect-workload-class.yaml
```

If you are using dynamically launched pods, you will see the connect-spawner being scheduled onto stateless or stateful nodes. The connect node will no longer be used by the connect-spawner. You should consider removing connect workload class nodes from your cluster infrastructure.

Updates to cachelocator and cacheserver Services

Because of changes to sas-cachelocator and sas-cacheserver, in order to deploy the new versions, you must scale their replicas to 0 when you update to release 2022.09.

```
kubectl -n name-of-namespace scale statefulsets sas-cacheserver --replicas=0
kubectl -n name-of-namespace scale statefulsets sas-cachelocator --replicas=0
```

The update will set the replica count back to 2.

CAS Configuration Change

In version 2022.09, a file that is used to configure CAS has been modified. If you included the transformer file in the base kustomization.yaml file, you must replace the earlier version of it with an updated version to ensure a successful update.

1. In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "cas-enable-host.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `cas-enable-host2.yaml`. For example, if the file is located at **site-config/cas/configure/cas-enable-host.yaml**, change the file name to **site-config/cas/configure/cas-enable-host2.yaml**.
3. Copy **sas-bases/examples/cas/configure/cas-enable-host.yaml** to the directory in step 2. You should now have a cas-enable-host.yaml file and a cas-enable-host2.yaml file in the same directory.
4. Using the cas-enable-host2.yaml file as a guide, revise the cas-enable-host.yaml file so that it includes the same values you used in the earlier version of the file.
5. After your revisions are complete, delete the cas-enable-host2.yaml file.

Remove Redis Objects

If you have had a deployment since at least version 2021.2.3, there is a possibility that there are Redis objects in your deployment that could prevent successful updates to version 2022.09. Use the following steps to determine if those objects are present and, if so, remove them.

1. Exec into the sas-consul-server-0 pod:

   ```
   kubectl -n name-of-namespace exec -it sas-consul-server-0 -c sas-consul-server -- /bin/bash
   ```

2. Run the following commands:

   ```
   export CONSUL_HTTP_ADDR=https://localhost:8500
   ```

   Note: If your deployment uses Front-door TLS or no TLS, replace `https` with `http`.

   ```
   /opt/sas/viya/home/bin/sas-bootstrap-config kv read config/ --recurse | grep -i redis
   ```

   If the command returns any results, continue with step 3. If you have no results, no further steps are necessary.

3. Delete the Consul key/value pairs that are associated with Redis with the following commands:

   ```
   /opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/sas.cache.redis/host
   ```

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/
sas.cache.redis/password

/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/
sas.cache.redis/tls

/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/
sas.cache.redis/username
```

Scale Down Consul Before Update

Note: If you are performing updates with the SAS Viya Platform Deployment Operator, skip this note.

IMPORTANT Ensure that you perform the tasks described in this note after you perform the tasks described in the deployment note, "Remove Redis Objects" on page 26.

If you are updating your SAS Viya platform deployment manually, you must scale down Consul before performing the update. Use this command:

```
kubectl -n name-of-namespace scale statefulsets sas-consul-server --replicas=0
```

SAS Configurator for Open Source Upgrade

SAS Configurator for Open Source upgrades to version 1.8 with this version of the SAS Viya platform. Use the following steps to ensure a successful upgrade.

1. In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-configuration2.yaml`. For example, if the file is located at **$deploy/site-config/sas-pyconfig/change-configuration.yaml**, change the file name to **$deploy/site-config/sas-pyconfig/change-configuration2.yaml**.
3. Copy the file in **$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.
4. Follow the instructions in the SAS Configurator for Open Source README to revise the change-configuration.yaml file to enable support for R, Python, or both. The README file is located at **$deploy/sas-bases/examples/sas-pyconfig/README.md** (for Markdown format) or **$deploy/sas-bases/docs/sas_configurator_for_open_source_options.htm** (for HTML format).

Replace YAML File for SAS Configurator for Open Source

The SAS Configurator for Open Source has been rewritten to run from a CronJob rather than a Kubernetes Job. As a result, if you have deloyed the SAS Configurator for Open Source, you must replace one of the YAML files that it uses for deployment.

1. Evaluate the transformers block of the base kustomization.yaml file. If the block contains a line that includes "change-limits.yaml", continue with the steps in this note. Otherwise, skip to the next deployment note.
2. Evaluate the content of the $deploy/site-config/sas-pyconfig/change-limits.yaml file. If the word "CronJob" appears in the file, then you have performed these steps in an earlier cadence and can skip the rest of this deployment note.
3. Change the name of the $deploy/site-config/sas-pyconfig/change-limits.yaml file to change-limits-2.yaml.
4. Copy `$deploy/sas-bases/examples/sas-pyconfig/change-limits.yaml` to `$deploy/site-config/sas-pyconfig/change-limits.yaml`.
5. Set the values for CPU and memory in the new change-limits.yaml file by using the values in the change-limits-2.yaml file.
6. Delete the change-limits-2.yaml file.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2022.09 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Required PostgreSQL Configuration

IMPORTANT If you are performing an update up to or past version 2022.10 (October 2022), you should skip this deployment note since the version of PostgreSQL is being upgraded with that update.

Note: If your deployment uses an external instance of PostgreSQL, skip this note.

Every update to the PostgreSQL database generates WAL (Write-Ahead-Log) data. WAL data grows much faster than the database, filling up the disk and causing a Disk Full Error if it is not regularly cleaned up. If you have not done so already, use the instructions at Perform Required PostgreSQL Server Maintenance set up a scheduled pgBackRest backup with a retention policy to clean up old backups and unnecessary WAL data.

Verify pgBackRest Backup Schedule

IMPORTANT If you are performing an update up to or past version 2022.10 (October 2022), you should skip this deployment note since the version of PostgreSQL is being upgraded with that update.

Note: If your deployment uses an external instance of PostgreSQL, skip this note.

This deployment note is the verification step for the pgBackRest backup schedule created through the deployment note “Configure Internal Instances of PostgreSQL” on page 121. If you did not perform the tasks described in that deployment note, go back to it and perform those tasks before attempting the validation described here.

To verify the pgBackRest backup schedule that you created earlier, follow the steps described in Verify That the Backup Schedule Was Created.

SAS Visual Analytics

Before Deployment Commands

Reduce Required Memory for SAS Conversation Designer

The SAS Visual Analytics offering contains SAS Conversation Designer and the SAS Natural Language Understanding API, which require at least 4 GB of memory to run. If you are not using SAS Conversation Designer and the SAS Natural Language Understanding API, beginning in this version, you can reduce their memory allocation by approximately 3.5 GB. The instructions for reducing the memory allocation are located at the README file at **`$deploy/sas-bases/overlays/sas-natural-language-understanding/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/reduce_memory_resources_for_sas_natural_language_understanding.htm`** (for HTML format).

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2022.09, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2022.09, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

Configuration File Changes

Files that are used to configure SAS Event Stream Processing have been modified. If you included the transformer files in the base kustomization.yaml file, you must replace the earlier versions of them with updated versions to ensure a successful update.

1. In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "espconfig-env-variables.yaml". If that string is not present, go to step 6. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `espconfig-env-variables2.yaml`. For example, if the file is located at **$deploy/site-config/sas-esp-operator/ESPConfig/espconfig-env-variables.yaml**, change the file name to **$deploy/site-config/sas-esp-operator/ESPConfig/espconfig-env-variables2.yaml**.
3. Copy **$deploy/sas-bases/examples/sas-esp-operator/ESPConfig/espconfig-env-variables.yaml** to the directory in step 2. You should now have an espconfig-env-variables.yaml file and an espconfig-env-variables2.yaml file in the same directory.
4. Using the espconfig-env-variables2.yaml file as a guide, revise the espconfig-env-variables.yaml file so that it includes the same values you used in the earlier version of the file.
5. After your revisions are complete, delete the espconfig-env-variables2.yaml file.
6. In your base kustomization.yaml file, search for the string "espconfig-properties.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 7.
7. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `espconfig-properties2.yaml`. For example, if the file is located at **$deploy/site-config/sas-esp-operator/ESPConfig/espconfig-properties.yaml**, change the file name to **$deploy/site-config/sas-esp-operator/ESPConfig/espconfig-properties2.yaml**.
8. Copy **$deploy/sas-bases/examples/sas-esp-operator/ESPConfig/espconfig-properties.yaml** to the directory in step 2. You should now have an espconfig-properties.yaml file and an espconfig-properties2.yaml file in the same directory.
9. Using the espconfig-properties2.yaml file as a guide, revise the espconfig-properties.yaml file so that it includes the same values you used in the earlier version of the file.
10. After your revisions are complete, delete the espconfig-properties2.yaml file.

Directory Name Change

The **$deploy/sas-bases/examples/sas-esp-operator/ESPconfig** has been renamed

`$deploy/sas-bases/examples/sas-esp-operator/espconfig`

If you have automated your deployment, you should determine if the directory name change requires changes to that automation.

2022.1 (May 2022)

General Deployment

Before Deployment Commands

Deprecation Notice: The $deploy/sas-bases/overlays/network/ingress Directory

The **`$deploy/sas-bases/overlays/network/ingress`** directory is deprecated. The directory and the overlay that it contains will be removed from the the SAS Viya platform deployment assets in the long-term support 2022.1 version.

Therefore, you must revise the reference to the directory in the base kustomization.yaml file if it has not been revised already. In the resources block, replace `- sas-bases/overlays/network/ingress` with `- sas-bases/overlays/network/networking.k8s.io`:

```
resources:
- sas-bases/base
- sas-bases/overlays/cert-manager-issuer
- sas-bases/overlays/network/networking.k8s.io
```

The example kustomization.yaml file located at "Create the File" in *SAS Viya Platform: Deployment Guide* reflects this revision.

Support for PostgreSQL CustomResourceDefinition with Kubernetes API v1

Note: If you are using an external instance of PostgreSQL, skip this deployment note.

Note: If you are using only one deployment of the SAS Viya platform in your Kubernetes cluster, skip this deployment note.

Before version 2022.1, the PostgreSQL CustomResourceDefinition (CRD) used the v1beta1 Kubernetes API. However, the v1beta1 API has been deprecated and removed in Kubernetes 1.22. A revised PostgreSQL CRD using Kubernetes API v1 is being introduced in this version of the SAS Viya platform. Therefore, if you are updating any deployment of the SAS Viya platform with an internal instance of PostgreSQL in a Kubernetes cluster that contains multiple deployments of the SAS Viya platform with internal instances of PostgreSQL, each deployment in the cluster must be fully updated to version 2021.2 before updating any of them to version 2022.1.

Changes in the Directory Structure for Multi-tenancy

Note: If your deployment does not have multi-tenancy enabled, skip this note.

Changes in the directory structure for multi-tenancy require performing some steps before updating your SAS Viya platform software.

1. If you are unsure whether you enabled multi-tenancy, review your base kustomization.yaml file. If the site-config/multi-tenant/sas-shared-configmap-transformer.yaml file is listed in the transformers block, multi-tenancy is enabled and you should continue with these steps.
2. Delete the **`$deploy/site-config/multi-tenant/sas-shared-configmap-transformer.yaml`** file.
3. In the base kustomization.yaml file, remove the reference to the site-config/multi-tenant/sas-shared-configmap-transformer.yaml file and replace it with a reference to sas-bases/overlays/multi-tenant/sas-shared-configmap-transformer.yaml. Here is an example of the base kustomization.yaml file before the change:

   ```
   transformers:
   ...
   - site-config/multi-tenant/sas-shared-configmap-transformer.yaml
   ```

   Here is an example of the same file after the change:

   ```
   transformers:
   ...
   - sas-bases/overlays/multi-tenant/sas-shared-configmap-transformer.yaml
   ```

Change in Storage Class Configuration for SAS Programming Environment

The method by which you assign the storage class for runtime storage for processes such as SAS Compute server, SAS/CONNECT server, and SAS Batch server has been modified. If you used a YAML file to assign the storage class rather than use the default emptyDir, you must make revisions to your deployment.

Note: For more information about assigning storage classes for the SAS Programming Environment, see the README at **`$deploy/sas-bases/examples/sas-programming-environment/storage/README.md`** (for Markdown format) or at

$deploy/sas-bases/docs/sas_programming_environment_storage_tasks.htm (for HTML format).

1. Search for the following line in the transformers block of the base kustomization.yaml file.

   ```
   - site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml
   ```

   If that line is present, continue with these steps. Otherwise, there are no required changes and you should skip the rest of this deployment note.

2. Copy the **$deploy/sas-bases/examples/sas-programming-environment/storage/change-viya-volume-storage-class.yaml** file to the site-config directory, overwriting the existing file.

3. In the copied file, replace the `{{ VOLUME-STORAGE-CLASS }}` variable with the volume definition of the storage location you want to use.

4. In the base kustomization.yaml file, remove the `- site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml` from the transformers block.

5. Also in the base kustomization.yaml file, add the following content to the patches block:

   ```
   patches:
   - path: site-config/change-viya-volume-storage-class.yaml
     target:
       kind: PodTemplate
       labelSelector: "sas.com/template-intent=sas-launcher"
   ```

Delete sas-admin-content-loader Job

Note: If you are updating your software with the SAS Viya Platform Deployment Operator, skip this note.

If you use the manual process to update your software to version 2022.1, you must delete the sas-admin-content-loader Job before beginning the update:

```
kubectl -n name-of-namespace delete job sas-admin-content-loader
```

Spawn SAS/CONNECT Servers Within the Spawner Pod

Beginning in 2022.1, the ability to spawn servers within the spawner pod has been removed from SAS/CONNECT Spawner. However, this capability may be re-introduced by applying the security settings in the enable-spawned-servers.yaml example file. For the steps to re-enable spawning servers within the spawner pod, see the “Allow the Ability to Spawn Servers within the Spawner Pod” section of the “Configure SAS/CONNECT Spawner in SAS Viya” README file located at **$deploy/sas-bases/examples/sas-connect-spawner/README.md** (for Markdown format) and at **$deploy/sas-bases/docs/configure_sasconnect_spawner_in_sas_viya.htm** (for HTML format).

Remove login_hint for SAS Logon Manager

Prior to the 2022.1 version, users could apply a patch transformer when deploying the SAS Viya platform to update the SAS Logon Manager ingress definition. This patch transformer used a server snippet annotation to insert a login_hint into requests. Adding the login_hint allowed customers using SAML or OpenID Connect to bypass SAS Logon Manager, meaning that end-users would automatically get redirected to the third-party SAML or OIDC Identity Provider (IdP). However, an issue with custom snippets was discovered in ingress-nginx, as documented in CVE-2021-25742. Therefore, the patch transformer must be removed for version 2022.1 and later.

To remove the patch transformer, review the transformers block of the base kustomization.yaml file. If the following line is in that block, remove it.

```
- site-config/sas-logon-app/login-hint-transformer.yaml
```

If the line is not present, then no action needs to be taken.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2022.1 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Check Open Distro for Elasticsearch Cluster

Note: If your deployment does not contain Open Distro for Elasticsearch, skip these instructions.

Due to a timing issue, an update can result in an invalid StatefulSet manifest, which prevents the sas-opendistro-default-0 (or other master/data nodes) from coming up.

1 After updating the SAS Viya platform, ensure that the Open Distro for Elasticsearch cluster pods are running. If your deployment uses the default topology for Open Distro for Elasticsearch, use the following command:

   ```
   kubectl -n name-of-namespace get pods | grep sas-opendistro-default
   ```

   If you created a custom topology using the custom-topology.yaml example file, use the following command:

   ```
   kubectl -n name-of-namespace get pods | grep sas-opendistro-node-set
   ```

   In this command, *node-set* is a custom Elasticsearch node set defined in the example file. If you have more than one custom node set, the command should be performed for each node set.

2 Restart each pod that is in a `CrashLoopBackOff` state in the output from step 1. To restart an Elasticsearch pod in a default topology, delete it:

   ```
   kubectl -n name-of-namespace delete pod sas-opendistro-default-pod-number
   ```

   To restart an Elasticsearch pod in a custom topology:

   ```
   kubectl -n name-of-namespace delete pod sas-opendistro-node-set-pod-number
   ```

3 Review the log of each sas-opendistro-default-*pod-number* pod (or each sas-opendistro-*node-set-pod-number* pod if you have custom topologies) for the following error.

   ```
   org.elasticsearch.bootstrap.StartupException: java.lang.IllegalArgumentException:
   unknown setting [node.roles] please check that any required plugins are
   installed, or check the breaking changes documentation for removed settings
   ```

   If the error is in the log, restart the pod using the commands described in step 2.

Ignore Log Entry from Crunchy Data

An update can result in an error entry in the sas-crunchy-data-postgres pod log that you can safely ignore.

When you update the SAS Viya platform from version 2021.2 to version 2022.1, you might see an error entry if you are using the internal instance of SAS Infrastructure Data Server. The internal instance is based on Crunchy PostgreSQL for

Kubernetes. After the update, the sas-crunchy-data-postgres pod log includes the following entry:

```
level=error msg=queryNamespaceMappings returned 1 errors
```

SAS testing has determined that this error does not affect the functionality of the deployment. SAS is working with Crunchy Data to remove the relevant error reporting.

Avoid Long Delay for SAS/CONNECT Server Sign-on from SAS Studio

Setting `TCPNOIPADDR` to 1 avoids a long (2–minute) delay during sign-on to a SAS/CONNECT server from an internal SAS Viya platform client, such as SAS Studio.

This problem has been observed on Microsoft Azure, AWS, and GCP. The delay is caused by an errant connection back to the SAS/CONNECT spawner. The spawner hangs until time-out, at which point a successful connection is made to the launched SAS/CONNECT server.

A fix is planned. In the meantime, run your programs with `TCPNOIPADDR=1` to work around the issue that causes the delay.

Multi-tenancy and SAS Micro Analytic Service ASTORE and Resource Files

Changes in how SAS Micro Analytic Service handles analytic store (ASTORE) and resource files require changes to the configuration of those files.

1. To determine if this change affects your deployment, examine your base kustomization.yaml file.
   a. Look for the following line in the transformers block of that file:

      ```
      - sas-bases/overlays/multi-tenant/sas-shared-configmap-transformer.yaml
      ```

      If the line is present, continue with the steps in this note. Otherwise, your deployment does not require this change and you should skip the rest of this note.

   b. Look for the following lines, also in the transformer block of the base kustomization.yaml file:

      - `- sas-bases/overlays/sas-microanalytic-score/astores/astores-transformer.yaml`
      - `- site-config/sas-open-source-config/python/python-transformer.yaml`

      If either of these lines is present, continue with the steps in this note. If neither of these lines are present, your deployment does not require this change and you should skip the rest of this note.

2. Perform the steps described in the “Multi-Tenant Configuration” section of the “Configure SAS Micro Analytic Service to Support Analytic Stores” README, located at **`$deploy/sas-bases/examples/sas-microanalytic-store/astores/README.md`** (for Markdown format) or **`$deploy//sas-bases/docs/configure_sas_micro_analytic_service_to_support_analytic_stores.htm`** (for HTML format).

3 After completing the steps in the README, perform the steps to republish any existing models for the offerings that use SAS Micro Analytic Score:

- To republish model objects from SAS Model Manager, see "Publishing Models" in *SAS Model Manager: User's Guide*.
- To republish decisions that contain models from SAS Intelligent Decisioning, see "Publishing a Decision" in *SAS Intelligent Decisioning: User's Guide*.
- To republish analytic store models published from Model Studio, see "Publish Models" in *SAS Viya: Machine Learning User's Guide*.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2022.1 the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2022.1, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.2 (November 2021)

General Deployment

Before Deployment Commands

Changes to TLS

Note: If your SAS Viya platform deployment does not use TLS, you should skip this deployment note.

Beginning with release 2021.2, the SAS Viya platform uses Kustomize components to deploy and configure TLS. Moving to the new process requires a number of changes to your base kustomization.yaml file (**`$deploy/kustomization.yaml`**).

1. In the resources block, remove the `sas-bases/overlays/network/ingress/security` line.
2. In the transformers block, remove any line that includes `tls`. Here are lines that should be removed:
   - `sas-bases/overlays/network/ingress/security/transformers/product-tls-transformers.yaml`
   - `sas-bases/overlays/network/ingress/security/transformers/ingress-tls-transformers.yaml`
   - `sas-bases/overlays/network/ingress/security/transformers/backend-tls-transformers.yaml`
   - `sas-bases/overlays/network/ingress/security/transformers/truststore-transformers-without-backend-tls.yaml`
   - `sas-bases/overlays/network/ingress/security/transformers/cas-connect-tls-transformers.yaml`
3. To enable NGINX TLS for full-stack TLS mode, add the following lines to the components block of the base kustomization.yaml file. Create the components block if it does not already exist.

```
components:
- sas-bases/components/security/core/base/full-stack-tls
- sas-bases/components/security/network/networking.k8s.io/ingress/nginx.ingress.kubernetes.io/full-stack-tls
```

For information about the required additions to the base kustomization.yaml for other modes of TLS, see the README file at **$deploy/sas-bases/examples/security/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_network_security_and_encryption_using_sas_security_certificate_framework.htm** (for HTML format).

Note: The example kustomization.yaml file at "Create the File" in *SAS Viya Platform: Deployment Guide* contains the appropriate changes to enable NGINX TLS for full-stack TLS mode.

Revise Reference to Ingress in Initial kustomization.yaml File

Because of changes to Kubernetes, the reference to ingress in the base kustomization.yaml file for earlier versions of SAS Viya must be revised for version 2021.2.

Here is how the base kustomization.yaml file should appear for earlier versions:

```
resources:
- sas-bases/base
- sas-bases/overlays/cert-manager-issuer
- sas-bases/overlays/network/ingress
```

Here is the same block for version 2021.2:

```
resources:
- sas-bases/base
- sas-bases/overlays/cert-manager-issuer
- sas-bases/overlays/network/networking.k8s.io
```

Internal Instances of PostgreSQL Moving to Operator

IMPORTANT If you are performing an update up to or past version 2022.10 (October 2022), you should skip this deployment note since the version of PostgreSQL is being upgraded with that update.

Note: If you have an external instance of PostgreSQL, skip this deployment note.

Starting with version 2021.2, deployment and administration of internal instances of PostgreSQL are performed with the SAS Data Server Operator. If you are updating to version 2021.2, you must remove the references to the previous deployment method and replace them with operator-specific files.

1 In the base kustomization.yaml file, make the following changes:

- Remove `- sas-bases/overlays/crunchydata` from the resources block.
- Remove `- sas-bases/overlays/internal-postgres/internal-postgres-transformer.yaml` from the transformers block.

- Remove `- site-config/postgres/postgres-custom-config.yaml` from the generators block. If it is the only file listed in that block, remove the block as well.

2 To apply custom resource limits:

   a Copy the **`$deploy/sas-bases/examples/postgres/pod-resource-limits-settings/postgres-pods-resource-limits-settings-transformer.yaml`** file to your site-config directory. If postgres-pods-resource-limits-settings-transformer.yaml already exists in the site-config directory, delete the old file and replace it with the new.

   b Using the comments in the file, revise the copied file with the appropriate values for the internal instance of PostgreSQL that it represents.

   Note: By default, the values apply to all PostgreSQL clusters. If you want different values among the clusters, make an individual entry per cluster.

   c In the transformers block of the base kustomization.yaml file, add a reference to the postgres-pods-resource-limits-settings-transformer.yaml file. Here is an example that assumes the copied file was placed in **`$deploy/site-config/postgres/pod-resource-limits-settings/postgres-pods-resource-limits-settings-transformer.yaml`**:

```
transformers:
...
- site-config/postgres/pod-resource-limits-settings/postgres-pods-resource-limits-settings-transformer.yaml
...
```

   If you replaced an existing version of postgres-pods-resource-limits-settings-transformer.yaml with a new one for this update, do not change the base kustomization.yaml file since the reference already exists.

3 To use custom replica counts:

   a Copy the **`$deploy/sas-bases/examples/postgres/replicas/postgres-replicas-transformer.yaml`** file to your site-config directory. If postgres-replicas-transformer.yaml already exists in the site-config directory, delete the old file and replace it with the new.

   b Using the comments in the file, revise the copied file with the appropriate values for the internal instance of PostgreSQL that it represents.

   Note: By default the values apply to all PostgreSQL clusters. If you want different values among the clusters, make an individual entry per cluster.

   c In the transformers block of the base kustomization.yaml file, add a reference to the postgres-replicas-transformer.yaml file. Here is an example that assumes the copied file was placed in **`$deploysite-config/postgres/replicas/postgres-replicas-transformer.yaml`**:

```
transformers:
...
- site-config/postgres/replicas/postgres-replicas-transformer.yaml
...
```

If you replaced an existing version of postgres-replicas-transformer.yaml with a new one for this update, do not change the base kustomization.yaml file since the reference already exists.

4 To use custom storage settings:

   a Copy the **$deploy/sas-bases/examples/postgres/storage/postgres-storage-transformer.yaml** file to your site-config directory. If postgres-storage-transformer.yaml already exists in the site-config directory, delete the old file and replace it with the new.

   b Using the comments in the file, revise the copied file with the appropriate values for the internal instance of PostgreSQL that it represents.

   Note: By default the values apply to all PostgreSQL clusters. If you want different values among the clusters, make an individual entry per cluster.

   c In the transformers block of the base kustomization.yaml file, add a reference to the postgres-storage-transformer.yaml file. Here is an example that assumes the copied file was placed in **$deploy/site-config/postgres/storage/postgres-storage-transformer.yaml**:

   ```
   transformers:
   ...
   - site-config/postgres/storage/postgres-storage-transformer.yaml
   ...
   ```

   If you replaced an existing version of postgres-storage-transformer.yaml with a new one for this update, do not change the base kustomization.yaml file since the reference already exists.

5 Determine whether Common Planning Service is included in your deployment by running the following command:

   ```
   kubectl -n name-of-namespace get pods | grep cpspostgres
   ```

   If the results include any pod names, then Common Planning Service is included in your deployment. You should consider potential configurations for the Common Planning Service instance of PostgreSQL in the remaining steps.

6 To use custom PostgreSQL settings:

   a Copy the following files to your site-config directory:

      - **$deploy/sas-bases/examples/postgres/custom-config/sas-postgres-custom-config.yaml**
      - **$deploy/sas-bases/examples/postgres/custom-config/sas-postgres-custom-config-transformer.yaml**
      - If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspostgres-custom-config.yaml**
      - If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspostgres-custom-config-transformer.yaml**

   b Using the comments in the files, revise the copied files with the appropriate values for the internal instance of PostgreSQL that it represents.

c In the base kustomization.yaml file, add references to the modified files.

- In the resources block, add references for sas-postgres-custom-config.yaml and sas-cpspostgres-custom-config.yaml (if needed). Here is an example that assumes the copied files were placed in **$deploy/site-config/postgres/custom-config**:

```
resources:
...
- site-config/postgres/custom-config/sas-postgres-custom-config.yaml
- site-config/postgres/custom-config/sas-cpspostgres-custom-config.yaml
...
```

- In the transformers block, add references for sas-postgres-custom-config-transformer.yaml and sas-cpspostgres-custom-config-transformer.yaml (if needed). Here is an example that assumes the copied files were placed in **$deploy/site-config/postgres/custom-config**:

```
transformers:
...
- site-config/postgres/custom-config/sas-postgres-custom-config-transformer.yaml
- site-config/postgres/custom-config/sas-cpspostgres-custom-config-transformer.yaml
...
```

7 To use custom pg_hba settings:

a Copy the following files to your site-config directory:

- **$deploy/sas-bases/examples/postgres/custom-config/sas-pghba-custom-config.yaml**
- **$deploy/sas-bases/examples/postgres/custom-config/sas-pghba-custom-config-transformer.yaml**
- If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspghba-custom-config.yaml**
- If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspghba-custom-config-transformer.yaml**

b Using the comments in the files, revise the copied files with the appropriate values for the internal instance of PostgreSQL that it represents.

c In the base kustomization.yaml file, add references to the modified files.

- In the resources block, add references for sas-pghba-custom-config.yaml and sas-cpspghba-custom-config.yaml (if needed). Here is an example that assumes the files were placed in **$deploy/site-config/postgres/custom-config**:

```
resources:
...
- site-config/postgres/custom-config/sas-pghba-custom-config.yaml
- site-config/postgres/custom-config/sas-cpspghba-custom-config.yaml
...
```

- In the transformers block, add references for sas-pghba-custom-config-transformer.yaml and sas-cpspghba-custom-config-transformer.yaml (if needed). Here is an example that assumes the files were placed in **$deploy/site-config/postgres/custom-config**:

```
transformers:
...
- site-config/postgres/custom-config/sas-pghba-custom-config-transformer.yaml
- site-config/postgres/custom-config/sas-cpspghba-custom-config-transformer.yaml
...
```

8 To use custom Patroni settings:

   a Copy the following files to your site-config directory:

      - **$deploy/sas-bases/examples/postgres/custom-config/sas-patroni-custom-config.yaml**
      - **$deploy/sas-bases/examples/postgres/custom-config/sas-patroni-custom-config-transformer.yaml**
      - If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspatroni-custom-config.yaml**
      - If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspatroni-custom-config-transformer.yaml**

   b Using the comments in the files, revise the copied files with the appropriate values for the internal instance of PostgreSQL that it represents.

   c In the base kustomization.yaml file, add references to the modified files.

      - In the resources block of the base kustomization.yaml file, add a reference to the sas-patroni-custom-config.yaml file and the sas-cpspatroni-custom-config.yaml file (if needed). Here is an example that assumes the files were placed in **$deploy/site-config/postgres/custom-config**:

        ```
        resources:
        ...
        - site-config/postgres/custom-config/sas-patroni-custom-config.yaml
        - site-config/postgres/custom-config/sas-cpspatroni-custom-config.yaml
        ...
        ```

      - In the transformers block of the base kustomization.yaml file, add a reference to the sas-patroni-custom-config-transformer.yaml file and the sas-cpspatroni-custom-config-transformer.yaml file (if needed). Here is an example that assumes the files were placed in **$deploy/site-config/postgres/custom-config**:

        ```
        transformers:
        ...
        - site-config/postgres/custom-config/sas-patroni-custom-config-transformer.yamll
        - site-config/postgres/custom-config/sas-cpspatroni-custom-config-transformer.yaml
        ...
        ```

Changes for External Instances of PostgreSQL

Note: If you have an internal instance of PostgreSQL, skip this deployment note.

Version 2021.2 introduces a new method of connecting your SAS Viya platform deployment to an external instance of PostgreSQL. If you are updating to version 2021.2, you must remove artifacts of the old connection method and replace them with the new method.

1. For each instance of external PostgreSQL in your SAS Viya platform deployment, make a copy of the **`$deploy/examples/postgres/configure/external-postgres.yaml`** and place it in your site-config directory. Each copy should be uniquely named.
2. Using the comments in the file, revise each copy of external-postgres.yaml with the appropriate values for the external instance of PostgreSQL that it represents.
3. In the resources block of the base kustomization.yaml, add a reference for each external-postgres.yaml. Here is an example with two external instances of PostgreSQL:

   ```
   resources
   ...
   -site-config/postgres/configure/external-postgres-1.yaml
   -site-config/postgres/configure/external-postgres-2.yaml
   ...
   ```

4. Also, in the base kustomization.yaml file, remove the secretGenerator and configMapGenerator for the external instances of PostgreSQL. The secretGenerator begins with `- name: postgres-sas-user` and the configMapGenerator begins with `- name: sas-postgres-config`.

Changes for External Instances of Common Planning Service PostgreSQL

Version 2021.2 introduces a new method of Common Planning Service to an external instance of PostgreSQL. If you are updating to version 2021.2, you must remove artifacts of the old connection method and replace them with the new method.

1. If you are uncertain whether your deployment uses Common Planning Service PostgreSQL, examine the transformers block in the base kustomization.yaml file. If both of the following two lines are included in that block, continue with these instructions. If both lines are not included in the block, or only one of them is in the block, skip the rest of this deployment note.
   - `- sas-bases/overlays/sas-planning/sas-planning-transformer.yaml`
   - `- sas-bases/overlays/external-postgres/external-postgres-transformer.yaml`
2. Copy the **`$deploy/sas-bases/examples/postgres/configure/external-cpspostgres.yaml`** file to the site-config subdirectory, such as**`$deploy/site-config/postgres/configure/external-cpspostgres.yaml`**.
3. Adjust the values in the copied file as necessary for your external PostgreSQL instance.
4. Update the base kustomization.yaml file by adding the location of the copied file to the resources block of that file. Here is an example:

   ```
   resources:
   ```

```
...
- site-config/postgres/configure/external-cpspostgres.yaml
...
```

5 Remove the associated secretGenerators from the base kustomization.yaml file. Here are the names of the secretGenerators to be removed:

- `- name: postgres-sas-user`
- `- name: cpspostgres-sas-user`

6 Remove the configMapGenerator with the name `- name: sas-postgres-config` from the base kustomization.yaml file.

Change in Storage Class Configuration for SAS Programming Environment

The method by which you assign the storage class for runtime storage for processes such as SAS Compute server, SAS/CONNECT server, and SAS Batch server has been modified. If you used a YAML file to assign the storage class rather than use the default emptyDir, you must make revisions to your deployment.

Note: For more information about assigning storage classes for the SAS Programming Environment, see the README at **`$deploy/sas-bases/examples/sas-programming-environment/storage/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/sas_programming_environment_storage_tasks.htm`** (for HTML format).

1 Search for the following line in the transformers block of the base kustomization.yaml file.

```
- site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml
```

If that line is present, continue with these steps. Otherwise, there are no required changes and you should skip the rest of this deployment note.

2 Copy the **`$deploy/sas-bases/examples/sas-programming-environment/storage/change-viya-volume-storage-class.yaml`** file to the site-config directory, overwriting the existing file.

3 In the copied file, replace the `{{ VOLUME-STORAGE-CLASS }}` variable with the volume definition of the storage location you want to use.

4 In the base kustomization.yaml file, remove the `- site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml` from the transformers block.

5 Also in the base kustomization.yaml file, add the following content to the patches block:

```
patches:
- path: site-config/change-viya-volume-storage-class.yaml
  target:
    kind: PodTemplate
    labelSelector: "sas.com/template-intent=sas-launcher"
```

Revise Reference to sitedefault.yaml in the kustomization.yaml File

Version 2021.2 includes changes in how the SAS Viya platform works with the sitedefault file. Prior to version 2021.2, users were directed to create an entry in the configMapGenerator block of their base kustomization.yaml if their deployments use a sitedefault.yaml file. Beginning in version 2021.2, SAS recommends that the reference be moved to the secretGenerator block of the base kustomization.yaml file for security purposes. See "Add a sitedefault File to Your Deployment" in *SAS Viya Platform: Deployment Guide* for details of the new location.

Revise PersistentVolumeClaim annotationSelector

Due to changes in the software for version 2021.2, the patches block of the base kustomization.yaml file must be modified. In the annotationSelector line, add `sas-pyconfig`:

```
patches:
- path: site-config/storageclass.yaml
  target:
    kind: PersistentVolumeClaim
    annotationSelector: sas.com/component-name in (sas-backup-job,sas-
data-quality-services,sas-commonfiles,sas-cas-operator,sas-pyconfig)
```

To see the code sample in context, go to "Create the File" in *SAS Viya Platform: Deployment Guide*.

Changes to sas-model-publish Overlays

Release 2021.2 contains changes to the overlay files for Git and Kaniko, which are located in **`$deploy/sas-bases/overlays/sas-model-publish`**. To ensure that your update is successful, evaluate the transformers block of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**).

- If that block contains `sas-bases/overlays/sas-model-publish/kaniko/kaniko-transformer.yaml`, you should have SAS Model Manager in your SAS Viya platform deployment. If SAS Model Manager is not in your deployment, delete the `sas-bases/overlays/sas-model-publish/kaniko/kaniko-transformer.yaml` line.
- If that block contains `sas-bases/overlays/sas-model-publish/git/git-transformer.yaml`, you should have SAS Intelligent Decisioning or SAS Model Manager in your SAS Viya platform deployment. If neither of those offerings is in your deployment, delete the `sas-bases/overlays/sas-model-publish/git/git-transformer.yaml` line.

Enable Host Launch for the CAS Server

Note: If you meet any of the following conditions in your deployment, you must enable host launch:

- Your deployment uses Kerberos.
- Your deployment enables SSSD and disables CASCLOUDNATIVE.
- Your deployment uses the CASHostAccountRequired custom group or sets the CASALLHOSTACCOUNTS environment variable.

By default, CAS cannot launch sessions under a user's host identity. All sessions run under the cas service account instead. CAS can be configured to allow for host identity launches by including a patch transformer in the kustomization.yaml file. To enable host launch for CAS, see the "Enable Host Launch in the CAS Server" section of the README file located at `$deploy/sas-bases/examples/cas/configure/README.md` (for Markdown format) or at `$deploy/sas-bases/docs/configuration_settings_for_cas.htm` (for HTML format).

Changes to Compute Server Example Files

Changes have been made to the Compute Server example files to ensure that the Compute Server and Batch Server have the same mounts. Perform the following steps if you are updating to version 2021.2:

1. Look in the transformers block of the base kustomization.yaml file for references to the following example files:
   - compute-server-add-host-mount.yaml
   - compute-server-add-nfs-mount.yaml

   If neither of those files is included in the transformers block, then there are no further actions to take.
2. If either or both of the files is included in the transformer block, go to the location described in the transformer block and remove the file or files.
3. Copy the new version of the file or files from `$deploy/sas-bases/examples/sas-compute-server/configure` and paste them in the location from which you just deleted the old files.
4. Follow the instructions in the "Configuration Settings for Compute Server" README file located at `$deploy/sas-bases/examples/sas-compute-server/configure/README.md` (for Markdown format) or `$deploy/sas-bases/docs/configuration_settings_for_compute_server.htm` (for HTML format) to modify the new file or files for your deployment. You do not have to revise the base kustomization.yaml file since you are putting the new files in the same location that the old files were in.

Changes to Python Configuration

Beginning in version 2021.2, the python-transformer.yaml file contains changes that affect configuring Python volumes. If you configured Python in a previous version of the SAS Viya platform, you must modify the python-transformer.yaml file before updating to version 2021.2.

1. Search your base kustomization.yaml file for a reference to python-transformer.yaml in the transformers block. If the reference does not exist, no action is necessary.

2 Otherwise, delete the **$deploy/site-config/sas-open-source-config/python/python-transformer.yaml** file.

3 To create the new configuration, follow the steps in the README file at **$deploy/sas-bases/examples/sas-open-source-config/python/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_python_for_sas_viya.htm** (for HTML format).

Updates to cachelocator and cacheserver Services

Note: Because deleting the service might interfere with the completion of other deployment notes in the "Before Deployment Commands" section, perform the task in this note last. If your update involves deployment notes from other LTS releases (such as updating to the Long-Term Support 2022.09 release), re-order the deployment notes so that this task is the last one performed.

Because of changes to sas-cachelocator and sas-cacheserver, in order to deploy the new versions, you must delete the old versions when you update to release 2021.2.

```
kubectl -n name-of-namespace delete service sas-cachelocator sas-cacheserver
```

After Deployment Commands

Update CAS Servers

Version 2021.2 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although deleting the CAS server pods is required in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Delete sas-cacheserver-0

A new version of sas-cacheserver is included in version 2021.2. If you update to version 2021.2, the sas-cacheserver-1 pod might be blocked from reaching a READY state by sas-cacheserver-0. If sas-cacheserver-1 does not reach the READY state in an appropriate amount of time, you should delete sas-cacheserver-0.

After updating your software, run the following command to determine the status of the sas-cacheserver pods:

```
kubectl -n name-of-namespace get pods -l "app in (sas-cacheserver)"
```

The output should look like this:

```
NAMESPACE      NAME                                  READY   STATUS        RESTARTS   AGE
d10006         sas-cacheserver-0                     1/1     Running       0          5s
d10006         sas-cacheserver-1                     1/1     Running       0          5s
```

The READY value for the sas-cacheserver-1 pod should be `1/1` and the STATUS should be `RUNNING`. It might take a few moments for the READY state to reach `1/1` after the STATUS is `RUNNING`.

If the sas-cacheserver-1 pod does not reach the value of `1/1`, delete the sas-cacheserver-0 pod:

```
kubectl -n name-of-namespace delete pods sas-cacheserver-0
```

After the sas-cacheserver-0 pod is deleted and is automatically restarted by Kubernetes, the sas-cacheserver-1 pod finishes its initialization and reaches the READY state. Other pods that are unable to initialize might also reach the READY state after sas-cacheserver-0 is deleted and restarted.

Check Open Distro for Elasticsearch Cluster

Note: If your deployment does not contain Open Distro for Elasticsearch, skip these instructions.

Due to a timing issue, an update can result in an invalid StatefulSet manifest, which prevents the sas-opendistro-default-0 (or other master/data nodes) from coming up.

1. After updating the SAS Viya platform, ensure that the Open Distro for Elasticsearch cluster pods are running. If your deployment uses the default topology for Open Distro for Elasticsearch, use the following command:

   ```
   kubectl -n name-of-namespace get pods | grep sas-opendistro-default
   ```

   If you created a custom topology using the custom-topology.yaml example file, use the following command:

   ```
   kubectl -n name-of-namespace get pods | grep sas-opendistro-node-set
   ```

   In this command, *node-set* is a custom Elasticsearch node set defined in the example file. If you have more than one custom node set, the command should be performed for each node set.

2. Restart each pod that is in a `CrashLoopBackOff` state in the output from step 1. To restart an Elasticsearch pod in a default topology, delete it:

```
kubectl -n name-of-namespace delete pod sas-opendistro-default-pod-number
```

To restart an Elasticsearch pod in a custom topology:

```
kubectl -n name-of-namespace delete pod sas-opendistro-node-set-pod-number
```

3 Review the log of each sas-opendistro-default-*pod-number* pod (or each sas-opendistro-*node-set-pod-number* pod if you have custom topologies) for the following error.

```
org.elasticsearch.bootstrap.StartupException: java.lang.IllegalArgumentException:
unknown setting [node.roles] please check that any required plugins are
installed, or check the breaking changes documentation for removed settings
```

If the error is in the log, restart the pod using the commands described in step 2.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.2, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.2, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

SAS Risk Engine

Before Deployment Commands

Automatic Migration of Logic Objects Not Supported

The SAS Risk Engine user interface uses PostgreSQL to store the logic objects (for example, risk methods, model groups, and programs). The PostgreSQL schema was changed significantly in the 2021.2 release. Therefore, automatic migration from previous releases is not supported.

For the steps to migrate to release 2021.2, see Migrate the PostgreSQL Database.

SAS Information Catalog

After Deployment Commands

Rebuild Datasets Search Index

After an update, the structure of the datasets search index has been modified, requiring the index to be rebuilt. To rebuild the index, perform the steps described at "Respond to Notifications" in *SAS Information Catalog: Administrator's Guide*.

2021.1 (May 2021)

General Deployment

Before Deployment Commands

SAS Viya Platform Deployment Operator Changes

In version 2021.1, SAS has begun emphasizing using the SAS Viya Platform Deployment Operator to deploy your SAS Viya platform software. In addition to the organization changes in the documentation described in *What's New in SAS Viya Platform Operations*, SAS has also codified some of the best practices associated with deploying and using the operator. Existing deployments of the operator might already meet these guidelines, but if they do not, determine whether changes to the existing deployment should be made.

Here is a list of changes and links to where they are described in the deployment guide:

- Use the operator in only one mode, namespace or cluster-wide, in each cluster. If you select namespace mode, multiple operators can be used in the cluster. See "Overview of the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*,
- Use a unique directory for the files associated with the operator. This directory is referred to as `$operator-deploy` in the documentation. See "Retrieve the Files Required by the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide* and "Directories for the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*.
- Use a unique directory for the license files and certificates for all your deployments in a cluster. This directory is referred to as `$license` in the documentation. See "Retrieve Required Files" in *SAS Viya Platform: Deployment Guide* and "Directories for the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*.
- Name the SASDeployment custom resource for each SAS Viya platform deployment using the $deploy directory name as a prefix (for example, viya1-sasdeployment.yaml). See "Run the create sas-deployment-cr Command" in *SAS Viya Platform: Deployment Guide*.

Some Configuration Instances Disabled by Default

The SAS Administrator can use SAS Environment Manager to modify the configuration for the SAS Viya platform compute, SAS/CONNECT, SAS Batch, and CAS servers. Modifications to these servers involve updating configuration instances. Each server type has multiple instances that allow for modifications to configuration files, autoexec code, and start-up scripts that are used to launch the servers.

Prior to version 2021.1, these configuration instances were all enabled by default. With version 2021.1, configuration instances that allow start-up script modification are disabled by default.

In order to enable these configuration instances, a Kubernetes administrator must set SAS_ALLOW_ADMIN_SCRIPTS to true in the sas-shared-config configMap. The required steps are described in **`$deploy/sas-bases/overlays/sas-programming-environment/README.md`**.

For more information about the subset of configuration instances that are now disabled by default, see Edit Server Configuration Instances in *SAS Viya Platform Administration*.

Configure Open Distro for Elasticsearch

Version 2021.1 includes Open Distro for Elasticsearch, which is an Apache 2.0-licensed distribution of Elasticsearch enhanced with enterprise security. The SAS Viya platform includes Open Distro for Elasticsearch and uses its distributed search cluster in infrastructure and solution services.

If you are updating to version 2021.1, perform the following steps before the deployment commands in order to configure Open Distro for Elasticsearch.

Note: The example kustomization.yaml file, located at "Initial kustomization.yaml File" in *SAS Viya Platform: Deployment Guide*, includes these customizations.

1. Add the following line to the resources block of the base kustomization.yaml file:

```
resources:
...
- sas-bases/overlays/internal-elasticsearch
...
```

2. Add the following line to the transformers block of the base kustomization.yaml file:

```
transformers:
...
- sas-bases/overlays/internal-elasticsearch/internal-elasticsearch-transformer.yaml
...
```

3. Add a reference to the sysctl-transformers.yaml file to the transformers block of the base kustomization.yaml file. This transformer must be included after any TLS transformers and before the sas-bases/overlays/required/transformers.yaml transformer.

Note: The sysctl-transformers.yaml transformer uses a privileged container to set vm.max_map_count. If privileged containers are not allowed in your deployment, do not add this line. Instead, the Kubernetes administrator must set the `vm.max_map_count` property for stateful workload nodes manually.

Here is an example:

```
transformers:
...
- sas-bases/overlays/network/ingress/security/transformers/...
- sas-bases/overlays/internal-elasticsearch/sysctl-transformer.yaml
- sas-bases/overlays/required/transformers.yaml
...
```

Update annotationSelector in storageclass.yaml Reference

Before version 2021.1, the creation of a storageclass.yaml file and its addition to the example kustomization.yaml file was treated as optional. Beginning in version 2021.1, it is being treated as required. See "Specify PersistentVolumeClaims to Use ReadWriteMany StorageClass" in *SAS Viya Platform: Deployment Guide* for information about creating the storagelass.yaml file.

Additionally, the `patches` content that you are directed to add to the initial kustomization.yaml file requires a revision. The `annotationSelector` should have `sas-commonfiles` added to it. Here is an example:

```
patches:
- path: site-config/storageclass.yaml
  target:
    kind: PersistentVolumeClaim
    annotationSelector: sas.com/component-name in (sas-backup-job,sas-
data-quality-services,sas-commonfiles)
```

The example kustomization.yaml file at "Create the File" in *SAS Viya Platform: Deployment Guide* has been revised to reflect this change.

Python Configuration Changes

In version 2021.1, the files that are used to configure Python have been modified. If you included the transformer file in the base kustomization.yaml file, you must replace the earlier version of it and the Python kustomization.yaml with updated versions to ensure a successful update.

Perform the following steps:

1. Search the transformers block of your base kustomization.yaml file for "python-transformer.yaml". If you do not find a reference to the transformer, no action is required.
2. Otherwise, delete the following files:
   - **`$deploy/site-config/sas-open-source-config/python/python-transformer.yaml`**

- **$deploy/site-config/sas-open-source-config/python/kustomization.yaml**

3 Follow the steps in the README file at **$deploy/sas-bases/examples/sas-open-source-config/python/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_python_for_sas_viya.htm** (for HTML format).

Delete RabbitMQ Service

Note: This deployment note applies only to manual deployments of the SAS Viya platform. If you use the SAS Viya Platform Deployment Operator to deploy and maintain your software, the procedures described in this deployment note are performed for you, and you should skip this deployment note.

A change has been added to RabbitMQ with version 2021.1 that requires the existing RabbitMQ service be deleted before an update.

If you are updating to version 2021.1, run the following command:

```
kubectl -n name-of-namespace delete service sas-rabbitmq-server
```

Delete sas-consul-server-internal Service

Note: This deployment note applies only to manual deployments of the SAS Viya platform. If you use the SAS Viya Platform Deployment Operator to deploy and maintain your software, the procedures described in this deployment note are performed for you, and you should skip this deployment note.

Updating to version 2021.1 of the SAS Viya platform requires deleting the sas-consul-server-internal service before performing the update.

To delete the service, run the following command:

```
kubectl -n name-of-namespace delete service sas-consul-server-internal
```

Delete Obsolete Crunchy Data Pods

Note: This deployment note applies only to manual deployments of the SAS Viya platform. If you use the SAS Viya Platform Deployment Operator to deploy and maintain your software, the procedures described in this deployment note are performed for you, and you should skip this deployment note.

Also, if you have an external instance of PostgreSQL, you should skip this deployment note.

The provider of the internal instance of PostgreSQL, Crunchy Data, has reorganized the pods in which the software is deployed. If you are updating to version 2021.1 of the SAS Viya platform and you have an internal instance of PostgreSQL, you must delete some pods before updating your software.

Before you perform the deployment commands for your update, run the following commands:

```
kubectl -n name-of-namespace delete job -l pg-task=backup-sas-crunchy-data-
postgres-pgdump
kubectl -n name-of-namespace delete job -l pg-task=restore-sas-crunchy-data-
postgres-pgdump
```

If there is no pod to delete, the command has the following output:

```
No resources found
```

This is an acceptable result.

For more information about the new organization of the Crunchy Data pods, see PostgreSQL and Crunchy Data.

Delete sas-commonfiles Job

Note: This deployment note applies to manual deployments of the SAS Viya platform only. If you use the SAS Viya Platform Deployment Operator to deploy and maintain your software, the procedures described in this deployment note are performed for you, and you should skip this deployment note.

The addition of the sas-commonfiles component requires that any sas-commonfiles Jobs from an earlier version of the SAS Viya platform be deleted before updating to version 2021.1. Run the following command to delete the Job:

```
kubectl -n name-of-namespace delete job -l app=sas-commonfiles
```

If a sas-commonfiles Job existed and was successfully deleted, the output of the command is

```
job.batch "sas-commonfiles" deleted
```

If there is no sas-commonfiles Job to be deleted, the output is

```
No resources found
```

Either output indicates success.

After Deployment Commands

Update CAS Servers

Version 2021.1 contains an update to CAS. However, until the old CAS servers are restarted, your deployment will continue to run the older version.

While deleting the CAS server pods is required to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, dropping all the in-memory data, and restarts new pods.
- The updated CAS servers might need to be re-loaded with data.
- Stopping and starting the CAS servers causes an outage that will be noticeable for the end-users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the old CAS pods which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Because of the length of the command and the margin of the page, this command appears as more than one line. The command should be entered as a single line.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Deployment Operator Environment Variables Deprecated

In version 2021.1, the SAS Deployment Operator has deprecated two of its environment variables, `MINIMUM_UPDATE_INTERVAL` and `SAS_UPDATE_CHECK_INTERVAL`. While these variables continue to work in version 2021.1, SAS recommends they be replaced with a new variable `AUTOUPDATE_SCHEDULE`. For more information about the new environment variable, see "Environment Variables for the Operator Pod" in *SAS Viya Platform: Deployment Guide*.

SAS/CONNECT Lockdown Default Changed

Using the LOCKDOWN System Option and the LOCKDOWN Statement, you can limit access to files and to specific features in a SAS/CONNECT server session. Prior to version 2021.1, LOCKDOWN was disabled by default. But with version 2021.1, LOCKDOWN has been enabled by default.

Note: For more information about the LOCKDOWN system option, see "LOCKDOWN System Option" in *SAS Viya Platform: Programming Run-Time Servers*.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.1, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.1, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

SAS Model Publish Service

Before Deployment Commands

Deployment Assets Directory Structure Change

In version 2021.1, the SAS Model Publish Service has added content for Git, causing the directory structure to change. As a result, if you are updating to version 2021.1 and you previously configured sas-model-publish for Kaniko, you must modify your original .yaml files.

1. Delete the following files from your directory structure:
   - **$deploy/site-config/examples/sas-model-publish/podtemplate.yaml**
   - **$deploy/site-config/examples/sas-model-publish/storage.yaml**
2. Remove the following entries from your base kustomization.yaml file:
   - In the resources block, remove `- site-config/sas-model-publish`
   - In the transformers block, remove `- sas-bases/overlays/sas-model-publish/transformers.yaml`
3. Follow the instructions in the new README file, located at **$deploy/sas-bases/examples/sas-model-publish/kaniko/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configure_kaniko_for_sas_model_publish_service.htm** (for HTML format). The README file contains the necessary steps to copy and modify the new .yaml files and add the appropriate references to the base kustomization.yaml file.

3

Stable Releases

2023.08 (August 2023)

General Deployment

Before Deployment Commands

SAS Workflow Orchestrator Included in Deployment by Default

SAS Workload Orchestrator is now deployed and enabled, by default, as part of the SAS Viya platform deployment. A license is no longer required. If you are updating a deployment that does not include SAS Workflow Orchestrator, after the update, the deployment will include it and it will be enabled. However, SAS Workload Orchestrator requires at least one node labeled for the compute workload class. Therefore, unless you choose to disable SAS Workload Orchestrator, at least one node of your cluster must be labeled for the compute workload class before you update your deployment.

Note: If your deployment includes SAS Workload Orchestrator but it has been disabled, after the update, it will remain disabled.

For the instructions to disable SAS Workload Orchestrator, see the README file located at `$deploy/sas-bases/examples/sas-workload-orchestrator/enable-disable/README.md` (for Markdown format) or at `$deploy/sas-bases/docs/disabling_and_enabling_the_sas_workload_orchestrator_service.htm` (for HTML format).

Revised SAS Configurator for Open Source Transformer

Note: Although this deployment note appears to repeat a note that has been included in cadences 2023.03 through 2023.07, it is actually for a different change in the same file. Therefore, even if you have already performed this task in a cadence other than 2023.08, you must perform it again when updating to 2023.08. If you are updating over multiple cadences between 2023.03 and 2023.08, you only should perform the task once.

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised and should be replaced during your update.

1. In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-configuration2.yaml`. For example, if the file is located at **$deploy/site-config/sas-pyconfig/change-configuration.yaml**, change the file name to **$deploy/site-config/sas-pyconfig/change-configuration2.yaml**.
3. Copy the file in **$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.
4. Using the change-configuration2.yaml file as a guide, revise the change-configuration.yaml file so that it includes the same values you used in the earlier version of the file.
5. After your revisions are complete, delete the change-configuration2.yaml file.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or with sas-orchestration deploy and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2023.08 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2023.08, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.08, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2023.07 (July 2023)

General Deployment

Before Deployment Commands

Revised SAS Configurator for Open Source Transformer

Note: This deployment note is repeated from version 2023.03 because of the timing of its arrival. The repetition is to ensure that users are aware of the change. If you have performed the task described in this note already, do not repeat it.

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised and should be replaced during your update.

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-configuration2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-pyconfig/change-configuration.yaml`**, change the file name to **`$deploy/site-config/sas-pyconfig/change-configuration2.yaml`**.
3. Copy the file in **`$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml`** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.
4. Follow the instructions in the SAS Configurator for Open Source README to revise the change-configuration.yaml file to enable support for R, Python, or both. The README file is located at **`$deploy/sas-bases/examples/sas-pyconfig/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/sas_configurator_for_open_source_options.htm`** (for HTML format). You do not need to add content to the base kustomization.yaml file because the new file replaces the old one in the same location.

   Alternatively, you can compare the change-configuration.yaml and change-configuration2.yaml files. Use the change-configuration2.yaml file as a guide for

the revisions to be made to the change-configuration.yaml file in order to match the configuration settings used in your existing deployment.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or with sas-orchestration deploy and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2023.07 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2023.07, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.07, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2023.06 (June 2023)

General Deployment

Before Deployment Commands

Revised SAS Configurator for Open Source Transformer

Note: This deployment note is repeated from version 2023.03 because of the timing of its arrival. The repetition is to ensure that users are aware of the change. If you have performed the task described in this note already, do not repeat it.

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised and should be replaced during your update.

1. In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-configuration2.yaml`. For example, if the file is located at **$deploy/site-config/sas-pyconfig/change-configuration.yaml**, change the file name to **$deploy/site-config/sas-pyconfig/change-configuration2.yaml**.
3. Copy the file in **$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.
4. Follow the instructions in the SAS Configurator for Open Source README to revise the change-configuration.yaml file to enable support for R, Python, or both. The README file is located at **$deploy/sas-bases/examples/sas-pyconfig/README.md** (for Markdown format) or **$deploy/sas-bases/docs/sas_configurator_for_open_source_options.htm** (for HTML format). You do not need to add content to the base kustomization.yaml file because the new file replaces the old one in the same location.

   Alternatively, you can compare the change-configuration.yaml and change-configuration2.yaml files. Use the change-configuration2.yaml file as a guide for the revisions to be made to the change-configuration.yaml file in order to match the configuration settings used in your existing deployment.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or with sas-orchestration deploy and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2023.06 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.

- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2023.06, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.06, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

SAS Risk Modeling

Before Deployment Commands

Update Transformer File

In 2023.06, one of the SAS Risk Modeling transformer files has been modified, requiring you to change a key and the associated value. Prior to 2023.06, the **$deploy/site-config/sas-risk-cirrus-rm/resources/rm_transform.yaml** file contained an `rm_builder_repo_token` variable for which you used the encoded version of the personal access token. In 2023.06, that variable has been replaced with `SAS_RISK_CIRRUS_SOLUTION_BUILDER_REPO_TOKEN`.

To replace the variable:

1. Find the sas-risk-cirrus-rm-secret:

   ```
   kubectl -n name-of-namespace get secrets |grep sas-risk-cirrus-rm-secret
   ```

   The output should look something like this:

   ```
   sas-risk-cirrus-rm-secret-97725ckf4b        Opaque        4      12d
   ```

   In this example, the name of the secret is `sas-risk-cirrus-rm-secret-97725ckf4b`.

2. Determine if `rm_builder_repo_token` was used in the existing deployment:

   ```
   kubectl -n name-of-namespace describe secret secret-name
   ```

   The *secret-name* is the output you got in step 1. If a key is returned, then you must modify the existing deployment. If a key is not returned, no further steps are necessary.

3. Go to **$deploy/site-config/sas-risk-cirrus-rm/resources/rm_transform.yaml**.

4. Edit the file by replacing `rm_builder_repo_token` with `SAS_RISK_CIRRUS_SOLUTION_BUILDER_REPO_TOKEN`. The `value` line that follows should not be revised unless your Git personal access token has changed. If it has changed, use the new value in this line.

5. Delete the old secret:

   ```
   kubectl -n name-of-namespace patch secret secret-name --type=json -
   p='[{"op": "remove", "path": "/data/rm_builder_repo_token"}]'
   ```

6. Verify the secret has been removed:

   ```
   kubectl -n name-of-namespace describe secret secret-name
   ```

   If `rm_builder_repo_token` is not included in the output, then the secret has been successfully deleted.

7 Save and close the **$deploy/site-config/sas-risk-cirrus-rm/resources/rm_transform.yaml** file.

2023.05 (May 2023)

General Deployment

Before Deployment Commands

Revised SAS Configurator for Open Source Transformer

Note: This deployment note is repeated from version 2023.03 because of the timing of its arrival. The repetition is to ensure that users are aware of the change. If you have performed the task described in this note already, do not repeat it.

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised and should be replaced during your update.

1 In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.

2 The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-configuration2.yaml`. For example, if the file is located at **$deploy/site-config/sas-pyconfig/change-configuration.yaml**, change the file name to **$deploy/site-config/sas-pyconfig/change-configuration2.yaml**.

3 Copy the file in **$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.

4 Follow the instructions in the SAS Configurator for Open Source README to revise the change-configuration.yaml file to enable support for R, Python, or both. The README file is located at **$deploy/sas-bases/examples/sas-pyconfig/README.md** (for Markdown format) or **$deploy/sas-bases/docs/sas_configurator_for_open_source_options.htm** (for HTML format). You do not need to add content to the base kustomization.yaml file because the new file replaces the old one in the same location.

Alternatively, you can compare the change-configuration.yaml and change-configuration2.yaml files. Use the change-configuration2.yaml file as a guide for the revisions to be made to the change-configuration.yaml file in order to match the configuration settings used in your existing deployment.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or with sas-orchestration deploy and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2023.05 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2023.05, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.05, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2023.04 (April 2023)

General Deployment

Before Deployment Commands

Modified change-viya-volume-storage-class.yaml File

Note: This deployment note is repeated from version 2023.02 because of the timing of its arrival. The repetition is to ensure that users are aware of the change. If you have performed the task described in this note already, do not repeat it.

The change-viya-volume-storage-class.yaml file has been modified. If you used the change-viya-volume-storage-class.yaml file, you must replace the existing version with the new version in your deployment.

1 In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "change-viya-volume-storage-class.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.

2 The reference in the base kustomization.yaml file is the location of the yaml file. Go to that location and change the file name to `change-viya-volume-storage-class2.yaml`. For example, if the file is located at **site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml**, change the file name to **site-config/sas-programming-environment/storage/change-viya-volume-storage-class2.yaml**.

3 Copy **sas-bases/examples/sas-programming-environment/storage/change-viya-volume-storage-class.yaml** to the directory in step 2. You should now have a change-viya-volume-storage-class.yaml file and a change-viya-volume-storage-class2.yaml file in the same directory.

4 Using the change-viya-volume-storage-class2.yaml file as a guide, revise the change-viya-volume-storage-class file so that it includes the same values you used in the earlier version of the file. The change requires you to replace `{{ VOLUME-STORAGE-CLASS }}` with the value you used in the original file. For example, assume change-viya-volume-storage-class2.yaml looks like this:

```
kind: PodTemplate
  metadata:
    name: change-viya-volume-storage-class
  template:
   spec:
    volumes:
    - $patch: delete
      name: viya
    - name: viya
      nfs:
        server: myserver.mycompany.com
        path: /path/to/my/location
```

The highlighted code shows the value that was used to replace `{{ VOLUME-STORAGE-CLASS }}` in the original file.Therefore, in change-viya-volume-storage-class.yaml, you would replace `{{ VOLUME-STORAGE-CLASS }}` with the same content.

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      name: viya
      nfs:
        server: myserver.mycompany.com
        path: /path/to/my/location
```

5 After your revisions are complete, delete the change-viya-volume-storage-class2.yaml file.

6 In the base kustomization.yaml file, remove the reference to the change-viya-volume-storage-class.yaml file from the patches block. If the change-viya-

volume-storage-class.yaml file was saved to **$deploy/site-config**, the reference would look like this:

```
patches:
- path: site-config/change-viya-volume-storage-class.yaml
  target:
    kind: PodTemplate
    labelSelector: "sas.com/template-intent=sas-launcher"
```

Everything that is highlighted in the example should be removed.

Note: If there are no other patches in the block after the change-viya-volume-storage-class.yaml patch is removed, remove the entire patches block.

7 In the transformers block of the base kustomization.yaml file, add a reference to the change-viya-volume-storage-class.yaml file. The reference must be before the `sas-bases/overlays/required/transformers.yaml` entry. If the change-viya-volume-storage-class.yaml file was saved to **$deploy/site-config**, the reference would look like this:

```
transformers:
...
- site-config/change-viya-volume-storage-class.yaml
- sas-bases/overlays/required/transformers.yaml
```

Revised SAS Configurator for Open Source Transformer

Note: This deployment note is repeated from version 2023.03 because of the timing of its arrival. The repetition is to ensure that users are aware of the change. If you have performed the task described in this note already, do not repeat it.

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised and should be replaced during your update.

1 In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.

2 The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-configuration2.yaml`. For example, if the file is located at **$deploy/site-config/sas-pyconfig/change-configuration.yaml**, change the file name to **$deploy/site-config/sas-pyconfig/change-configuration2.yaml**.

3 Copy the file in **$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.

4 Follow the instructions in the SAS Configurator for Open Source README to revise the change-configuration.yaml file to enable support for R, Python, or both. The README file is located at **$deploy/sas-bases/examples/sas-pyconfig/README.md** (for Markdown format) or **$deploy/sas-bases/docs/**

`sas_configurator_for_open_source_options.htm` (for HTML format). You do not need to add content to the base kustomization.yaml file because the new file replaces the old one in the same location.

Alternatively, you can compare the change-configuration.yaml and change-configuration2.yaml files. Use the change-configuration2.yaml file as a guide for the revisions to be made to the change-configuration.yaml file in order to match the configuration settings used in your existing deployment.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or with sas-orchestration deploy and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2023.04 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Business Orchestration Services

Before Deployment Commands

Avoid Updating to Version 2023.04

A security patch has been made available for SAS Business Orchestration Services at version 2023.03 for both the Stable and Long-Term Support cadences. The same fix has been included in Stable 2023.05, but not in Stable 2023.04. Therefore, if you are at Stable 2023.03 and update to Stable 2023.04, you could potentially lose the fixes included in the patch for the duration of your Stable 2023.04 deployment. If you are at Stable 2023.03 and want to update your deployment, SAS recommends that you skip Stable 2023.04 and update directly to Stable 2023.05.

SAS Dynamic Actuarial Modeling

Before Deployment Commands

Update Transformer File

In 2023.04, one of the SAS Dynamic Actuarial Modeling transformer files has been modified, requiring you to change a key and the associated value. Prior to 2023.04, the **`$deploy/site-config/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml`** file contained a `pcpricing_builder_repo_token` variable for which you used the encoded version of the personal access token. In 2023.04, that variable has been replaced with `SAS_RISK_CIRRUS_SOLUTION_BUILDER_REPO_TOKEN`. Additionally, the same transformer contained a `sas_risk_cirrus_code_lib_repo_token` variable for which you used the encoded version of the personal access token for the code library. In 2023.04, that variable has been replaced with `SAS_RISK_CIRRUS_CODE_LIB_REPO_TOKEN`.

To replace the variables:

1 Find the sas-risk-cirrus-pcpricing-secret:

```
kubectl -n name-of-namespace get secrets |grep sas-risk-cirrus-pcpricing-
secret
```

The output should look something like this:

```
sas-risk-cirrus-pcpricing-secret-97725ckf4b        Opaque        4      12d
```

In this example, the name of the secret is `sas-risk-cirrus-pcpricing-secret-97725ckf4b`.

2 Determine if `pcpricing_builder_repo_token` or `sas_risk_cirrus_code_lib_repo_token` was used in the existing deployment:

```
kubectl -n name-of-namespace describe secret secret-name
```

The *secret-name* is the output you got in step 1. If a key is returned, then you must modify the existing deployment. If a key is not returned, no further steps are necessary.

3 Go to **`$deploy/site-config/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml`**.

4 Edit the pcpricing_transform.yaml file.

- If `pcpricing_builder_repo_token` is present in the file, replace it with `SAS_RISK_CIRRUS_SOLUTION_BUILDER_REPO_TOKEN`. The `value` line that follows should not be revised unless your Git personal access token has changed. If it has changed, use the new value in this line.
- If `sas_risk_cirrus_code_lib_repo_token` is present in the file, replace it with `SAS_RISK_CIRRUS_CODE_LIB_REPO_TOKEN`. The `value` line that follows should not be revised unless your Git personal access token for the code library has changed. If it has changed, use the new value in this line.

5 Delete the old secret.

- If only `pcpricing_builder_repo_token` was replaced, use this command:

```
kubectl -n name-of-namespace patch secret secret-name --type=json -
p='[{"op": "remove", "path": "/data/pcpricing_builder_repo_token"}]'
```

- If only `sas_risk_cirrus_code_lib_repo_token` was replaced, use this command:

```
kubectl -n name-of-namespace patch secret secret-name --type=json -
p='[{"op": "remove", "path": "/data/
sas_risk_cirrus_code_lib_repo_token"}]'
```

- If both were replaced, use this command:

```
kubectl -n name-of-namespace patch secret secret-name --type=json -
p='[{"op": "remove", "path": "/data/pcpricing_builder_repo_token"},
{"op": "remove", "path": "/data/sas_risk_cirrus_code_lib_repo_token"}]'
```

6 Verify the secret has been removed:

```
kubectl -n name-of-namespace describe secret secret-name
```

If the `pcpricing_builder_repo_token` and `sas_risk_cirrus_code_lib_repo_token` are not included in the output, then the secret has been successfully deleted.

7 Save and close the **`$deploy/site-config/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml`** file.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2023.04, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.04, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2023.03 (March 2023)

General Deployment

Before Deployment Commands

SAS Viya Platform Deployment Operator Requires an Update

Note: If you do not use the SAS Viya Platform Deployment Operator to deploy your software, skip this note. Additionally, if you do not use a mirror registry in your deployment, skip this note.

The SAS Viya Platform Deployment Operator has been updated with a mandatory change. If you use the operator to deploy your software, you must update the

operator by redeploying it. Before you redeploy it, consider the following best practices:

- Save the existing $deploy-operator directory to a different location so that you can refer to it.
- When editing the site-config/transformers.yaml file, the values for the name and the namespace must match existing values.
- If cluster-wide mode is used, edit the kustomization.yaml and uncomment the specified line as was done previously.

When those tasks are completed, redeploy the operator using the instructions at "Deploy the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*.

Create Pod Templates for Multi-Tenancy

Beginning in version 2023.03, each tenant in a multi-tenant deployment requires SAS Programming Environment pod templates. Each tenant in an existing multi-tenant deployment must have the pod templates added to it. The steps to add the pod templates are described in the "Create Kubernetes Resources" section of the README file located at **`$deploy/sas-bases/examples/sas-tenant-job/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/onboard_or_offboard_tenants.htm`** (for HTML format).

Modified change-viya-volume-storage-class.yaml File

Note: This deployment note is repeated from version 2023.02 because of the timing of its arrival. The repetition is to ensure that users are aware of the change. If you have performed the task described in this note already, do not repeat it.

The change-viya-volume-storage-class.yaml file has been modified. If you used the change-viya-volume-storage-class.yaml file, you must replace the existing version with the new version in your deployment.

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "change-viya-volume-storage-class.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the yaml file. Go to that location and change the file name to `change-viya-volume-storage-class2.yaml`. For example, if the file is located at **`site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml`**, change the file name to **`site-config/sas-programming-environment/storage/change-viya-volume-storage-class2.yaml`**.
3. Copy **`sas-bases/examples/sas-programming-environment/storage/change-viya-volume-storage-class.yaml`** to the directory in step 2. You should now have a change-viya-volume-storage-class.yaml file and a change-viya-volume-storage-class2.yaml file in the same directory.
4. Using the change-viya-volume-storage-class2.yaml file as a guide, revise the change-viya-volume-storage-class file so that it includes the same values you

used in the earlier version of the file. The change requires you to replace {{ VOLUME-STORAGE-CLASS }} with the value you used in the original file. For example, assume change-viya-volume-storage-class2.yaml looks like this:

```
kind: PodTemplate
  metadata:
    name: change-viya-volume-storage-class
  template:
   spec:
    volumes:
    - $patch: delete
      name: viya
    - name: viya
      nfs:
        server: myserver.mycompany.com
        path: /path/to/my/location
```

The highlighted code shows the value that was used to replace {{ VOLUME-STORAGE-CLASS }} in the original file.Therefore, in change-viya-volume-storage-class.yaml, you would replace {{ VOLUME-STORAGE-CLASS }} with the same content.

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      name: viya
      nfs:
        server: myserver.mycompany.com
        path: /path/to/my/location
```

5 After your revisions are complete, delete the change-viya-volume-storage-class2.yaml file.

6 In the base kustomization.yaml file, remove the reference to the change-viya-volume-storage-class.yaml file from the patches block. If the change-viya-volume-storage-class.yaml file was saved to **$deploy/site-config**, the reference would look like this:

```
patches:
- path: site-config/change-viya-volume-storage-class.yaml
  target:
    kind: PodTemplate
    labelSelector: "sas.com/template-intent=sas-launcher"
```

Everything that is highlighted in the example should be removed.

Note: If there are no other patches in the block after the change-viya-volume-storage-class.yaml patch is removed, remove the entire patches block.

7 In the transformers block of the base kustomization.yaml file, add a reference to the change-viya-volume-storage-class.yaml file. The reference must be before the sas-bases/overlays/required/transformers.yaml entry. If the change-viya-volume-storage-class.yaml file was saved to **$deploy/site-config**, the reference would look like this:

```
transformers:
...
- site-config/change-viya-volume-storage-class.yaml
```

```
- sas-bases/overlays/required/transformers.yaml
```

Revised SAS Configurator for Open Source Transformer

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised and should be replaced during your update.

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-configuration2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-pyconfig/change-configuration.yaml`**, change the file name to **`$deploy/site-config/sas-pyconfig/change-configuration2.yaml`**.
3. Copy the file in **`$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml`** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.
4. Follow the instructions in the SAS Configurator for Open Source README to revise the change-configuration.yaml file to enable support for R, Python, or both. The README file is located at **`$deploy/sas-bases/examples/sas-pyconfig/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/sas_configurator_for_open_source_options.htm`** (for HTML format). You do not need to add content to the base kustomization.yaml file because the new file replaces the old one in the same location.

   Alternatively, you can compare the change-configuration.yaml and change-configuration2.yaml files. Use the change-configuration2.yaml file as a guide for the revisions to be made to the change-configuration.yaml file in order to match the configuration settings used in your existing deployment.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or with sas-orchestration deploy and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see “Enable CAS Auto-Restart After Updates” in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2023.03 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2023.03, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.03, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

SAS Model Risk Management

Before Deployment Commands

Update Transformer File

In 2023.03, one of the SAS Model Risk Management transformer files has been modified, requiring you to change a key and the associated value. Prior to 2023.03, the **$deploy/site-config/sas-risk-cirrus-mrm/resources/mrm_transfer.yaml** file contained an `mrm_builder_repo_token` variable for which you used the encoded version of the personal access token. In 2023.03, that variable has been replaced with `SAS_RISK_CIRRUS_SOLUTION_BUILDER_REPO_TOKEN`.

To replace the variable:

1. Find the sas-risk-cirrus-mrm-secret:

   ```
   kubectl -n name-of-namespace get secrets |grep sas-risk-cirrus-mrm-secret
   ```

   The output should look something like this:

   ```
   sas-risk-cirrus-mrm-secret-97725ckf4b        Opaque        4      12d
   ```

   In this example, the name of the secret is `sas-risk-cirrus-mrm-secret-97725ckf4b`.

2. Determine if `mrm_builder_repo_token` was used in the existing deployment:

   ```
   kubectl -n name-of-namespace describe secret secret-name
   ```

   The *secret-name* is the output you got in step 1. If a key is returned, then you must modify the existing deployment. If a key is not returned, no further steps are necessary.

3. Go to **$deploy/site-config/sas-risk-cirrus-mrm/resources/mrm_transfer.yaml**.

4. Edit the file by replacing `mrm_builder_repo_token` with `SAS_RISK_CIRRUS_SOLUTION_BUILDER_REPO_TOKEN`. The `value` line that follows should not be revised unless your Git personal access token has changed. If it has changed, use the new value in this line.

5. Delete the old secret:

   ```
   kubectl -n name-of-namespace patch secret secret-name --type=json -
   p='[{"op": "remove", "path": "/data/mrm_builder_repo_token"}]'
   ```

6. Verify the secret has been removed:

   ```
   kubectl -n name-of-namespace describe secret secret-name
   ```

   If `mrm_builder_repo_token` is not included in the output, then the secret has been successfully deleted.

7 Save and close the **$deploy/site-config/sas-risk-cirrus-mrm/resources/mrm_transfer.yaml** file.

2023.02 (February 2023)

General Deployment

Before Deployment Commands

Modified change-viya-volume-storage-class.yaml File

The change-viya-volume-storage-class.yaml file was modified with version 2023.02. If you used the change-viya-volume-storage-class.yaml file, you must replace the existing version with the new version in your deployment.

1 In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "change-viya-volume-storage-class.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.

2 The reference in the base kustomization.yaml file is the location of the yaml file. Go to that location and change the file name to `change-viya-volume-storage-class2.yaml`. For example, if the file is located at **site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml**, change the file name to **site-config/sas-programming-environment/storage/change-viya-volume-storage-class2.yaml**.

3 Copy **sas-bases/examples/sas-programming-environment/storage/change-viya-volume-storage-class.yaml** to the directory in step 2. You should now have a change-viya-volume-storage-class.yaml file and a change-viya-volume-storage-class2.yaml file in the same directory.

4 Using the change-viya-volume-storage-class2.yaml file as a guide, revise the change-viya-volume-storage-class file so that it includes the same values you used in the earlier version of the file. The change requires you to replace `{{ VOLUME-STORAGE-CLASS }}` with the value you used in the original file. For example, assume change-viya-volume-storage-class2.yaml looks like this:

```
kind: PodTemplate
  metadata:
    name: change-viya-volume-storage-class
  template:
   spec:
    volumes:
```

```
    - $patch: delete
      name: viya
    - name: viya
      nfs:
        server: myserver.mycompany.com
        path: /path/to/my/location
```

The highlighted code shows the value that was used to replace {{ VOLUME-STORAGE-CLASS }} in the original file.Therefore, in change-viya-volume-storage-class.yaml, you would replace {{ VOLUME-STORAGE-CLASS }} with the same content.

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      name: viya
      nfs:
        server: myserver.mycompany.com
        path: /path/to/my/location
```

5 After your revisions are complete, delete the change-viya-volume-storage-class2.yaml file.

6 In the base kustomization.yaml file, remove the reference to the change-viya-volume-storage-class.yaml file from the patches block. If the change-viya-volume-storage-class.yaml file was saved to **$deploy/site-config**, the reference would look like this:

```
patches:
- path: site-config/change-viya-volume-storage-class.yaml
  target:
    kind: PodTemplate
    labelSelector: "sas.com/template-intent=sas-launcher"
```

Everything that is highlighted in the example should be removed.

Note: If there are no other patches in the block after the change-viya-volume-storage-class.yaml patch is removed, remove the entire patches block.

7 In the transformers block of the base kustomization.yaml file, add a reference to the change-viya-volume-storage-class.yaml file. The reference must be before the sas-bases/overlays/required/transformers.yaml entry. If the change-viya-volume-storage-class.yaml file was saved to **$deploy/site-config**, the reference would look like this:

```
transformers:
...
- site-config/change-viya-volume-storage-class.yaml
- sas-bases/overlays/required/transformers.yaml
```

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or with sas-orchestration deploy and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2023.02 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2023.02, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.02, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2023.01 (January 2023)

General Deployment

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or with sas-orchestration deploy and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2023.01 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Remove cachelocator and cacheserver PVCs

Note: This deployment note is repeated from version 2022.11 because of the timing of its arrival. The repetition is to ensure that users are aware of the change. If you have performed the task described in this note already, do not repeat it.

With the 2022.11 release of the SAS Viya platform, Redis fully replaces Apache Geode and SAS Cache Server. However, two PVCs from the previous caching software should be removed manually from your deployment.

Perform the following commands to remove the PVCs:

```
kubectl -n name-of-namespace delete pvc -l app.kubernetes.io/name=sas-cacheserver
kubectl -n name-of-namespace delete pvc -l app.kubernetes.io/name=sas-cachelocator
```

SAS Event Stream Processing

Before Deployment Commands

Required PersistentVolumeClaim with RWX Permissions

Note: This deployment note is repeated from version 2022.11 because of the timing of its arrival. The repetition is to ensure that users of SAS Event Stream Processing are aware of the change in the requirements. If you have performed the task described in this note already, do not repeat it.

Beginning with this version of the SAS Viya platform, SAS Event Stream Processing requires a PVC that is set to RWX. In a multi-tenant deployment, one persistent volume is required per tenant. SAS Event Stream Processing Studio must be configured to use the PV. For more information, see "Managing Persistent Volumes" in *SAS Event Stream Processing: Using SAS Event Stream Processing in a Kubernetes Environment*.

Delete espserver before the Update

When you update to version 2023.01, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.01, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

SAS Risk Engine

Before Deployment Commands

Delete sas-risk-samples Job

Before updating to version 2023.01, delete the sas-risk-samples job:

```
kubectl -n name-of-namespace delete job sas-risk-samples
```

If there is no job to delete, the command responds with a message that the job was not found. This is an acceptable result.

Add CDS PostgreSQL to SAS Risk Engine

In version 2023.01, an instance of CDS PostgreSQL is being added to SAS Risk Engine. A CDS PostgreSQL instance is used because the character of the data it will contain is hierarchically different than the data generally stored in the platform PostgreSQL database. Adding the CDS PostgreSQL instance requires modifying YAML files in the deployment. For the specific steps required, see the "Common Data Store (CDS) PostgreSQL" section of the README file located at **`$deploy/sas-bases/examples/postgres/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/configure_postgresql.htm`** (for HTML format).

Note: For the requirements for an instance of CDS PostgreSQL, see “PostgreSQL Server Requirements” in *System Requirements for the SAS Viya Platform*.

2022.12 (December 2022)

General Deployment

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or with sas-orchestration deploy and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2022.12 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Remove cachelocator and cacheserver PVCs

Note: This deployment note is repeated from version 2022.11 because of the timing of its arrival. The repetition is to ensure that users are aware of the change. If you have performed the task described in this note already, do not repeat it.

With the 2022.11 release of the SAS Viya platform, Redis fully replaces Apache Geode and SAS Cache Server. However, two PVCs from the previous caching software should be removed manually from your deployment.

Perform the following commands to remove the PVCs:

```
kubectl -n name-of-namespace delete pvc -l app.kubernetes.io/name=sas-cacheserver
kubectl -n name-of-namespace delete pvc -l app.kubernetes.io/name=sas-cachelocator
```

SAS Event Stream Processing

Before Deployment Commands

Required PersistentVolumeClaim with RWX Permissions

Note: This deployment note is repeated from version 2022.11 because of the timing of its arrival. The repetition is to ensure that users of SAS Event Stream Processing are aware of the change in the requirements. If you have performed the task described in this note already, do not repeat it.

Beginning with this version of the SAS Viya platform, SAS Event Stream Processing requires a PVC that is set to RWX. In a multi-tenant deployment, one persistent volume is required per tenant. SAS Event Stream Processing Studio must be configured to use the PV. For more information, see “Managing Persistent Volumes” in *SAS Event Stream Processing: Using SAS Event Stream Processing in a Kubernetes Environment*.

Delete espserver before the Update

When you update to version 2022.12, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2022.12, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:
`kubectl -n name-of-namespace get espserver`

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

SAS Inventory Optimization

Before Deployment Commands

Update Warning

Do not update SAS Inventory Optimization to version 2022.12. The latest update contains changes that will break your software deployment. Instead, you should uninstall any deployment you have from before version 2022.12 and perform a new deployment using version 2022.12.

2022.11 (November 2022)

General Deployment

Before Deployment Commands

SAS Configurator for Open Source Configuration Change

SAS Configurator for Open Source has revised one of the configuration files used to deploy. As a result, you must replace your old configuration file in order to update successfully.

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-`

`configuration2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-pyconfig/change-configuration.yaml`**, change the file name to **`$deploy/site-config/sas-pyconfig/change-configuration2.yaml`**.

3 Copy the file in **`$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml`** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.

4 Using the change-configuration2.yaml file as a guide, revise the change-configuration.yaml file so that it includes the same values you used in the earlier version of the file.

5 After your revisions are complete, delete the change-configuration2.yaml file.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2022.11 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Remove cachelocator and cacheserver PVCs

With the 2022.11 release of the SAS Viya platform, Redis fully replaces Apache Geode and SAS Cache Server. However, two PVCs from the previous caching software should be removed manually from your deployment.

Perform the following commands to remove the PVCs:

```
kubectl -n name-of-namespace delete pvc -l app.kubernetes.io/name=sas-cacheserver
kubectl -n name-of-namespace delete pvc -l app.kubernetes.io/name=sas-cachelocator
```

SAS Dynamic Actuarial Modeling

Before Deployment Commands

Add CDS PostgreSQL to SAS Dynamic Actuarial Modeling

In version 2022.11, an instance of CDS PostgreSQL is being added to SAS Dynamic Actuarial Modeling. A CDS PostgreSQL instance is used because the character of the data it will contain is hierarchically different than the data generally stored in the platform PostgreSQL database. Adding the CDS PostgreSQL instance requires modifying YAML files in the deployment. For the specific steps required, see the "Common Data Store (CDS) PostgreSQL" section of the README file located at **`$deploy/sas-bases/examples/postgres/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/configure_postgresql.htm`** (for HTML format).

Note: For the requirements for an instance of CDS PostgreSQL, see "PostgreSQL Server Requirements" in *System Requirements for the SAS Viya Platform*.

SAS Event Stream Processing

Before Deployment Commands

Required PersistentVolumeClaim with RWX Permissions

Beginning with this version of the SAS Viya platform, SAS Event Stream Processing requires a PVC that is set to RWX. In a multi-tenant deployment, one persistent volume is required per tenant. SAS Event Stream Processing Studio must be configured to use the PV. For more information, see "Managing Persistent Volumes" in *SAS Event Stream Processing: Using SAS Event Stream Processing in a Kubernetes Environment*.

Delete espserver before the Update

When you update to version 2022.11, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2022.11, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

SAS Model Risk Management

Before Deployment Commands

Add CDS PostgreSQL to SAS Model Risk Management

In version 2022.11, an instance of CDS PostgreSQL is being added to SAS Model Risk Management. A CDS PostgreSQL instance is used because the character of

the data it will contain is hierarchically different than the data generally stored in the platform PostgreSQL database. Adding the CDS PostgreSQL instance requires modifying YAML files in the deployment. For the specific steps required, see the "Common Data Store (CDS) PostgreSQL" section of the README file located at `$deploy/sas-bases/examples/postgres/README.md` (for Markdown format) or `$deploy/sas-bases/docs/configure_postgresql.htm` (for HTML format).

Note: For the requirements for an instance of CDS PostgreSQL, see "PostgreSQL Server Requirements" in *System Requirements for the SAS Viya Platform*.

2022.10 (October 2022)

General Deployment

Before Deployment Commands

General Update Considerations

SAS has updated the base image for patches required in the 2022.09 Long-Term Support and 2022.11 Stable versions. If you are planning to switch from Long-Term Support 2022.1 to a Stable cadence, you can update to Long-Term Support 2022.09 and then update to a Stable version in Standard Support.

Support for Updates to Red Hat OpenShift Deployments with Crunchy Data 4

As of 17 January 2023, customers with internal instances of Crunchy Data 4 PostgreSQL on Red Hat OpenShift can update to a Stable version in Standard Support, which includes Crunchy Data 5. If you intend to update, ensure that you have downloaded deployment assets for a Stable version in Standard Support before starting the update process.

Update to Crunchy Data 5 for Internal Instances of PostgreSQL

The version of PostgreSQL that SAS uses, Crunchy Data, is upgrading from 4 to 5. Crunchy Data 5 is more declarative in nature than the previous version of Crunchy Data. It releases a new Custom Resource Definition (CRD) named postgresclusters.postgres-operator.crunchydata.com, and it uses fewer containers.

Because of the magnitude of the changes of Crunchy Data 5, internal instances of PostgreSQL must go through a disruptive migration process to the new version.

1 If your deployment is on Red Hat OpenShift, ensure that the SAS Viya Platform Deployment Operator is at version 1.83.2 or higher.

   a Determine the version of the SAS Viya Platform Deployment Operator that you are using:

   ```
   kubectl -n name-of-namespace get deployments/sas-deployment-operator -o
   jsonpath='{.metadata.annotations.sas\.com/version}'
   ```

   If you are using the operator in cluster-wide mode, be sure to use the name of the deployment operator namespace for *name-of-namespace*.

   If the output of the command is 1.83.2 or higher, your version of the operator does not need to be updated. Skip to step 2.

   b Update the SAS Viya Platform Deployment Operator using the instructions at "Update the Deployment Operator" in *SAS Viya Platform Operations: Updating Software*.

2 Perform a search of your base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the strings "postgres", "crunchy", and "pgo-client". Delete any line that includes any of those strings unless the line also contains "site-config".

   In the resources block, delete `- site-config/configure-postgres/internal/pgo-client` despite it not meeting the conditions in the first paragraph. Crunchy Data 5 does not support pgo-client. It was used to schedule pgBackRest backups, but with Crunchy Data 5 backups are performed automatically. For more information about backups, see the "Configuration Settings for PostgreSQL Backups" README located at **`$deploy/sas-bases/examples/crunchydata/backups/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/configuration_settings_for_postgresql_backups.htm`** (for HTML format).

3 Create storage transformers to record and protect the state of your existing storage. Storage settings are usually immutable but the declarative nature of Kubernetes can still result in changes if values are not explicitly declared. To prevent unwanted changes, create a storage transformer for each PostgreSQL cluster in your deployment with the storage settings from the cluster.

   IMPORTANT Do not attempt to change the settings while upgrading to Crunchy Data 5. After the upgrade, refer to the comments in the crunchy-storage-transfer.yaml file to determine which settings can be changed and which cannot.

   a Remove any references to old storage transformers (such as `postgres-storage-transformer`) from the transformers block in your base kustomization.yaml.

   b Determine the PostgreSQL clusters in your deployment. Every deployment will have at least one PostgreSQL cluster, sas-crunchy-data-postgres, but some may have a second, sas-crunchy-data-cdspostgres. Run the following command to check:

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

The output of this command should replace *crunchy-4-cluster-name* in step 3c. If your deployment includes the sas-crunchy-data-cdspostgres cluster, use that name in your second pass through step 3c.

c Follow the steps in the "Configuration Settings for PostgreSQL Storage" README located at **$deploy/sas-bases/examples/crunchydata/storage/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configuration_settings_for_postgresql_storage.htm** (for HTML format), to create a new storage transformer file. In the file, specify the variables as described below.

Note: In the bulleted list below, use the value that is returned by the commands in the storage transformer. Do not replace the returned value with what you feel would be a more appropriate value.

- Find the size of the cluster's PostgreSQL PVC and use the returned value in place of `{{ POSTGRES-STORAGE-SIZE }}` in the file.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name -o
  jsonpath='{.spec.resources.requests.storage}'
  ```

- Find the size of the cluster's pgBackRest PVC and use the returned value in place of `{{ BACKREST-STORAGE-SIZE }}` in the file.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name-pgbr-repo
  -o jsonpath='{.spec.resources.requests.storage}'
  ```

  Note: In the command, the name of the Crunchy Data 4 cluster has `-pgbr-repo` appended to it.

- Find the access mode for the cluster's PostgreSQL PVC and use the returned value in place of `{{ POSTGRES-ACCESS-MODE }}` in the file. Be sure to remove any brackets or quotation marks in the returned value.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name -o
  jsonpath='{.spec.accessModes}'
  ```

- Find the access mode for the cluster's pgBackRest PVC and use the returned value in place of `{{ BACKREST-ACCESS-MODE }}` in the file. Be sure to remove any brackets or quotation marks in the returned value.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name-pgbr-repo
  -o jsonpath='{.spec.accessModes}'
  ```

  Note: In the command, the name of the Crunchy Data 4 cluster has `-pgbr-repo` appended to it.

- Find the storage class for the cluster's PostgreSQL PVC and use the returned value in place of `{{ POSTGRES-STORAGE-CLASS }}` in the file.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name -o
  jsonpath='{.spec.storageClassName}'
  ```

- Find the storage class for the cluster's pgBackRest PVC and use the returned value in place of `{{ BACKREST-STORAGE-CLASS }}` in the file.

  ```
  kubectl -n name-of-namespace get pvc crunchy-4-cluster-name-pgbr-repo
  -o jsonpath='{.spec.storageClassName}'
  ```

Note: In the command, the name of the Crunchy Data 4 cluster has `-pgbr-repo` appended to it.

- Replace each instance of `{{ CLUSTER-NAME }}` in the file with the name of the Crunchy Data 5 cluster. For Platform PostgreSQL, use `sas-crunchy-platform-postgres`. For CDS PostgreSQL, use `sas-crunchy-cds-postgres`.

d If your deployment includes the sas-crunchy-data-cdspostgres cluster, create a second storage transformer for CDS PostgreSQL and repeat step 3c for that transformer.

e Include a reference to the newly created storage transformer file or files in the transformer block of the base kustomization.yaml file. For example, if you named the storage transformer platform-postgres-crunchy-storage-transformer.yaml, then the kustomization.yaml file would look like this:

```
transformers:
...
- site-config/crunchydata/platform-postgres-crunchy-storage-transformer.yaml
```

4 If your deployment is managed by the SAS Viya Platform Deployment Operator, skip this step.

Prepare your PostgreSQL database or databases for migration by using the steps described in the README file located at **`$deploy/sas-bases/examples/crunchydata/pgo4upgrade/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/upgrade_to_postgresql_operator_v5.htm`** (for HTML format). The README includes the direction to add temporary transformers for the migration from Crunchy Data 4 to 5. You will remove those temporary migration transformers later.

5 Modify the base kustomization.yaml file by adding the following lines to the resources and components blocks:

```
resources:
- sas-bases/overlays/postgres/platform-postgres
- sas-bases/overlays/crunchydata/postgres-operator
…
components:
- sas-bases/components/crunchydata/internal-platform-postgres
```

Note: Ensure that any line added to the components block is placed before any lines for TLS components.

If your deployment includes SAS Common Data Store (CDS PostgreSQL), add the following lines to your base kustomization.yaml file also:

```
resources:
- sas-bases/overlays/postgres/cds-postgres
…
components:
- sas-bases/components/crunchydata/internal-cds-postgres
```

6 Refer to the lines in the base kustomization.yaml file that contained "site-config" and which you did not delete from step 2.

a Use this list to find the information to replace the Crunchy Data 4 transformer with the analogous Crunchy Data 5 transformer.

- If you have a transformer named postgres-pods-resource-limits-settings-transformer.yaml, see the “Configuration Settings for PostgreSQL Pod Resources” README located at **$deploy/sas-bases/examples/crunchydata/pod-resources/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configuration_settings_for_postgresql_pod_resources.htm** (for HTML format).
- If you have a transformer named postgres-replicas-transformer.yaml, see the “Configuration Settings for PostgreSQL Replicas Count” README located at **$deploy/sas-bases/examples/crunchydata/replicas/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configuration_settings_for_postgresql_replicas_count.htm** (for HTML format).
- If you have a transformer that contains "postgres-custom-config", "pghba-custom-config", or "patroni-custom-config", see the “Configuration Settings for PostgreSQL Database Tuning” README located at **$deploy/sas-bases/examples/crunchydata/tuning/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configuration_settings_for_postgresql_database_tuning.htm** (for HTML format).

b Ensure that the Crunchy Data 5 version of the transformers that you create are included in the base kustomization.yaml file.

c Delete the Crunchy Data 4 version of the transformers that you are replacing from the base kustomization.yaml file. Remember that the single tuning and configuration Crunchy Data 5 transformer represents as many as six Crunchy Data 4 transformers, so be sure to delete them all.

7 Add transformers to the base kustomization.yaml file to migrate existing data to Crunchy Data 5.

Note: The transformers added in this step will be removed after the update. For the details, see “Finish Upgrading Crunchy Data” on page 110.

a Migrate the platform PostgreSQL data (`sas-crunchy-data-postgres`) by adding the following content to the transformer block:

```
transformers:
...
- sas-bases/examples/crunchydata/pgo4upgrade/crunchy-upgrade-platform-transformer.yaml
```

b If you have CDS PostgreSQL data (`sas-crunchy-data-cdspostgres`), migrate it by add the following content to the transformer block:

```
transformers:
...
- sas-bases/examples/crunchydata/pgo4upgrade/crunchy-upgrade-cdstransformer.
yaml
```

8 Execute the base kustomization.yaml file validation script to determine whether there are any errors in that file for Crunchy Data 5.

a Copy **$deploy/sas-bases/examples/crunchydata/pgo4upgrade/check_base_kustomization_for_crunchy4_5_uip.sh** to the top level of the $deploy directory.

b Modify the permissions of the script so that it can be run, and then run the script:

```
cd $deploy

chmod +x check_base_kustomization_for_crunchy4_5_uip.sh

./check_base_kustomization_for_crunchy4_5_uip.sh name-of-namespace
```

For *name-of-namespace*, use the namespace that contains the PostgreSQL cluster to be upgraded.

c If any messages are generated by the script, review and revise the base kustomization.yaml file and storage transformers according to the messages. Run the script again until no more messages are generated.

Update to Crunchy Data 5 for External Instances of PostgreSQL

The version of PostgreSQL that SAS uses, Crunchy Data, is upgrading from 4 to 5. As a result, the interface to connect to external instances of PostgreSQL is being upgraded as well, requiring deployments with external instances of PostgreSQL to make disruptive changes.

1 Perform a search of your base kustomization.yaml file ($deploy/kustomization.yaml) and delete any line that contains the strings "postgres" or "sql-proxy".

2 Follow the instructions in the “External PostgreSQL Configuration” section of the “Configure PostgreSQL” README file located at **$deploy/sas-bases/examples/postgres/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configure_postgresql.htm** (for HTML format).

Update the mirror.yaml File

Note: If you are not using a mirror registry, skip this note. If you are not performing a manual update, skip this note.

Changes in the software for version 2022.10 require new images for your deployment. To ensure you retrieve those images, you must update your mirror.yaml file. To update the mirror.yaml file, perform steps 1 and 2 of “Add a Mirror Registry to Your SAS Viya Platform Deployment” in *SAS Viya Platform: Deployment Guide*.

Changes in GID and UID Management for Identities Microservice

The update to version 2022.10 includes default assignments for new configuration properties for the Identities service. The new default settings might contradict how

your existing deployment is configured, resulting in users being unable to access content.

In your SAS Viya platform environment, if your end-users interact with the file system from the SAS Compute Server, CAS server host-launched sessions, or SAS/CONNECT sessions, the UID and GID that are provided by the identities service are crucial to the implementation of file system permissions. The Launcher Service launches pods using user attribute information, including the UID and sometimes the GID, from the identities service. The default option for the SAS/CONNECT Spawner and for host-launched sessions of CAS also uses user attribute information that is returned from the identities service. The changes to the configuration properties affect the way that the identities service obtains UID and GID information.

Before making the updated deployment available to your end users, review the new Identities settings at "Identities Service" in *SAS Viya Platform: Configuration Reference* and revise the configuration to best suit your needs. More details can be found in "Identity Management: Overview" in *SAS Viya Platform: Identity Management*.

CAS Configuration Change

Note: This deployment note appears to be a duplicate of the deployment note with the same name for version 2022.09. However, while the steps are similar, the file that is being referred to was modified again for version 2022.10. Therefore, if you have already completed an update to version 2022.09, these steps must be repeated to pick up the changes in that file at 2022.10. If you are updating through both the 2022.09 and 2022.10 versions at the same time, these steps should only be performed once.

In version 2022.10, a file that is used to configure CAS has been modified. If you included the transformer file in the base kustomization.yaml file, you must replace the earlier version of it with an updated version to ensure a successful update.

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "cas-enable-host.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `cas-enable-host2.yaml`. For example, if the file is located at **`site-config/cas/configure/cas-enable-host.yaml`**, change the file name to **`site-config/cas/configure/cas-enable-host2.yaml`**.
3. Copy **`sas-bases/examples/cas/configure/cas-enable-host.yaml`** to the directory in step 2. You should now have a cas-enable-host.yaml file and a cas-enable-host2.yaml file in the same directory.
4. Using the cas-enable-host2.yaml file as a guide, revise the cas-enable-host.yaml file so that it includes the same values you used in the earlier version of the file.
5. After your revisions are complete, delete the cas-enable-host2.yaml file.

CAS and seccomp Mode Changes

The yaml files that configure seccomp mode for CAS have been revised for version 2022.10. If you used these files in earlier versions, you must replace the original files with the new ones.

For the cas-disable-seccomp.yaml file:

1. In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "cas-disable-seccomp.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `cas-disable-seccomp2.yaml`. For example, if the file is located at **site-config/cas/configure/cas-disable-seccomp.yaml**, change the file name to **site-config/cas/configure/cas-disable-seccomp2.yaml**.
3. Copy **sas-bases/examples/cas/configure/cas-disable-seccomp.yaml** to the directory in step 2. You should now have a cas-disable-seccomp.yaml file and a cas-disable-seccomp2.yaml file in the same directory.
4. Using the cas-disable-seccomp2.yaml file as a guide, revise the cas-disable-seccomp.yaml file so that it includes the same values you used in the earlier version of the file.
5. After your revisions are complete, delete the cas-disable-seccomp2.yaml file.

For the cas-seccomp-profile.yaml file:

1. In your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "cas-seccomp-profile.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and delete the cas-seccomp-profile.yaml file.
3. Copy **sas-bases/examples/cas/configure/cas-seccomp-profile.yaml** to the directory in step 2.
4. Using the comments at the beginning of the cas-seccomp-profile.yaml file as a guide, revise the cas-seccomp-profile.yaml file.

Remove Redis Objects

Note: This deployment note is an exact copy of the deployment note with the same title for version 2022.09. Because it arrived after the release of version 2022.09, the note is being repeated here to ensure that all users updating their software will see it. If you performed this task for version 2022.09, do not repeat it for version 2022.10.

IMPORTANT If you are updating versions 2022.1.4 and 2022.10 at the same time, ensure that you perform the tasks described in this note before

you perform the tasks described in the 2022.1.4 deployment note, "Scale Down Consul Before Update" on page 117. The restrictions from the Note above still apply: if you performed this task for version 2022.09, do not repeat it for version 2022.10.

If you have had a deployment since at least version 2021.2.3, there is a possibility that there are Redis objects in your deployment that could prevent successful updates to version 2022.10. Use the following steps to determine if those objects are present and, if so, remove them.

1 Exec into the sas-consul-server-0 pod:

```
kubectl -n name-of-namespace exec -it sas-consul-server-0 -c sas-consul-
server -- /bin/bash
```

2 Run the following commands:

```
export CONSUL_HTTP_ADDR=https://localhost:8500
```

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv read config/ --recurse | grep
-i redis
```

If you have any results, continue with step 3. If you have no results, no further steps are necessary.

3 Delete the Consul key/value pairs that are associated with Redis with the following commands:

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/
sas.cache.redis/host
```

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/
sas.cache.redis/password
```

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/
sas.cache.redis/tls
```

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/
sas.cache.redis/username
```

Updates to cachelocator and cacheserver Services

Note: This deployment note is an exact copy of the deployment note with the same title for version 2022.09. The issue effects both the 2022.09 and 2022.10 versions, but if you are performing an update that includes both versions you should only perform this task once. The 2022.10 version of the deployment note exists for users whose initial SAS Viya platform deployment is 2022.09 and they are updating to 2022.10.

Because of changes to sas-cachelocator and sas-cacheserver, in order to deploy the new versions, you must scale their replicas to 0 when you update.

```
kubectl -n name-of-namespace scale statefulsets sas-cacheserver --replicas=0
kubectl -n name-of-namespace scale statefulsets sas-cachelocator --replicas=0
```

The update will set the replica count back to 2 if your software contains retail components.

SAS Configurator for Open Source Upgrade

Note: This deployment note is an exact copy of the deployment note with the same title for version 2022.1.3. Because it arrived after the release of version 2022.1.3, the note is being repeated here to ensure that all users updating their software will see it. If you performed this task for version 2022.1.3, do not repeat it for version 2022.10.

SAS Configurator for Open Source has upgraded to version 1.8. Use the following steps to ensure a successful upgrade.

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-configuration2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-pyconfig/change-configuration.yaml`**, change the file name to **`$deploy/site-config/sas-pyconfig/change-configuration2.yaml`**.
3. Copy the file in **`$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml`** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.
4. Follow the instructions in the SAS Configurator for Open Source README to revise the change-configuration.yaml file to enable support for R, Python, or both. The README file is located at **`$deploy/sas-bases/examples/sas-pyconfig/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/sas_configurator_for_open_source_options.htm`** (for HTML format).

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Platform Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2022.10 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Finish Upgrading Crunchy Data

Note: If your deployment uses an external instance of PostgreSQL, skip this note.

Note: The steps described in this deployment note can only be performed after using the deployment note "Update to Crunchy Data 5 for Internal Instances of PostgreSQL" on page 100. If you have not already, go to that note, perform the steps described there, and perform the deployment commands again.

When the Crunchy upgrade has completed, you should see pods that resemble the following:

```
$ kubectl -n name-of-namespace get pod | grep postgres
sas-crunchy-platform-postgres-00-dtbg-0                          5/5     Running   # Three data pods (1 primary + 2 replicas)
sas-crunchy-platform-postgres-00-gkmw-0                          5/5     Running
sas-crunchy-platform-postgres-00-glkn-0                          5/5     Running
sas-crunchy-platform-postgres-backup-95xj-6x74d                  0/1     Completed # Initial backup job
sas-crunchy-platform-postgres-move-pgbackrest-repo-dir-1-td6t5   0/1     Completed # Crunchy 4 to 5 upgrade job
sas-crunchy-platform-postgres-move-pgdata-dir-1-6sjtv            0/1     Completed # Crunchy 4 to 5 upgrade job
sas-crunchy-platform-postgres-repo-host-0                        2/2     Running   # pgBackRest backup/archive repo
sas-crunchy-platform-postgres-repo1-full-27896040-2v7rk          0/1     Completed
sas-crunchy5-postgres-operator-f98f56f96-q25vr                   1/1     Running   # Crunchy Operator
sas-crunchy5-postgres-operator-upgrade-6cd7c58569-7l6tv          1/1     Running   # Crunchy Operator Upgrader
```

Finish the upgrade to Crunchy Data 5:

1 Remove the temporary migration transformers you added to the base kustomization.yaml file ($deploy/kustomization.yaml) before you ran the deployment commands. In the transformers block, remove the following line:

```
transformers:
- sas-bases/examples/crunchydata/pgo4upgrade/crunchy-upgrade-platform-transformer.yaml
```

If your deployment includes SAS Common Data Store (CDS PostgreSQL), remove this line also:

```
transformers:
- sas-bases/examples/crunchydata/pgo4upgrade/crunchy-upgrade-cds-transformer.yaml
```

2 Patch the newly created PostgresCluster resources.

```
kubectl -n name-of-namespace patch postgrescluster sas-crunchy-platform-
postgres --type json -p '[{"op": "remove", "path": "/spec/dataSource"}]'
```

If your deployment includes SAS Common Data Store (CDS PostgreSQL), patch the CDS PostgreSQL resource as well.

```
kubectl -n name-of-namespace patch postgrescluster sas-crunchy-cds-postgres
--type json -p '[{"op": "remove", "path": "/spec/dataSource"}]'
```

3 Because names have been changed in Crunchy Data 5, if you have any tools with dependencies on the names of PostgreSQL deployment objects (such as pods, secrets, and configmaps), remove those dependencies.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2022.10, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2022.10, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

Configuration File Changes

Note: This deployment note is an exact copy of the deployment note with the same title for version 2022.1.3. Because it arrived after the release of version 2022.1.3, the note is being repeated here to ensure that all users updating their software will see it. If you performed this task for version 2022.1.3, do not repeat it for version 2022.10.

Files that are used to configure SAS Event Stream Processing have been modified. If you included the transformer files in the base kustomization.yaml file, you must replace the earlier versions of them with updated versions to ensure a successful update.

Note: In previous releases of SAS Event Stream Processing, the espconfig directory was named ESPConfig.

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "espconfig-env-variables.yaml". If that string is not present, go to step 6. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `espconfig-env-variables2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-esp-operator/espconfig/espconfig-env-variables.yaml`**, change the file name to **`$deploy/site-config/sas-esp-operator/espconfig/espconfig-env-variables2.yaml`**.
3. Copy **`$deploy/sas-bases/examples/sas-esp-operator/espconfig/espconfig-env-variables.yaml`** to the directory in step 2. You should now have an espconfig-env-variables.yaml file and an espconfig-env-variables2.yaml file in the same directory.
4. Using the espconfig-env-variables2.yaml file as a guide, revise the espconfig-env-variables.yaml file so that it includes the same values you used in the earlier version of the file.
5. After your revisions are complete, delete the espconfig-env-variables2.yaml file.
6. In your base kustomization.yaml file, search for the string "espconfig-properties.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 7.
7. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `espconfig-properties2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-esp-operator/espconfig/espconfig-properties.yaml`**, change the file name to **`$deploy/site-config/sas-esp-operator/espconfig/espconfig-properties2.yaml`**.
8. Copy **`$deploy/sas-bases/examples/sas-esp-operator/espconfig/espconfig-properties.yaml`** to the directory in step 2. You should now have an espconfig-properties.yaml file and an espconfig-properties2.yaml file in the same directory.

9 Using the espconfig-properties2.yaml file as a guide, revise the espconfig-properties.yaml file so that it includes the same values you used in the earlier version of the file.

10 After your revisions are complete, delete the espconfig-properties2.yaml file.

2022.09 (September 2022)

General Deployment

Before Deployment Commands

Updates to cachelocator and cacheserver Services

Because of changes to sas-cachelocator and sas-cacheserver, in order to deploy the new versions, you must scale their replicas to 0 when you update to release 2022.09.

```
kubectl -n name-of-namespace scale statefulsets sas-cacheserver --replicas=0
kubectl -n name-of-namespace scale statefulsets sas-cachelocator --replicas=0
```

The update will set the replica count back to 2.

Note: If your update includes versions after 2022.09, then the replica count is reset to 2 only if there are retail components in your deployment. Do not reset the replica count manually.

CAS Configuration Change

In version 2022.09, a file that is used to configure CAS has been modified. If you included the transformer file in the base kustomization.yaml file, you must replace the earlier version of it with an updated version to ensure a successful update.

1 In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "cas-enable-host.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.

2 The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `cas-enable-host2.yaml`. For example, if the file is located at **`site-config/cas/configure/cas-enable-host.yaml`**, change the file name to **`site-config/cas/configure/cas-enable-host2.yaml`**.

3 Copy **sas-bases/examples/cas/configure/cas-enable-host.yaml** to the directory in step 2. You should now have a cas-enable-host.yaml file and a cas-enable-host2.yaml file in the same directory.

4 Using the cas-enable-host2.yaml file as a guide, revise the cas-enable-host.yaml file so that it includes the same values you used in the earlier version of the file.

5 After your revisions are complete, delete the cas-enable-host2.yaml file.

Remove Redis Objects

IMPORTANT If you are updating versions 2022.1.4 and 2022.09 at the same time, ensure that you perform the tasks described in this note before you perform the tasks described in the 2022.1.4 deployment note, "Scale Down Consul Before Update" on page 117.

If you have had a deployment since at least version 2021.2.3, there is a possibility that there are Redis objects in your deployment that could prevent successful updates to version 2022.09. Use the following steps to determine if those objects are present and, if so, remove them.

1 Exec into the sas-consul-server-0 pod:

```
kubectl -n name-of-namespace exec -it sas-consul-server-0 -c sas-consul-server -- /bin/bash
```

2 Run the following commands:

```
export CONSUL_HTTP_ADDR=https://localhost:8500
```

Note: If your deployment uses Front-door TLS or no TLS, replace `https` with `http`.

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv read config/ --recurse | grep -i redis
```

If the command returns any results, continue with step 3. If you have no results, no further steps are necessary.

3 Delete the Consul key/value pairs that are associated with Redis with the following commands:

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/sas.cache.redis/host
```

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/sas.cache.redis/password
```

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/sas.cache.redis/tls
```

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv delete config/application/sas.cache.redis/username
```

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2022.09 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2022.09, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2022.09, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2022.1.4 (August 2022)

General Deployment

Before Deployment Commands

Update the SAS Viya Deployment Operator

IMPORTANT This deployment note is repeated from the 2022.1.3 cadence. Because the 2022.1.3 deployment note was published after the actual release of version 2022.1.3, the deployment note is being repeated for 2022.1.4 to ensure that all updating users see it. If you performed the tasks described in this note while updating to 2022.1.3, you do not have to repeat them for 2022.1.4.

Note: If you do not use the SAS Viya Deployment Operator to deploy your software, skip this note.

The SAS Viya Deployment Operator has been updated with a mandatory change. If you use the operator to deploy your software, you should determine if the operator should be updated and, if so, update the operator before performing any other deployment tasks.

1 Run the following command:

```
kubectl -n name-of-namespace get deployments/sas-deployment-operator -
o jsonpath='{.metadata.annotations.sas\.com/version}'
```

If you are using the operator in cluster-wide mode, be sure to use the name of the operator namespace for *name-of-namespace*.

If the output of the command is 1.71.3 or higher, your version of the operator does not need to be updated and you can skip the rest of this note. If the output is less than 1.71.3, continue with step 2.

2 Update the SAS Viya Deployment Operator by following the steps at "Deploy the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*. Here are some best practices to follow while updating the operator:

- Save the existing $deploy-operator directory to a different location so that you can refer to it.
- When editing the site-config/transformers.yaml file, the values for the name and the namespace must match existing values.
- If you are using the operator in cluster-wide mode, edit the kustomization.yaml file and uncomment the specified line as in your previous deployment.

Scale Down Consul Before Update

Note: If you are performing updates with the SAS Viya Deployment Operator, skip this note.

IMPORTANT If you are updating versions 2022.1.4 and 2022.09 at the same time, ensure that you perform the tasks described in this note after you perform the tasks described in the 2022.09 deployment note, "Remove Redis Objects" on page 114. Because that note is repeated for version 2022.10, if you choose to perform the tasks from that version, it still must be completed before performing the tasks described in this deployment note.

If you are updating your SAS Viya deployment manually, you must scale down Consul before performing the update. Use this command:

```
kubectl -n name-of-namespace scale statefulsets sas-consul-server --replicas=0
```

Replace YAML File for SAS Configurator for Open Source

IMPORTANT This deployment note is repeated from the 2022.1.3 cadence. Because the 2022.1.3 deployment note was published after the actual release of version 2022.1.3, the deployment note is being repeated for 2022.1.4 to ensure that all updating users see it. If you performed the tasks described in this note while updating to 2022.1.3, you do not have to repeat them for 2022.1.4.

The SAS Configurator for Open Source has been rewritten to run from a CronJob rather than a Kubernetes Job. As a result, if you have deloyed the SAS Configurator for Open Source, you must replace one of the YAML files that it uses for deployment.

1. Evaluate the transformers block of the base kustomization.yaml file. If the block contains a line that includes "change-limits.yaml", continue with the steps in this note. Otherwise, skip to the next deployment note.
2. Evaluate the content of the $deploy/site-config/sas-pyconfig/change-limits.yaml file. If the word "CronJob" appears in the file, then you have performed these steps in an earlier cadence and can skip the rest of this deployment note.
3. Change the name of the $deploy/site-config/sas-pyconfig/change-limits.yaml file to change-limits-2.yaml.
4. Copy `$deploy/sas-bases/examples/sas-pyconfig/change-limits.yaml` to `$deploy/site-config/sas-pyconfig/change-limits.yaml`.
5. Set the values for CPU and memory in the new change-limits.yaml file by using the values in the change-limits-2.yaml file.
6. Delete the change-limits-2.yaml file.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see “Enable CAS Auto-Restart After Updates” in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2022.1.4 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2022.1.4, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2022.1.4, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

Directory Name Change

The **$deploy/sas-bases/examples/sas-esp-operator/ESPconfig** has been renamed

`$deploy/sas-bases/examples/sas-esp-operator/espconfig`

If you have automated your deployment, you should determine if the directory name change requires changes to that automation.

2022.1.3 (July 2022)

General Deployment

Before Deployment Commands

Update the SAS Viya Deployment Operator

Note: If you do not use the SAS Viya Deployment Operator to deploy your software, skip this note.

The SAS Viya Deployment Operator has been updated with a mandatory change. If you use the operator to deploy your software, you should determine if it should be updated and, if so, update it before performing any other deployment tasks.

1. Perform the following command:

   ```
   kubectl -n name-of-namespace get deployments/sas-deployment-operator -
   o jsonpath='{.metadata.annotations.sas\.com/version}'
   ```

   If you are using the operator in cluster-wide mode, be sure to use the name of the operator namespace for *name-of-namespace*.

   If the output of the command is 1.71.3 or higher, your version of the operator does not need to be updated and you can skip the rest of this note. If the output is less than 1.71.3, continue with step 2.

2. Update the SAS Viya Deployment Operator by following the steps at "Deploy the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*. Here are some best practices to follow while updating the operator:

   - Save the existing $deploy-operator directory to a different location so that you can refer to it.

- When editing the site-config/transformers.yaml file, the values for the name and the namespace must match existing values.
- If you are using the operator in cluster-wide mode, edit the kustomization.yaml file and uncomment the specified line as in your previous deployment.

Kubernetes Server-Side Apply

SAS Viya now uses Kubernetes server-side apply (SSA) for select resources deployed to the Kubernetes cluster. SSA is a Kubernetes feature that provides improved field management for applied resources and addresses issues with increasingly large resource specifications.

The SAS Deployment Operator now uses SSA where appropriate. Also, the manual deployment steps for SAS Viya have been modified to enable SSA. For the revised manual deployment steps, see "Deployment Using Kubernetes Commands" in *SAS Viya Platform: Deployment Guide*, especially step 2.

For more information about SSA, see the official Kubernetes SSA documentation.

Configure Internal Instances of PostgreSQL

IMPORTANT If you are performing an update up to or past version 2022.10 (October 2022), you should skip this deployment note since the version of PostgreSQL is being upgraded with that update.

Note: If your deployment uses an external instance of PostgreSQL, skip this note.

Changes to the deployment of internal instances of PostgreSQL require revisions to the base kustomization.yaml file when you update to version 2022.1.3. Those changes also require evaluating the space available for backups and potentially cleaning up the space used for backups in order to prevent Disk Full errors. For the full set of tasks to be performed, see Perform Required PostgreSQL Server Maintenance.

Note: There is a post-deployment validation step associated with the changes described in this deployment note. See "Verify pgBackRest Backup Schedule" on page 123 for the details.

SAS Configurator for Open Source Upgrade

SAS Configurator for Open Source upgrades to version 1.8 with version 2022.1.3 of SAS Viya. Use the following steps to ensure a successful upgrade.

1 In your base kustomization.yaml file (`$deploy/kustomization.yaml`), search for the string "change-configuration.yaml". If that string is not present, no further steps are required. If the string is present, continue with step 2.

2 The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `change-configuration2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-pyconfig/change-configuration.yaml`**, change the file name to **`$deploy/site-config/sas-pyconfig/change-configuration2.yaml`**.

3 Copy the file in **`$deploy/sas-bases/examples/sas-pyconfig/change-configuration.yaml`** to the directory in step 2. You should now have a change-configuration.yaml file and a change-configuration2.yaml file in the same directory.

4 Follow the instructions in the SAS Configurator for Open Source README to revise the change-configuration.yaml file to enable support for R, Python, or both. The README file is located at **`$deploy/sas-bases/examples/sas-pyconfig/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/sas_configurator_for_open_source_options.htm`** (for HTML format).

Replace YAML File for SAS Configurator for Open Source

The SAS Configurator for Open Source has been rewritten to run from a CronJob rather than a Kubernetes Job. As a result, if you have deployed the SAS Configurator for Open Source, you must replace one of the YAML files that it uses for deployment.

1 Evaluate the transformers block of the base kustomization.yaml file. If the block contains a line that includes "change-limits.yaml", continue with the steps in this note. Otherwise, skip to the next deployment note.

2 Evaluate the content of the $deploy/site-config/sas-pyconfig/change-limits.yaml file. If the word "CronJob" appears in the file, then you have performed these steps in an earlier cadence and can skip the rest of this deployment note.

3 Change the name of the $deploy/site-config/sas-pyconfig/change-limits.yaml file to change-limits-2.yaml.

4 Copy **`$deploy/sas-bases/examples/sas-pyconfig/change-limits.yaml`** to **`$deploy/site-config/sas-pyconfig/change-limits.yaml`**.

5 Set the values for CPU and memory in the new change-limits.yaml file by using the values in the change-limits-2.yaml file.

6 Delete the change-limits-2.yaml file.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2022.1.3 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Verify pgBackRest Backup Schedule

IMPORTANT If you are performing an update up to or past version 2022.10 (October 2022), you should skip this deployment note since the version of PostgreSQL is being upgraded with that update.

Note: If your deployment uses an external instance of PostgreSQL, skip this note.

This deployment note is the verification step for the pgBackRest backup schedule created through the deployment note “Configure Internal Instances of PostgreSQL” on page 121. If you did not perform the tasks described in that deployment note, go back to it and perform those tasks before attempting the validation described here.

To verify the pgBackRest backup schedule that you created earlier, follow the steps described in Verify That the Backup Schedule Was Created.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2022.1.3, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2022.1.3, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

Configuration File Changes

In version 2022.1.3, files that are used to configure SAS Event Stream Processing have been modified. If you included the transformer files in the base kustomization.yaml file, you must replace the earlier versions of them with updated versions to ensure a successful update.

Note: In previous releases of SAS Event Stream Processing, the espconfig directory was named ESPConfig.

1. In your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "espconfig-env-variables.yaml". If that string is not present, go to step 6. If the string is present, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `espconfig-env-variables2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-esp-operator/espconfig/espconfig-env-variables.yaml`**, change the file name to **`$deploy/site-config/sas-esp-operator/espconfig/espconfig-env-variables2.yaml`**.

3 Copy **$deploy/sas-bases/examples/sas-esp-operator/espconfig/espconfig-env-variables.yaml** to the directory in step 2. You should now have an espconfig-env-variables.yaml file and an espconfig-env-variables2.yaml file in the same directory.

4 Using the espconfig-env-variables2.yaml file as a guide, revise the espconfig-env-variables.yaml file so that it includes the same values you used in the earlier version of the file.

5 After your revisions are complete, delete the espconfig-env-variables2.yaml file.

6 In your base kustomization.yaml file, search for the string "espconfig-properties.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 7.

7 The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `espconfig-properties2.yaml`. For example, if the file is located at **$deploy/site-config/sas-esp-operator/espconfig/espconfig-properties.yaml**, change the file name to **$deploy/site-config/sas-esp-operator/espconfig/espconfig-properties2.yaml**.

8 Copy **$deploy/sas-bases/examples/sas-esp-operator/espconfig/espconfig-properties.yaml** to the directory in step 2. You should now have an espconfig-properties.yaml file and an espconfig-properties2.yaml file in the same directory.

9 Using the espconfig-properties2.yaml file as a guide, revise the espconfig-properties.yaml file so that it includes the same values you used in the earlier version of the file.

10 After your revisions are complete, delete the espconfig-properties2.yaml file.

2022.1.2 (June 2022)

General Deployment

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml

transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2022.1.2 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2022.1.2, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2022.1.2, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2022.1.1 (May 2022)

General Deployment

Before Deployment Commands

Changes for cert-manager in kustomization.yaml

Note: If you are not using cert-manager as the certificate generator in your SAS Viya deployment, skip this note.

The YAML files used in selecting cert-manager as the certificate generator for your SAS Viya deployment have been revised.

If you are using cert-manager as your certificate generator:

1. In the base kustomization.yaml file, remove the `- site-config/security/cert-manager-provided-ingress-certificate.yaml` line from the transformers block.
2. In the base kustomization.yaml file, add a reference to the location of the new transformer for the cert-manager-provided-ingress-certificate in the transformers block:

```
transformers:
...

- sas-bases/overlays/cert-manager-provided-ingress-certificate/
ingress-annotation-transformer.yaml
```

Changes to Cross-Origin Resource Sharing Configuration

Changes to third-party software included in SAS Viya may force changes to how you have configured Cross-Origin Resource Sharing (CORS). The `sas.commons.web.security.cors` property can no longer use the asterisk (`*`) wildcard to replace a list of URIs. If you have used the asterisk wildcard, you must replace it with a full list and then restart your services.

For the details to find the `sas.commons.web.security.cors` property and revise it, see "Configure Cross-Origin Resource Sharing" in *SAS Viya Platform:*

Authentication. For the instructions to restart your services, see "Managing a SAS Viya Platform Deployment" in *SAS Viya Platform Operations: Servers and Services*.

Removal: $deploy/sas-bases/overlays/network/ingress Directory

The **$deploy/sas-bases/overlays/network/ingress** directory has been removed. Therefore, you must revise the reference to the directory in the base kustomization.yaml file if it has not been revised already.

In the resources block, replace `- sas-bases/overlays/network/ingress` with `- sas-bases/overlays/network/networking.k8s.io`:

```
resources:
- sas-bases/base
- sas-bases/overlays/cert-manager-issuer
- sas-bases/overlays/network/networking.k8s.io
```

The example kustomization.yaml file located at "Create the File" in *SAS Viya Platform: Deployment Guide* reflects this revision.

Same-Site Transformer Change

With the removal of the **$deploy/sas-bases/overlays/network/ingress** directory (see "Removal: $deploy/sas-bases/overlays/network/ingress Directory" on page 128), the content of the same-site transformer from previous versions of SAS Viya has been moved to a different file and location.

To determine if your deployment is affected and what actions to take:

1. Examine the base kustomization.yaml file (**$deploy/kustomization.yaml**). In the transformers block, look for the following line:

   ```
   - sas-bases/overlays/network/ingress/security/transformers/sas-ingress-cookie-samesite-transformer.yaml
   ```

   If the line is present, continue with these steps. Otherwise, skip the rest of this note.

2. Remove the `- sas-bases/overlays/network/ingress/security/transformers/sas-ingress-cookie-samesite-transformer.yaml` line.

3. In the components block of the base kustomization.yaml file, add the following line.

   ```
   components:
   ...
   - sas-bases/components/security/web/samesite-none
   ```

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2022.1.1 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Required PostgreSQL Configuration

IMPORTANT If you are performing an update up to or past version 2022.10 (October 2022), you should skip this deployment note since the version of PostgreSQL is being upgraded with that update.

Note: If your deployment uses an external instance of PostgreSQL, skip this note.

Every update to the PostgreSQL database generates WAL (Write-Ahead-Log) data. WAL data grows much faster than the database, filling up the disk and causing a Disk Full Error if it is not regularly cleaned up. If you have not done so already, use the instructions at Perform Required PostgreSQL Server Maintenance set up a scheduled pgBackRest backup with a retention policy to clean up old backups and unnecessary WAL data.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2022.1.1, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2022.1.1, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.2.6 (April 2022)

General Deployment

Before Deployment Commands

Default Certificate Generator Change

Beginning in version 2021.2.6, SAS Viya uses openssl as the default certificate generator. The openssl certificate generator is proprietary SAS software that uses the OpenSSL open-source project. SAS recommends using the openssl certificate

generator because it is provided to meet the needs of the SAS Viya software and is tested for compatibility with each release of SAS Viya.

Note: For more information about certificate generators, see "Certificate Generators" in *SAS Viya Platform Encryption: Data in Motion*. To determine which certificate generator your deployment uses, see "Determine Which Certificate Generator Is Being Used" in "Manage Certificates and Secrets" in *SAS Viya Platform Encryption: Data in Motion*.

If you are updating to 2021.2.6 and had been using cert-manager as your certificate generator but want to use openssl as your certificate generator going forward, perform the following steps:

1. In the resources block of your base kustomization.yaml file, remove the `- sas-bases/overlays/cert-manager-issuer` line.
2. In the transformers block of your base kustomization.yaml file, remove the `- site-config/security/cert-manager-provided-ingress-certificate.yaml` line.
3. If your deployment does not use a customer-provided ingress certificate:
   a. Copy the **`$deploy/sas-bases/examples/security/openssl-generated-ingress-certificate.yaml`** file to **`$deploy/site-config/security/openssl-generated-ingress-certificate.yaml`**.
   b. Add a reference to the copied file to the resources block of the base kustomization.yaml file. Here is an example:

```
resources:
...
- site-config/security/openssl-generated-ingress-certificate.yaml
...
```

4. To complete the conversion to openssl and disassociate the deployment from cert-manager so that cert-manager can safely be removed from your deployment, see "Remove Cert-manager from a SAS Viya Platform Deployment" in *SAS Viya Platform Encryption: Data in Motion*.

To override the default and continue using cert-manager as the certificate generator for your deployment, see the "Using the cert-manager Certificate Generator" section of the security README file located at **`$deploy/sas-bases/examples/security/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/configure_network_security_and_encryption_using_sas_security_certificate_framework.htm`** (for HTML format).

Connect Workload Class Changes

SAS/CONNECT provides the ability for clients to launch SAS compute processes within a local or remote SAS environment. The SAS/CONNECT Spawner is a SAS Viya service that launches these processes on behalf of SAS/CONNECT clients. The processes can be launched in their own pods (referred to as "dynamically launched pods") or in the SAS/CONNECT Spawner pod.

Beginning in version 2021.2.6, SAS/CONNECT Spawner is deployed in the stateless work class by default. The connect workload class is only required if you

are not using dynamically launched pods. If you choose not to use dynamically launched pods, you must add a transformer to your base kustomization.yaml file.

Note: For more information about dynamically launched pods, see "Workload Classes" in *SAS Viya Platform: Deployment Guide*.

To add the use-connect-workload-class.yaml transformer, revise the transformers block of the base kustomization.yaml file (`$deploy/kustomization.yaml`) to include the path to the transformer:

```
transformers:
...
- sas-bases/overlays/sas-connect-spawner/enable-spawned-servers/use-
connect-workload-class.yaml
```

If you are using dynamically launched pods, you will see the connect-spawner being scheduled onto stateless or stateful nodes. The connect node will no longer be used by the connect-spawner. You should consider removing connect workload class nodes from your cluster infrastructure.

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2021.2.6 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Visual Analytics

Before Deployment Commands

Reduce Required Memory for SAS Conversation Designer

The SAS Visual Analytics offering contains SAS Conversation Designer and the SAS Natural Language Understanding API, which require at least 4 GB of memory to run. If you are not using SAS Conversation Designer and the SAS Natural Language Understanding API, beginning in version 2021.2.6, you can reduce their memory allocation by approximately 3.5 GB. The instructions for reducing the memory allocation are located at the README file at **`$deploy/sas-bases/overlays/sas-natural-language-understanding/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/reduce_memory_resources_for_sas_natural_language_understanding.htm`** (for HTML format).

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.2.6, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.2.6, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.2.5 (March 2022)

General Deployment

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Note: If you have enabled state transfers for your CAS servers, skip this deployment note. For more information about state transfers for CAS servers, see "Enable State Transfer for CAS Servers" in *SAS Viya Platform: Deployment Guide*.

Version 2021.2.5 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.2.5, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.2.5, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.2.4 (February 2022)

General Deployment

Before Deployment Commands

Deprecation Notice: The $deploy/sas-bases/overlays/network/ingress Directory

The **$deploy/sas-bases/overlays/network/ingress** directory is deprecated. The directory and the overlay that it contains will be removed from the SAS Viya deployment assets in the stable 2021.2.5 version and the long-term support 2022.1 version.

Therefore, you must revise the reference to the directory in the base kustomization.yaml file if it has not been revised already. In the resources block,

replace `- sas-bases/overlays/network/ingress` with `- sas-bases/overlays/network/networking.k8s.io`:

```
resources:
- sas-bases/base
- sas-bases/overlays/cert-manager-issuer
- sas-bases/overlays/network/networking.k8s.io
```

The example kustomization.yaml file located at "Create the File" in *SAS Viya Platform: Deployment Guide* reflects this revision.

Support for PostgreSQL CustomResourceDefinition with Kubernetes API v1

Note: If you are using an external instance of PostgreSQL, skip this deployment note.

Note: If you are using only one deployment of SAS Viya in your Kubernetes cluster, skip this deployment note.

Before version 2021.2.4, the PostgreSQL CustomResourceDefinition (CRD) used the v1beta1 Kubernetes API. However, the v1beta1 API has been deprecated and removed in Kubernetes 1.22. A revised PostgreSQL CRD using Kubernetes API v1 is being introduced in this version of SAS Viya. Therefore, if you are updating any deployment of SAS Viya with an internal instance of PostgreSQL in a Kubernetes cluster that contains multiple deployments of SAS Viya with internal instances of PostgreSQL, each deployment in the cluster must be fully updated to version 2021.2.3 before updating any of them to version 2021.2.4.

Remove login_hint for SAS Logon Manager

Prior to the 2021.2.4 version, users could apply a patch transformer when deploying SAS Viya to update the SAS Logon Manager ingress definition. This patch transformer used a server snippet annotation to insert a login_hint into requests. Adding the login_hint allowed customers using SAML or OpenID Connect to bypass SAS Logon Manager, meaning that end-users would automatically get redirected to the third-party SAML or OIDC Identity Provider (IdP). However, an issue with custom snippets was discovered in ingress-nginx, as documented in CVE-2021-25742. Therefore, the patch transformer must be removed for version 2021.2.4 and later.

To remove the patch transformer, review the transformers block of the base kustomization.yaml file. If the following line is in that block, remove it.

```
- site-config/sas-logon-app/login-hint-transformer.yaml
```

If the line is not present, then no action needs to be taken.

Remove Redis Components from the Deployment

Redis is a third-party product that will eventually be available in SAS Viya. However, components of Redis were inadvertently added to the deployment assets prior to its

being ready for release. Because of its presence, some administration tasks do not work as expected, so SAS recommends removing those components.

1 To determine if the Redis components are present in your deployment:

```
kubectl -n name-of-namespace get statefulsets | grep redis
```

If the command returns output that lists Redis statefulsets, continue with these steps. Otherwise, your deployment is unaffected and there are no further steps to take.

2 Perform each of the following commands:

- `kubectl -n name-of-namespace delete statefulsets redis-statefulset-name`

  Repeat this command for each statefulset that is listed in the output from step 1. For example, if the command in step 1 returns a list of three statefulsets, sas-redis-server-0, sas-redis-server-1, and sas-redis-server-2, you would perform this command three times, once for each statefulset.

- `kubectl -n name-of-namespace delete deployment sas-redis-operator`

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Version 2021.2.4 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-
cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.2.4, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.2.4, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.2.3 (January 2022)

General Deployment

Before Deployment Commands

Delete sas-admin-content-loader Job

Note: If you are updating your software with the SAS Viya Deployment Operator, skip this note.

If you use the manual process to update your software to version 2021.2.3, you must delete the sas-admin-content-loader Job before beginning the update:

```
kubectl -n name-of-namespace delete job sas-admin-content-loader
```

Change in Storage Class Configuration for SAS Programming Environment

The method by which you assign the storage class for runtime storage for processes such as SAS Compute server, SAS/CONNECT server, and SAS Batch server has been modified. If you used a YAML file to assign the storage class rather than use the default emptyDir, you must make revisions to your deployment.

Note: For more information about assigning storage classes for the SAS Programming Environment, see the README at **`$deploy/sas-bases/examples/sas-programming-environment/storage/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/sas_programming_environment_storage_tasks.htm`** (for HTML format).

1. Search for the following line in the transformers block of the base kustomization.yaml file.

   ```
   - site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml
   ```

   If that line is present, continue with these steps. Otherwise, there are no required changes and you should skip the rest of this deployment note.

2. Copy the **`$deploy/sas-bases/examples/sas-programming-environment/storage/change-viya-volume-storage-class.yaml`** file to the site-config directory, overwriting the existing file.

3. In the copied file, replace the `{{ VOLUME-STORAGE-CLASS }}` variable with the volume definition of the storage location you want to use.

4. In the base kustomization.yaml file, remove the `- site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml` from the transformers block.

5. Also in the base kustomization.yaml file, add the following content to the patches block:

   ```
   patches:
   - path: site-config/change-viya-volume-storage-class.yaml
     target:
       kind: PodTemplate
       labelSelector: "sas.com/template-intent=sas-launcher"
   ```

Remove Redis Components from the Deployment

Redis is a third-party product that will eventually be available in SAS Viya. However, components of Redis were inadvertently added to the deployment assets prior to its being ready for release. Because of its presence, some administration tasks do not work as expected, so SAS recommends removing those components.

1. To determine if the Redis components are present in your deployment:

   ```
   kubectl -n name-of-namespace get statefulsets | grep redis
   ```

If the command returns output that lists Redis statefulsets, continue with these steps. Otherwise, your deployment is unaffected and there are no further steps to take.

2 Perform each of the following commands:

- `kubectl -n name-of-namespace delete statefulsets redis-statefulset-name`

  Repeat this command for each statefulset that is listed in the output from step 1. For example, if the command in step 1 returns a list of three statefulsets, sas-redis-server-0, sas-redis-server-1, and sas-redis-server-2, you would perform this command three times, once for each statefulset.

- `kubectl -n name-of-namespace delete deployment sas-redis-operator`

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Version 2021.2.3 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Multi-tenancy and SAS Micro Analytic Service ASTORE and Resource Files

Changes in how SAS Micro Analytic Service handles analytic store (ASTORE) and resource files require changes to the configuration of those files.

1 To determine if this change affects your deployment, examine your base kustomization.yaml file.

   a Look for the following line in the transformers block of that file:

   ```
   - sas-bases/overlays/multi-tenant/sas-shared-configmap-transformer.yaml
   ```

   If the line is present, continue with the steps in this note. Otherwise, your deployment does not require this change and you should skip the rest of this note.

   b Look for the following lines, also in the transformer block of the base kustomization.yaml file:

   - `- sas-bases/overlays/sas-microanalytic-score/astores/astores-transformer.yaml`
   - `- site-config/sas-open-source-config/python/python-transformer.yaml`

   If either of these lines is present, continue with the steps in this note. If neither of these lines are present, your deployment does not require this change and you should skip the rest of this note.

2 Perform the steps described in the “Multi-Tenant Configuration” section of the “Configure SAS Micro Analytic Service to Support Analytic Stores” README, located at **`$deploy/sas-bases/examples/sas-microanalytic-store/astores/README.md`** (for Markdown format) or **`$deploy//sas-bases/docs/configure_sas_micro_analytic_service_to_support_analytic_stores.htm`** (for HTML format).

3 After completing the steps in the README, perform the steps to republish any existing models for the offerings that use SAS Micro Analytic Score:

- To republish model objects from SAS Model Manager, see “Publishing Models” in *SAS Model Manager: User’s Guide*.
- To republish decisions that contain models from SAS Intelligent Decisioning, see “Publishing a Decision” in *SAS Intelligent Decisioning: User’s Guide*.
- To republish analytic store models published from Model Studio, see “Publish Models” in *SAS Viya: Machine Learning User’s Guide*.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.2.3, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.2.3, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.2.2 (December 2021)

General Deployment

Before Deployment Commands

Change in Storage Class Configuration for SAS Programming Environment

The method by which you assign the storage class for runtime storage for processes such as SAS Compute server, SAS/CONNECT server, and SAS Batch server has been modified. If you used a YAML file to assign the storage class rather than use the default emptyDir, you must make revisions to your deployment.

Note: For more information about assigning storage classes for the SAS Programming Environment, see the README at **`$deploy/sas-bases/examples/`**

sas-programming-environment/storage/README.md (for Markdown format) or at **$deploy/sas-bases/docs/sas_programming_environment_storage_tasks.htm** (for HTML format).

1 Search for the following line in the transformers block of the base kustomization.yaml file.

```
- site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml
```

If that line is present, continue with these steps. Otherwise, there are no required changes and you should skip the rest of this deployment note.

2 Copy the **$deploy/sas-bases/examples/sas-programming-environment/storage/change-viya-volume-storage-class.yaml** file to the site-config directory, overwriting the existing file.

3 In the copied file, replace the `{{ VOLUME-STORAGE-CLASS }}` variable with the volume definition of the storage location you want to use.

4 In the base kustomization.yaml file, remove the `- site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml` from the transformers block.

5 Also in the base kustomization.yaml file, add the following content to the patches block:

```
patches:
- path: site-config/change-viya-volume-storage-class.yaml
  target:
    kind: PodTemplate
    labelSelector: "sas.com/template-intent=sas-launcher"
```

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Version 2021.2.2 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.

- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Ignore Log Entry from Crunchy Data

An update can result in an error entry in the sas-crunchy-data-postgres pod log that you can safely ignore.

When you update SAS Viya 2021.1.6 to SAS Viya 2021.2.2, you might see an error entry if you are using the internal instance of SAS Infrastructure Data Server. The internal instance is based on Crunchy PostgreSQL for Kubernetes. After the update, the sas-crunchy-data-postgres pod log includes the following entry:

```
level=error msg=queryNamespaceMappings returned 1 errors
```

SAS testing has determined that this error does not affect the functionality of the deployment. SAS is working with Crunchy Data to remove the relevant error reporting.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.2.2, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.2.2, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.2.1 (November 2021)

General Deployment

Before Deployment Commands

Changes in the Directory Structure for Multi-tenancy

Note: If your deployment does not have multi-tenancy enabled, skip this note.

Changes in the directory structure for multi-tenancy require performing some steps before updating your SAS Viya software.

1. If you are unsure whether you enabled multi-tenancy, review your base kustomization.yaml file. If the site-config/multi-tenant/sas-shared-configmap-transformer.yaml file is listed in the transformers block, multi-tenancy is enabled and you should continue with these steps.
2. Delete the **`$deploy/site-config/multi-tenant/sas-shared-configmap-transformer.yaml`** file.
3. In the base kustomization.yaml file, remove the reference to the site-config/multi-tenant/sas-shared-configmap-transformer.yaml file and replace it with a reference to sas-bases/overlays/multi-tenant/sas-shared-configmap-transformer.yaml. Here is an example of the base kustomization.yaml file before the change:

   ```
   transformers:
   ...
   - site-config/multi-tenant/sas-shared-configmap-transformer.yaml
   ```

   Here is an example of the same file after the change:

   ```
   transformers:
   ...
   - sas-bases/overlays/multi-tenant/sas-shared-configmap-transformer.yaml
   ```

Change in Storage Class Configuration for SAS Programming Environment

The method by which you assign the storage class for runtime storage for processes such as SAS Compute server, SAS/CONNECT server, and SAS Batch

server has been modified. If you used a YAML file to assign the storage class rather than use the default emptyDir, you must make revisions to your deployment.

Note: For more information about assigning storage classes for the SAS Programming Environment, see the README at **$deploy/sas-bases/examples/sas-programming-environment/storage/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_programming_environment_storage_tasks.htm** (for HTML format).

1. Search for the following line in the transformers block of the base kustomization.yaml file.

   ```
   - site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml
   ```

   If that line is present, continue with these steps. Otherwise, there are no required changes and you should skip the rest of this deployment note.

2. Copy the **$deploy/sas-bases/examples/sas-programming-environment/storage/change-viya-volume-storage-class.yaml** file to the site-config directory, overwriting the existing file.

3. In the copied file, replace the `{{ VOLUME-STORAGE-CLASS }}` variable with the volume definition of the storage location you want to use.

4. In the base kustomization.yaml file, remove the `- site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml` from the transformers block.

5. Also in the base kustomization.yaml file, add the following content to the patches block:

   ```
   patches:
   - path: site-config/change-viya-volume-storage-class.yaml
     target:
       kind: PodTemplate
       labelSelector: "sas.com/template-intent=sas-launcher"
   ```

Spawn SAS/CONNECT Servers Within the Spawner Pod

Beginning in 2021.2.1, the ability to spawn servers within the spawner pod has been removed from SAS/CONNECT Spawner. However, this capability may be re-introduced by applying the security settings in the enable-spawned-servers.yaml example file. For the steps to re-enable spawning servers within the spawner pod, see the “Allow the Ability to Spawn Servers within the Spawner Pod” section of the “Configure SAS/CONNECT Spawner in SAS Viya” README file located at **$deploy/sas-bases/examples/sas-connect-spawner/README.md** (for Markdown format) and at **$deploy/sas-bases/docs/configure_sasconnect_spawner_in_sas_viya.htm** (for HTML format).

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Version 2021.2.1 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Check Open Distro for Elasticsearch Cluster

Note: If your deployment does not contain Open Distro for Elasticsearch, skip these instructions.

Due to a timing issue, an update can result in an invalid StatefulSet manifest, which prevents the sas-opendistro-default-0 (or other master/data nodes) from coming up.

1 After updating SAS Viya, ensure that the Open Distro for Elasticsearch cluster pods are running. If your deployment uses the default topology for Open Distro for Elasticsearch, use the following command:

   ```
   kubectl -n name-of-namespace get pods | grep sas-opendistro-default
   ```

   If you created a custom topology using the custom-topology.yaml example file, use the following command:

```
kubectl -n name-of-namespace get pods | grep sas-opendistro-node-set
```

In this command, *node-set* is a custom Elasticsearch node set defined in the example file. If you have more than one custom node set, the command should be performed for each node set.

2 Restart each pod that is in a `CrashLoopBackOff` state in the output from step 1. To restart an Elasticsearch pod in a default topology, delete it:

```
kubectl -n name-of-namespace delete pod sas-opendistro-default-pod-number
```

To restart an Elasticsearch pod in a custom topology:

```
kubectl -n name-of-namespace delete pod sas-opendistro-node-set-pod-number
```

3 Review the log of each sas-opendistro-default-*pod-number* pod (or each sas-opendistro-*node-set-pod-number* pod if you have custom topologies) for the following error.

```
org.elasticsearch.bootstrap.StartupException: java.lang.IllegalArgumentException:
unknown setting [node.roles] please check that any required plugins are
installed, or check the breaking changes documentation for removed settings
```

If the error is in the log, restart the pod using the commands described in step 2.

Avoid Long Delay for SAS/CONNECT Server Sign-on from SAS Studio

Setting `TCPNOIPADDR` to 1 avoids a long (2–minute) delay during sign-on to a SAS/CONNECT server from an internal SAS Viya client, such as SAS Studio.

This problem has been observed on Microsoft Azure, AWS, and GCP. The delay is caused by an errant connection back to the SAS/CONNECT spawner. The spawner hangs until time-out, at which point a successful connection is made to the launched SAS/CONNECT server.

A fix is planned. In the meantime, run your programs with `TCPNOIPADDR=1` to work around the issue that causes the delay.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.2.1, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.2.1, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.1.6 (October 2021)

General Deployment

Before Deployment Commands

Revise Reference to Ingress in Initial kustomization.yaml File

Because of changes to Kubernetes, the reference to ingress in the base kustomization.yaml file for earlier versions of SAS Viya must be revised for version 2021.1.6.

Here is how the base kustomization.yaml file should appear for earlier versions:

```
resources:
- sas-bases/base
- sas-bases/overlays/cert-manager-issuer
- sas-bases/overlays/network/ingress
```

Here is the same block for version 2021.1.6:

```
resources:
- sas-bases/base
- sas-bases/overlays/cert-manager-issuer
- sas-bases/overlays/network/networking.k8s.io
```

Change in Storage Class Configuration for SAS Programming Environment

The method by which you assign the storage class for runtime storage for processes such as SAS Compute server, SAS/CONNECT server, and SAS Batch server has been modified. If you used a YAML file to assign the storage class rather than use the default emptyDir, you must make revisions to your deployment.

Note: For more information about assigning storage classes for the SAS Programming Environment, see the README at **$deploy/sas-bases/examples/sas-programming-environment/storage/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_programming_environment_storage_tasks.htm** (for HTML format).

1 Search for the following line in the transformers block of the base kustomization.yaml file.

```
- site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml
```

If that line is present, continue with these steps. Otherwise, there are no required changes and you should skip the rest of this deployment note.

2 Copy the **$deploy/sas-bases/examples/sas-programming-environment/storage/change-viya-volume-storage-class.yaml** file to the site-config directory, overwriting the existing file.

3 In the copied file, replace the `{{ VOLUME-STORAGE-CLASS }}` variable with the volume definition of the storage location you want to use.

4 In the base kustomization.yaml file, remove the `- site-config/sas-programming-environment/storage/change-viya-volume-storage-class.yaml` from the transformers block.

5 Also in the base kustomization.yaml file, add the following content to the patches block:

```
patches:
- path: site-config/change-viya-volume-storage-class.yaml
  target:
    kind: PodTemplate
    labelSelector: "sas.com/template-intent=sas-launcher"
```

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Version 2021.1.6 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.

- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Check Open Distro for Elasticsearch Cluster

Note: If your deployment does not contain Open Distro for Elasticsearch, skip these instructions.

Due to a timing issue, an update can result in an invalid StatefulSet manifest, which prevents the sas-opendistro-default-0 (or other master/data nodes) from coming up.

1 After updating SAS Viya, ensure that the Open Distro for Elasticsearch cluster pods are running. If your deployment uses the default topology for Open Distro for Elasticsearch, use the following command:

```
kubectl -n name-of-namespace get pods | grep sas-opendistro-default
```

If you created a custom topology using the custom-topology.yaml example file, use the following command:

```
kubectl -n name-of-namespace get pods | grep sas-opendistro-node-set
```

In this command, *node-set* is a custom Elasticsearch node set defined in the example file. If you have more than one custom node set, the command should be performed for each node set.

2 Restart each pod that is in a `CrashLoopBackOff` state in the output from step 1. To restart an Elasticsearch pod in a default topology, delete it:

```
kubectl -n name-of-namespace delete pod sas-opendistro-default-pod-number
```

To restart an Elasticsearch pod in a custom topology:

```
kubectl -n name-of-namespace delete pod sas-opendistro-node-set-pod-number
```

3 Review the log of each sas-opendistro-default-*pod-number* pod (or each sas-opendistro-*node-set-pod-number* pod if you have custom topologies) for the following error.

```
org.elasticsearch.bootstrap.StartupException: java.lang.IllegalArgumentException:
unknown setting [node.roles] please check that any required plugins are
installed, or check the breaking changes documentation for removed settings
```

If the error is in the log, restart the pod using the commands described in step 2.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.1.6, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.1.6, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.1.5 (September 2021)

General Deployment

Before Deployment Commands

Updates to cachelocator and cacheserver Services

Because of changes to sas-cachelocator and sas-cacheserver, in order to deploy the new versions, you must delete the old versions when you update to release 2021.1.5.

```
kubectl -n name-of-namespace delete service sas-cachelocator sas-cacheserver
```

Revise PersistentVolumeClaim annotationSelector

Due to changes in the software for version 2021.1.5, the patches block of the base kustomization.yaml file must be modified. In the annotationSelector line, add `sas-pyconfig`:

```
patches:
- path: site-config/storageclass.yaml
  target:
    kind: PersistentVolumeClaim
    annotationSelector: sas.com/component-name in (sas-backup-job,sas-
data-quality-services,sas-commonfiles,sas-cas-operator,sas-pyconfig)
```

To see the code sample in context, go to "Create the File" in *SAS Viya Platform: Deployment Guide*.

Changes to Python Configuration

Beginning in version 2021.1.5, the python-transformer.yaml file contains changes that affect configuring Python volumes. If you configured Python in a previous version of SAS Viya, you must modify the python-transformer.yaml file before updating to version 2021.1.5.

1. Search your base kustomization.yaml file for a reference to python-transformer.yaml in the transformers block. If the reference does not exist, no action is necessary.
2. Otherwise, delete the **$deploy/site-config/sas-open-source-config/python/python-transformer.yaml** file.
3. To create the new configuration, follow the steps in the README file at **$deploy/sas-bases/examples/sas-open-source-config/python/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_python_for_sas_viya.htm** (for HTML format).

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Version 2021.1.5 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.1.5, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.1.5, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.1.4 (August 2021)

General Deployment

Before Deployment Commands

Internal Instances of PostgreSQL Moving to Operator

Note: If you have an external instance of PostgreSQL, skip this deployment note.

Starting with version 2021.1.4, deployment and administration of internal instances of PostgreSQL are performed with the SAS Data Server Operator. If you are updating to version 2021.1.4, you must remove the references to the previous deployment method and replace them with operator-specific files.

1 In the base kustomization.yaml file, make the following changes:
   - Remove `- sas-bases/overlays/crunchydata` from the resources block.
   - Remove `- sas-bases/overlays/internal-postgres/internal-postgres-transformer.yaml` from the transformers block.
   - Remove `- site-config/postgres/postgres-custom-config.yaml` from the generators block. If it is the only file listed in that block, remove the block as well.

2 To apply custom resource limits:
   a Copy the **`$deploy/sas-bases/examples/postgres/pod-resource-limits-settings/postgres-pods-resource-limits-settings-transformer.yaml`** file to your site-config directory. If postgres-pods-resource-limits-settings-transformer.yaml already exists in the site-config directory, delete the old file and replace it with the new.
   b Using the comments in the file, revise the copied file with the appropriate values for the internal instance of PostgreSQL that it represents.

   Note: By default, the values apply to all PostgreSQL clusters. If you want different values among the clusters, make an individual entry per cluster.

c In the transformers block of the base kustomization.yaml file, add a reference to the postgres-pods-resource-limits-settings-transformer.yaml file. Here is an example that assumes the copied file was placed in **$deploy/site-config/postgres/pod-resource-limits-settings/postgres-pods-resource-limits-settings-transformer.yaml**:

```
transformers:
...
- site-config/postgres/pod-resource-limits-settings/postgres-pods-resource-limits-settings-transformer.yaml
...
```

If you replaced an existing version of postgres-pods-resource-limits-settings-transformer.yaml with a new one for this update, do not change the base kustomization.yaml file since the reference already exists.

3 To use custom replica counts:

a Copy the **$deploy/sas-bases/examples/postgres/replicas/postgres-replicas-transformer.yaml** file to your site-config directory. If postgres-replicas-transformer.yaml already exists in the site-config directory, delete the old file and replace it with the new.

b Using the comments in the file, revise the copied file with the appropriate values for the internal instance of PostgreSQL that it represents.

Note: By default the values apply to all PostgreSQL clusters. If you want different values among the clusters, make an individual entry per cluster.

c In the transformers block of the base kustomization.yaml file, add a reference to the postgres-replicas-transformer.yaml file. Here is an example that assumes the copied file was placed in **$deploy/site-config/postgres/replicas/postgres-replicas-transformer.yaml**:

```
transformers:
...
- site-config/postgres/replicas/postgres-replicas-transformer.yaml
...
```

If you replaced an existing version of postgres-replicas-transformer.yaml with a new one for this update, do not change the base kustomization.yaml file since the reference already exists.

4 To use custom storage settings:

a Copy the **$deploy/sas-bases/examples/postgres/storage/postgres-storage-transformer.yaml** file to your site-config directory. If postgres-storage-transformer.yaml already exists in the site-config directory, delete the old file and replace it with the new.

b Using the comments in the file, revise the copied file with the appropriate values for the internal instance of PostgreSQL that it represents.

Note: By default the values apply to all PostgreSQL clusters. If you want different values among the clusters, make an individual entry per cluster.

c In the transformers block of the base kustomization.yaml file, add a reference to the postgres-storage-transformer.yaml file. Here is an example that

assumes the copied file was placed in **$deploy/site-config/postgres/storage/postgres-storage-transformer.yaml**:

```
transformers:
...
- site-config/postgres/storage/postgres-storage-transformer.yaml
...
```

If you replaced an existing version of postgres-storage-transformer.yaml with a new one for this update, do not change the base kustomization.yaml file since the reference already exists.

5 Determine whether Common Planning Service is included in your deployment by running the following command:

```
kubectl -n name-of-namespace get pods | grep cpspostgres
```

If the results include any pod names, then Common Planning Service is included in your deployment. You should consider potential configurations for the Common Planning Service instance of PostgreSQL in the remaining steps.

6 To use custom PostgreSQL settings:

a Copy the following files to your site-config directory:

- **$deploy/sas-bases/examples/postgres/custom-config/sas-postgres-custom-config.yaml**
- **$deploy/sas-bases/examples/postgres/custom-config/sas-postgres-custom-config-transformer.yaml**
- If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspostgres-custom-config.yaml**
- If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspostgres-custom-config-transformer.yaml**

b Using the comments in the files, revise the copied files with the appropriate values for the internal instance of PostgreSQL that it represents.

c In the base kustomization.yaml file, add references to the modified files.

- In the resources block, add references for sas-postgres-custom-config.yaml and sas-cpspostgres-custom-config.yaml (if needed). Here is an example that assumes the copied files were placed in **$deploy/site-config/postgres/custom-config**:

  ```
  resources:
  ...
  - site-config/postgres/custom-config/sas-postgres-custom-config.yaml
  - site-config/postgres/custom-config/sas-cpspostgres-custom-config.yaml
  ...
  ```

- In the transformers block, add references for sas-postgres-custom-config-transformer.yaml and sas-cpspostgres-custom-config-transformer.yaml (if needed). Here is an example that assumes the copied files were placed in **$deploy/site-config/postgres/custom-config**:

  ```
  transformers:
  ...
  - site-config/postgres/custom-config/sas-postgres-custom-config-transformer.yaml
  ```

```
- site-config/postgres/custom-config/sas-cpspostgres-custom-config-transformer.yaml
...
```

7 To use custom pg_hba settings:

   a Copy the following files to your site-config directory:

      - **$deploy/sas-bases/examples/postgres/custom-config/sas-pghba-custom-config.yaml**
      - **$deploy/sas-bases/examples/postgres/custom-config/sas-pghba-custom-config-transformer.yaml**
      - If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspghba-custom-config.yaml**
      - If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspghba-custom-config-transformer.yaml**

   b Using the comments in the files, revise the copied files with the appropriate values for the internal instance of PostgreSQL that it represents.

   c In the base kustomization.yaml file, add references to the modified files.

      - In the resources block, add references for sas-pghba-custom-config.yaml and sas-cpspghba-custom-config.yaml (if needed). Here is an example that assumes the files were placed in **$deploy/site-config/postgres/custom-config**:

        ```
        resources:
        ...
        - site-config/postgres/custom-config/sas-pghba-custom-config.yaml
        - site-config/postgres/custom-config/sas-cpspghba-custom-config.yaml
        ...
        ```

      - In the transformers block, add references for sas-pghba-custom-config-transformer.yaml and sas-cpspghba-custom-config-transformer.yaml (if needed). Here is an example that assumes the files were placed in **$deploy/site-config/postgres/custom-config**:

        ```
        transformers:
        ...
        - site-config/postgres/custom-config/sas-pghba-custom-config-transformer.yaml
        - site-config/postgres/custom-config/sas-cpspghba-custom-config-transformer.yaml
        ...
        ```

8 To use custom Patroni settings:

   a Copy the following files to your site-config directory:

      - **$deploy/sas-bases/examples/postgres/custom-config/sas-patroni-custom-config.yaml**
      - **$deploy/sas-bases/examples/postgres/custom-config/sas-patroni-custom-config-transformer.yaml**
      - If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspatroni-custom-config.yaml**

- If you have Common Planning Service in your deployment, **$deploy/sas-bases/examples/postgres/custom-config/sas-cpspatroni-custom-config-transformer.yaml**

b Using the comments in the files, revise the copied files with the appropriate values for the internal instance of PostgreSQL that it represents.

c In the base kustomization.yaml file, add references to the modified files.

- In the resources block of the base kustomization.yaml file, add a reference to the sas-patroni-custom-config.yaml file and the sas-cpspatroni-custom-config.yaml file (if needed). Here is an example that assumes the files were placed in **$deploy/site-config/postgres/custom-config**:

```
resources:
...
- site-config/postgres/custom-config/sas-patroni-custom-config.yaml
- site-config/postgres/custom-config/sas-cpspatroni-custom-config.yaml
...
```

- In the transformers block of the base kustomization.yaml file, add a reference to the sas-patroni-custom-config-transformer.yaml file and the sas-cpspatroni-custom-config-transformer.yaml file (if needed). Here is an example that assumes the files were placed in **$deploy/site-config/postgres/custom-config**:

```
transformers:
...
- site-config/postgres/custom-config/sas-patroni-custom-config-transformer.yamll
- site-config/postgres/custom-config/sas-cpspatroni-custom-config-transformer.yaml
...
```

Changes for External Instances of PostgreSQL

Note: If you have an internal instance of PostgreSQL, skip this deployment note.

Version 2021.1.4 introduces a new method of connecting your SAS Viya deployment to an external instance of PostgreSQL. If you are updating to version 2021.1.4, you must remove artifacts of the old connection method and replace them with the new method.

1 For each instance of external PostgreSQL in your SAS Viya deployment, make a copy of the **$deploy/examples/postgres/configure/external-postgres.yaml** and place it in your site-config directory. Each copy should be uniquely named.

2 Using the comments in the file, revise each copy of external-postgres.yaml with the appropriate values for the external instance of PostgreSQL that it represents.

3 In the resources block of the base kustomization.yaml, add a reference for each external-postgres.yaml. Here is an example with two external instances of PostgreSQL:

```
resources
...
-site-config/postgres/configure/external-postgres-1.yaml
-site-config/postgres/configure/external-postgres-2.yaml
```

```
...
```

4 Also, in the base kustomization.yaml file, remove the secretGenerator and configMapGenerator for the external instances of PostgreSQL. The secretGenerator begins with `- name: postgres-sas-user` and the configMapGenerator begins with `- name: sas-postgres-config`.

Changes for External Instances of Common Planning Service PostgreSQL

Version 2021.1.4 introduces a new method of Common Planning Service to an external instance of PostgreSQL. If you are updating to version 2021.1.4, you must remove artifacts of the old connection method and replace them with the new method.

1 If you are uncertain whether your deployment uses Common Planning Service PostgreSQL, examine the transformers block in the base kustomization.yaml file. If both of the following two lines are included in that block, continue with these instructions. If both lines are not included in the block, skip the rest of this deployment note.

- `- sas-bases/overlays/sas-planning/sas-planning-transformer.yaml`
- `- sas-bases/overlays/external-postgres/external-postgres-transformer.yaml`

2 Copy the **`$deploy/sas-bases/examples/postgres/configure/external-cpspostgres.yaml`** file to the site-config subdirectory, such as**`$deploy/site-config/postgres/configure/external-cpspostgres.yaml`**.

3 Adjust the values in the copied file as necessary for your external PostgreSQL instance.

4 Update the base kustomization.yaml file by adding the location of the copied file to the resources block of that file. Here is an example:

```
resources:
...
- site-config/postgres/configure/external-cpspostgres.yaml
...
```

5 Remove the associated secretGenerators from the base kustomization.yaml file. Here are the names of the secretGenerators to be removed:

- `- name: postgres-sas-user`
- `- name: cpspostgres-sas-user`

6 Remove the configMapGenerator with the name `- name: sas-postgres-config` from the base kustomization.yaml file.

Updates to cachelocator and cacheserver Services

Because of changes to sas-cachelocator and sas-cacheserver, in order to deploy the new versions, you must delete the old versions when you update to release 2021.1.4.

```
kubectl -n name-of-namespace delete service sas-cachelocator sas-cacheserver
```

Changes to Python Configuration

Beginning in version 2021.1.4, the python-transformer.yaml file contains changes that affect configuring Python volumes. If you configured Python in a previous version of SAS Viya, you must modify the python-transformer.yaml file before updating to version 2021.1.4.

1. Search your base kustomization.yaml file for a reference to python-transformer.yaml in the transformers block. If the reference does not exist, no action is necessary.
2. Otherwise, delete the **$deploy/site-config/sas-open-source-config/python/python-transformer.yaml** file.
3. To create the new configuration, follow the steps in the README file at **$deploy/sas-bases/examples/sas-open-source-config/python/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_python_for_sas_viya.htm** (for HTML format).

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Version 2021.1.4 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although you must delete the CAS server pods in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts the new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.1.4, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.1.4, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.1.3 (July 2021)

General Deployment

Before Deployment Commands

Changes to TLS

Note: If your SAS Viya deployment does not use TLS, you should skip this deployment note.

Beginning with release 2021.1.3, SAS Viya uses Kustomize components to deploy and configure TLS. Moving to the new process requires a number of changes to your base kustomization.yaml file (**`$deploy/kustomization.yaml`**).

1. In the resources block, remove the `sas-bases/overlays/network/ingress/security` line.
2. In the transformers block, remove any line that includes `tls`. Here are lines that should be removed:
   - `sas-bases/overlays/network/ingress/security/transformers/product-tls-transformers.yaml`
   - `sas-bases/overlays/network/ingress/security/transformers/ingress-tls-transformers.yaml`
   - `sas-bases/overlays/network/ingress/security/transformers/backend-tls-transformers.yaml`
   - `sas-bases/overlays/network/ingress/security/transformers/truststore-transformers-without-backend-tls.yaml`
   - `sas-bases/overlays/network/ingress/security/transformers/cas-connect-tls-transformers.yaml`
3. To enable NGINX TLS for full-stack TLS mode, add the following lines to the components block of the base kustomization.yaml file. Create the components block if it does not already exist.

```
components:
- sas-bases/components/security/core/base/full-stack-tls
- sas-bases/components/security/network/networking.k8s.io/ingress/nginx.ingress.kubernetes.io/full-stack-tls
```

For information about the required additions to the base kustomization.yaml for other modes of TLS, see the README file at `$deploy/sas-bases/examples/security/README.md` (for Markdown format) or at `$deploy/sas-bases/docs/configure_network_security_and_encryption_using_sas_security_certificate_framework.htm` (for HTML format).

Note: The example kustomization.yaml file at "Create the File" in *SAS Viya Platform: Deployment Guide* contains the appropriate changes to enable NGINX TLS for full-stack TLS mode.

Enable Host Launch for the CAS Server

Note: If you meet any of the following conditions in your deployment, you must enable host launch:

- Your deployment uses Kerberos.
- Your deployment enables SSSD and disables CASCLOUDNATIVE.
- Your deployment uses the CASHostAccountRequired custom group or sets the CASALLHOSTACCOUNTS environment variable.

By default, CAS cannot launch sessions under a user's host identity. All sessions run under the cas service account instead. CAS can be configured to allow for host identity launches by including a patch transformer in the kustomization.yaml file. To enable host launch for CAS, see the "Enable Host Launch in the CAS Server" section of the README file located at `$deploy/sas-bases/examples/cas/configure/README.md` (for Markdown format) or at `$deploy/sas-bases/docs/configuration_settings_for_cas.htm` (for HTML format).

Changes to Compute Server Example Files

Changes have been made to the Compute Server example files to ensure that the Compute Server and Batch Server have the same mounts. Perform the following steps if you are updating to version 2021.1.3:

1. Look in the transformers block of the base kustomization.yaml file for references to the following example files:
   - compute-server-add-host-mount.yaml
   - compute-server-add-nfs-mount.yaml

   If neither of those files is included in the transformers block, then there are no further actions to take.
2. If either or both of the files is included in the transformer block, go to the location described in the transformer block and remove the file or files.
3. Copy the new version of the file or files from `$deploy/sas-bases/examples/sas-compute-server/configure` and paste them in the location from which you just deleted the old files.

4 Follow the instructions in the “Configuration Settings for Compute Server” README file located at **$deploy/sas-bases/examples/sas-compute-server/configure/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configuration_settings_for_compute_server.htm** (for HTML format) to modify the new file or files for your deployment. You do not have to revise the base kustomization.yaml file since you are putting the new files in the same location that the old files were in.

Updates to cachelocator and cacheserver Services

Because of changes to sas-cachelocator and sas-cacheserver, in order to deploy the new versions, you must delete the old versions when you update to release 2021.1.3.

```
kubectl -n name-of-namespace delete service sas-cachelocator sas-cacheserver
```

Changes to Python Configuration

Beginning in version 2021.1.3, the python-transformer.yaml file contains changes that affect configuring Python volumes. If you configured Python in a previous version of SAS Viya, you must modify the python-transformer.yaml file before updating to version 2021.1.3.

1 Search your base kustomization.yaml file for a reference to python-transformer.yaml in the transformers block. If the reference does not exist, no action is necessary.

2 Otherwise, delete the **$deploy/site-config/sas-open-source-config/python/python-transformer.yaml** file.

3 To create the new configuration, follow the steps in the README file at **$deploy/sas-bases/examples/sas-open-source-config/python/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_python_for_sas_viya.htm** (for HTML format).

After Deployment Commands

Delete sas-cacheserver-0

A new version of sas-cacheserver is included in version 2021.1.3. If you update to version 2021.1.3, the sas-cacheserver-1 pod might be blocked from reaching a READY state by sas-cacheserver-0. If sas-cacheserver-1 does not reach the READY state in an appropriate amount of time, you should delete sas-cacheserver-0.

After updating your software, run the following command to determine the status of the sas-cacheserver pods:

```
kubectl -n name-of-namespace get pods -l "app in (sas-cacheserver)"
```

The output should look like this:

```
NAMESPACE      NAME                                  READY   STATUS        RESTARTS   AGE
d10006         sas-cacheserver-0                     1/1     Running       0          5s
d10006         sas-cacheserver-1                     1/1     Running       0          5s
```

The READY value for the sas-cacheserver-1 pod should be `1/1` and the STATUS should be `RUNNING`. It might take a few moments for the READY state to reach `1/1` after the STATUS is `RUNNING`.

If the sas-cacheserver-1 pod does not reach the value of `1/1`, delete the sas-cacheserver-0 pod:

```
kubectl -n name-of-namespace delete pods sas-cacheserver-0
```

After the sas-cacheserver-0 pod is deleted and is automatically restarted by Kubernetes, the sas-cacheserver-1 pod finishes its initialization and reaches the READY state. Other pods that are unable to initialize might also reach the READY state after sas-cacheserver-0 is deleted and restarted.

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, you should skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Version 2021.1.3 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although deleting the CAS server pods is required in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.1.3, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.1.3, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2021.1.2 (June 2021)

General Deployment

Before Deployment Commands

Changes to Python Configuration

Beginning in version 2021.1.2, the python-transformer.yaml file contains a PatchTransformer that sets paths to the Python executable so that the SAS runtime can execute Python code. If you configured Python in a previous version of SAS Viya, you must modify the python-transformer.yaml file before updating to version 2021.1.2.

1 Search your base kustomization.yaml file for a reference to python-transformer.yaml in the transformers block. If the reference does not exist, no action is necessary.

2 Otherwise, delete the `$deploy/site-config/sas-open-source-config/python/python-transformer.yaml` file.

3 To create the new configuration, follow the steps in the README file at `$deploy/sas-bases/examples/sas-open-source-config/python/README.md` (for Markdown format) or at `$deploy/sas-bases/docs/configure_python_for_sas_viya.htm` (for HTML format).

Changes to R Configuration

Beginning in version 2021.1.2, the r-transformer.yaml file contains a PatchTransformer that sets paths to the R interpreter so that the SAS runtime can execute the R script. If you configured R in a previous version of SAS Viya, you must modify the r-transformer.yaml file before updating to version 2021.1.2.

1 Search your base kustomization.yaml file for a reference to r-transformer.yaml in the transformers block. If the reference does not exist, no action is necessary.

2 Otherwise, delete the `$deploy/site-config/sas-open-source-config/r/r-transformer.yaml` file.

3 To create the new configuration, follow the steps in the README file located at `$deploy/sas-bases/examples/sas-open-source-config/r/README.md` (for Markdown format) or at `$deploy/sas-bases/docs/configure_r_for_sas_viya.htm` (for HTML format).

Deployment Commands

Additional Deployment Command for Manual Deployments

Note: If you are deploying or updating with SAS Viya Deployment Operator, skip this note.

A new command has been added to the series of commands to be performed when updating or deploying SAS Viya manually. For the updated list of steps, see "Deployment Using Kubernetes Commands" in *SAS Viya Platform: Deployment Guide*.

After Deployment Commands

Update PostgreSQL Client

Note: If you have an external instance of PostgreSQL, you should skip this deployment note.Also, if you have not deployed the optional pgo client, you should skip this deployment note.

A new version of pgo client from Crunchy Data is now available. After an update, you must ensure that the client and server are using the same version.

1 Determine the version of pgo client in your deployment with this command.

```
pgo version
```

The output looks like this:

```
pgo client version 4.6.3
Handling connection for 8443
pgo-apiserver version 4.6.3
```

If the version number for `pgo client` is `4.6.3`, no further actions are required.

2 If the version number for `pgo client` is not `4.6.3`, install the new pgo client with the instructions at "Configure the PostgreSQL Operator (pgo) Client Using kubectl Plug-in" in *SAS Viya Platform: Infrastructure Servers*.

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, you should skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Version 2021.1.2 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although deleting the CAS server pods is required in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.1.2, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.1.2, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

SAS Model Publish

Before Deployment Commands

Changes to sas-model-publish Overlays

Release 2021.1.2 contains changes to the overlay files for Git and Kaniko, which are located in **`$deploy/sas-bases/overlays/sas-model-publish`**. To ensure that your update is successful, evaluate the transformers block of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**).

- If that block contains `sas-bases/overlays/sas-model-publish/kaniko/kaniko-transformer.yaml`, you should have SAS Model Manager in your SAS Viya deployment. If SAS Model Manager is not in your deployment, delete the `sas-bases/overlays/sas-model-publish/kaniko/kaniko-transformer.yaml` line.
- If that block contains `sas-bases/overlays/sas-model-publish/git/git-transformer.yaml`, you should have SAS Intelligent Decisioning or SAS Model Manager in your SAS Viya deployment. If neither of those offerings is in your deployment, delete the `sas-bases/overlays/sas-model-publish/git/git-transformer.yaml` line.

SAS Risk Engine

Before Deployment Commands

Automatic Migration of Logic Objects Not Supported

The SAS Risk Engine user interface uses PostgreSQL to store the logic objects (for example, risk methods, model groups, and programs). The PostgreSQL schema was changed significantly in the 2021.1.2 release. Therefore, automatic migration from previous releases is not supported.

For the steps to migrate to release 2021.1.2, see Migrate the PostgreSQL Database.

2021.1.1 (May 2021)

General Deployment

Before Deployment Commands

Delete sas-cacheserver and sas-cachelocator Services

Note: This deployment note applies only to manual deployments of SAS Viya. If you use the SAS Viya Deployment Operator to deploy and maintain your software, the procedures described in this deployment note are performed for you, and you should skip this deployment note.

A change has been added to sas-cacheserver and sas-cachelocator with version 2021.1.1 that requires that you delete the existing services before an update.

If you are updating to version 2021.1.1, run the following command:

```
kubectl -n name-of-namespace delete service sas-cacheserver sas-cachelocator
```

After Deployment Commands

Update CAS Servers

Note: If you are updating your software with the SAS Viya Deployment Operator and you have applied the cas-auto-restart.yaml transformer to your deployment, you should skip this deployment note. For more information about the cas-auto-restart.yaml transformer, see "Enable CAS Auto-Restart After Updates" in *SAS Viya Platform: Deployment Guide*.

Version 2021.1.1 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although deleting the CAS server pods is required in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2021.1.1, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2021.1.1, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

SAS Information Catalog

After Deployment Commands

Rebuild Datasets Search Index

After an update, the structure of the datasets search index has been modified, requiring the index to be rebuilt. To rebuild the index, perform the steps described at "Respond to Notifications" in *SAS Information Catalog: Administrator's Guide*.

2020.1.5 (April 2021)

General Deployment

Before Deployment Commands

Revise Reference to sitedefault.yaml in the kustomization.yaml File

Version 2020.1.5 includes changes in how SAS Viya works with the sitedefault file. Prior to version 2020.1.5, users were directed to create an entry in the configMapGenerator block of their base kustomization.yaml if their deployments use a sitedefault.yaml file. Beginning in version 2020.1.5, SAS recommends that the reference be moved to the secretGenerator block of the base kustomization.yaml file for security purposes. See "Add a sitedefault File to Your Deployment" in *SAS Viya Platform: Deployment Guide* for details of the new location.

After Deployment Commands

Update CAS Servers

Version 2020.1.5 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although deleting the CAS server pods is required in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

Delete sas-cacheserver-0

A new version of sas-cacheserver is included in version 2020.1.5. If you update to version 2020.1.5, the sas-cacheserver-1 pod might be blocked from reaching a READY state by sas-cacheserver-0. If sas-cacheserver-1 does not reach the READY state in an appropriate amount of time, you should delete sas-cacheserver-0.

After updating your software, run the following command to determine the status of the sas-cacheserver pods:

```
kubectl -n name-of-namespace get pods -l "app in (sas-cacheserver)"
```

The output should look like this:

```
NAMESPACE      NAME                                  READY   STATUS        RESTARTS   AGE
d10006         sas-cacheserver-0                     1/1     Running       0          5s
d10006         sas-cacheserver-1                     1/1     Running       0          5s
```

The READY value for the sas-cacheserver-1 pod should be `1/1` and the STATUS should be `RUNNING`. It might take a few moments for the READY state to reach `1/1` after the STATUS is `RUNNING`.

If the sas-cacheserver-1 pod does not reach the value of `1/1`, delete the sas-cacheserver-0 pod:

```
kubectl -n name-of-namespace delete pods sas-cacheserver-0
```

After the sas-cacheserver-0 pod is deleted and is automatically restarted by Kubernetes, the sas-cacheserver-1 pod finishes its initialization and reaches the READY state. Other pods that are unable to initialize might also reach the READY state after sas-cacheserver-0 is deleted and restarted.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2020.1.5, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2020.1.5, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2020.1.4 (March 2021)

General Deployment

Before Deployment Commands

Delete RabbitMQ Service

Note: This deployment note applies only to manual deployments of SAS Viya. If you use the SAS Viya Deployment Operator to deploy and maintain your software,

the procedures described in this deployment note are performed for you, and you should skip this deployment note.

A change has been added to RabbitMQ with version 2020.1.4 that requires the existing RabbitMQ service be deleted before an update.

If you are updating to version 2020.1.4, run the following command:
`kubectl -n name-of-namespace delete service sas-rabbitmq-server`

Delete sas-consul-server-internal Service

Note: This deployment note applies only to manual deployments of SAS Viya. If you use the SAS Viya Deployment Operator to deploy and maintain your software, the procedures described in this deployment note are performed for you, and you should skip this deployment note.

Updating to version 2020.1.4 of SAS Viya requires deleting the sas-consul-server-internal service before performing the update.

To delete the service, run the following command:
`kubectl -n name-of-namespace delete service sas-consul-server-internal`

Update annotationSelector in storageclass.yaml Reference

Before version 2020.1.4, the creation of a storageclass.yaml file and its addition to the example kustomization.yaml file was treated as optional. Beginning in version 2020.1.4, it is being treated as required. See “Specify PersistentVolumeClaims to Use ReadWriteMany StorageClass” in *SAS Viya Platform: Deployment Guide* for information about creating the storagelass.yaml file.

Additionally, the `patches` content that you are directed to add to the initial kustomization.yaml file requires a revision. The `annotationSelector` should have `sas-commonfiles` added to it. Here is an example:

```
patches:
- path: site-config/storageclass.yaml
  target:
    kind: PersistentVolumeClaim
    annotationSelector: sas.com/component-name in (sas-backup-job,sas-
data-quality-services,sas-commonfiles)
```

The example kustomization.yaml file at “Create the File” in *SAS Viya Platform: Deployment Guide* has been revised to reflect this change.

Delete Obsolete Crunchy Data Pods

Note: This deployment note applies only to manual deployments of SAS Viya. If you use the SAS Viya Deployment Operator to deploy and maintain your software, the procedures described in this deployment note are performed for you, and you should skip this deployment note.

Also, if you have an external instance of PostgreSQL, you should skip this deployment note.

The provider of the internal instance of PostgreSQL, Crunchy Data, has reorganized the pods in which the software is deployed. If you are updating to version 2020.1.4 of SAS Viya and you have an internal instance of PostgreSQL, you must delete some pods before updating your software.

Before you perform the deployment commands for your update, run the following commands:

```
kubectl -n name-of-namespace delete job -l pg-task=backup-sas-crunchy-data-postgres-pgdump
kubectl -n name-of-namespace delete job -l pg-task=restore-sas-crunchy-data-postgres-pgdump
```

If there is no pod to delete, the command has the following output:

```
No resources found
```

This is an acceptable result.

For more information about the new organization of the Crunchy Data pods, see PostgreSQL and Crunchy Data.

Delete sas-commonfiles Job

Note: This deployment note applies only to manual deployments of SAS Viya. If you use the SAS Viya Deployment Operator to deploy and maintain your software, the procedures described in this deployment note are performed for you, and you should skip this deployment note.

The addition of the sas-commonfiles component requires that any sas-commonfiles Jobs from an earlier version of SAS Viya be deleted before updating to version 2020.1.4. Run the following command to delete the Job:

```
kubectl -n name-of-namespace delete job -l app=sas-commonfiles
```

If a sas-commonfiles Job existed and was successfully deleted, the output of the command is

```
job.batch "sas-commonfiles" deleted
```

If there is no sas-commonfiles Job to be deleted, the output is

```
No resources found
```

Either output indicates success.

Scale Down the Open Distro for Elasticsearch Operator

Version 2020.1.4 contains a new version of the Open Distro for Elasticsearch Operator. Before you can update to the new version, the previous version must be scaled down.

1 Determine whether Open Distro for Elasticsearch has been deployed with the following command:

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

If a pod is returned, continue with these steps. If a pod name is not returned, take no further actions for this issue.

2 As an administrator with namespace permissions, scale down the existing internal Open Distro for Elasticsearch Operator deployment with the following command:

```
kubectl -n name-of-namespace scale deployment --replicas=0 sas-opendistro-operator
```

3 As an administrator with namespace permissions, ensure that there are no pods running the internal Open Distro for Elasticsearch operator by using the following command:

```
kubectl -n name-of-namespace wait --for delete pod --selector name=sas-opendistro-operator --timeout 180s
```

This command waits until the Open Distro for Elasticsearch Operator pod has been terminated. If the pod has already terminated, the following output is returned:

```
error: no matching resources found
```

If the command waits and the operator pod is later terminated, output like the following is returned:

```
pod/sas-opendistro-operator-6b8964d66f-7xn78 condition met
```

Enable Access Methods Through LOCKDOWN System Option

Beginning in version 2020.1.4, a LOCKDOWN system option has been added to SAS Viya. By default, the following methods cannot be used to access files and specific SAS features for a SAS session that is executing in batch mode or server processing mode:

- EMAIL
- FTP
- HADOOP
- HTTP
- PYTHON
- PYTHON_EMBED
- SOCKET
- TCPIP
- URL

To enable any of these access methods, see the README file at **$deploy/sas-bases/examples/sas-programming-environment/lockdown/README.md** (for Markdown format) or **$deploy/sas-bases/docs/lockdown_settings_for_the_sas_programming_environment.htm** (for HTML format). For more information about the LOCKDOWN system option, see "LOCKDOWN System Option" in *SAS Viya Platform: Programming Run-Time Servers*.

SAS Viya Deployment Operator Changes

In version 2020.1.4, SAS has begun emphasizing using the SAS Viya Deployment Operator to deploy your SAS Viya software. In addition to the organization changes in the documentation described in *What's New in SAS Viya Platform Operations*, SAS has also codified some of the best practices associated with deploying and using the operator. Existing deployments of the operator might already meet these guidelines, but if they do not, determine whether changes to the existing deployment should be made.

Here is a list of changes and links to where they are described in the deployment guide:

- Use the operator in only one mode, namespace or cluster-wide, in each cluster. If you select namespace mode, multiple operators can be used in the cluster. See "Overview of the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*,
- Use a unique directory for the files associated with the operator. This directory is referred to as `$operator-deploy` in the documentation. See "Retrieve the Files Required by the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide* and "Directories for the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*.
- Use a unique directory for the license files and certificates for all your deployments in a cluster. This directory is referred to as `$license` in the documentation. See "Retrieve Required Files" in *SAS Viya Platform: Deployment Guide* and "Directories for the SAS Viya Platform Deployment Operator" in *SAS Viya Platform: Deployment Guide*.
- Name the SASDeployment custom resource for each SAS Viya deployment using the $deploy directory name as a prefix (for example, viya1-sasdeployment.yaml). See "Run the create sas-deployment-cr Command" in *SAS Viya Platform: Deployment Guide*.

Python Configuration Transformer Changes

In version 2020.1.4, the python-transformer.yaml file that is used to configure Python has been modified. If you included the transformer file in the base kustomization.yaml file, you must replace the pre-2020.1.4 version with an updated version to ensure a successful update.

Perform the following steps:

1. Search the transformers block of your base kustomization.yaml file for "python-transformer.yaml". If you do not find a reference to the transformer, no action is required.
2. Otherwise, delete the **`$deploy/site-config/sas-open-source-config/python/python-transformer.yaml`** file.
3. Copy the **`$deploy/sas-bases/examples/sas-open-source-config/python/python-transformer.yaml`** file to **`$deploy/site-config/sas-open-source-config/python/python-transformer.yaml`**.
4. Perform step 3 in the "Installation" section of the README file at **`$deploy/sas-bases/examples/sas-open-source-config/python/README.md`** (for Markdown

format) or at `$deploy/sas-bases/docs/configure_python_for_sas_viya.htm` (for HTML format).

After Deployment Commands

Update CAS Servers

Version 2020.1.4 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although deleting the CAS server pods is required in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2020.1.4, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2020.1.4, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2020.1.3 (February 2021)

General Deployment

Before Deployment Commands

Configure Open Distro for Elasticsearch

Version 2020.1.3 includes Open Distro for Elasticsearch, which is an Apache 2.0-licensed distribution of Elasticsearch enhanced with enterprise security. SAS Viya includes Open Distro for Elasticsearch and uses its distributed search cluster in infrastructure and solution services.

If you are updating to version 2020.1.3, perform the following steps before the deployment commands in order to configure Open Distro for Elasticsearch.

Note: The example kustomization.yaml file, located at "Initial kustomization.yaml File" in *SAS Viya Platform: Deployment Guide*, includes these customizations.

1 Add the following line to the resources block of the base kustomization.yaml file:

```
resources:
...
- sas-bases/overlays/internal-elasticsearch
...
```

2 Add the following line to the transformers block of the base kustomization.yaml file:

```
transformers:
...
- sas-bases/overlays/internal-elasticsearch/internal-elasticsearch-transformer.yaml
...
```

3 Add a reference to the sysctl-transformers.yaml file to the transformers block of the base kustomization.yaml file. This transformer must be included after any TLS transformers and before the sas-bases/overlays/required/transformers.yaml transformer.

Note: The sysctl-transformers.yaml transformer uses a privileged container to set vm.max_map_count. If privileged containers are not allowed in your

deployment, do not add this line. Instead, the Kubernetes administrator must set the `vm.max_map_count` property for stateful workload nodes manually.

Here is an example:

```
transformers:
...
- sas-bases/overlays/network/ingress/security/transformers/...
- sas-bases/overlays/internal-elasticsearch/sysctl-transformer.yaml
- sas-bases/overlays/required/transformers.yaml
...
```

Python Configuration Changes

In version 2020.1.3, the files that are used to configure Python have been modified. If you included the transformer file in the base kustomization.yaml file, you must replace the pre-2020.1.3 version of it and the Python kustomization.yaml with updated versions to ensure a successful update.

Perform the following steps:

1. Search the transformers block of your base kustomization.yaml file for "python-transformer.yaml". If you do not find a reference to the transformer, no action is required.
2. Otherwise, delete the following files:
   - **`$deploy/site-config/sas-open-source-config/python/python-transformer.yaml`**
   - **`$deploy/site-config/sas-open-source-config/python/kustomization.yaml`**
3. Follow the steps in the README file at **`$deploy/sas-bases/examples/sas-open-source-config/python/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_python_for_sas_viya.htm`** (for HTML format).

After Deployment Commands

Update CAS Servers

Version 2020.1.3 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although deleting the CAS server pods is required in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts new pods.
- The updated CAS servers might need to be reloaded with data.

- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2020.1.3, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2020.1.3, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2020.1.2 (January 2021)

General Deployment

After Deployment Commands

Update CAS Servers

Version 2020.1.2 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment will continue to run the earlier version.

Although deleting the CAS server pods is required in order to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, drops all the in-memory data, and restarts new pods.
- The updated CAS servers might need to be reloaded with data.
- Stopping and starting the CAS servers cause an outage that will be noticeable for the end users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the existing CAS pods, which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Enter the command on a single line. Multiple lines are used here to improve readability.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2020.1.2, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2020.1.2, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

2020.1.1 (December 2020)

General Deployment

Before Deployment Commands

Configuration Instances Disabled by Default

The SAS Administrator can use SAS Environment Manager to modify the configuration for SAS Viya compute, SAS/CONNECT, SAS Batch, and CAS servers. Modifications to these servers involve updating configuration instances. Each server type has multiple instances that allow for modifications to configuration files, autoexec code, and start-up scripts that are used to launch the servers.

Prior to version 2020.1.1, these configuration instances were all enabled by default. With version 2020.1.1, configuration instances that allow start-up script modification are disabled by default.

In order to enable these configuration instances, a Kubernetes administrator must set SAS_ALLOW_ADMIN_SCRIPTS to true in the sas-shared-config configMap. The required steps are described in **`$deploy/sas-bases/overlays/sas-programming-environment/README.md`**.

For more information about the subset of configuration instances that are now disabled by default, see Edit Server Configuration Instances in *SAS Viya Administration*.

After Deployment Commands

Update CAS Servers

Version 2020.1.1 contains an update to CAS. However, until the old CAS servers are restarted, your deployment will continue to run the older version.

While deleting the CAS server pods is required to restart CAS and complete your update, note the following caveats:

- Deleting the CAS server pods stops the running CAS servers, dropping all the in-memory data, and restarts new pods.
- The updated CAS servers might need to be re-loaded with data.
- Stopping and starting the CAS servers causes an outage that will be noticeable for the end-users. Make plans accordingly.

After you have performed the commands to deploy the updates, you must delete the old CAS pods which will restart an updated CAS server. As an administrator with namespace privileges, run the following command:

Note: Because of the length of the command and the margin of the page, this command appears as more than one line. The command should be entered as a single line.

```
kubectl -n name-of-namespace delete pods -l app.kubernetes.io/managed-by=sas-cas-operator
```

SAS Deployment Operator Environment Variables Deprecated

In version 2020.1.1, the SAS Deployment Operator has deprecated two of its environment variables, `MINIMUM_UPDATE_INTERVAL` and `SAS_UPDATE_CHECK_INTERVAL`. While these variables continue to work in version 2020.1.1, SAS recommends they be replaced with a new variable `AUTOUPDATE_SCHEDULE`.

SAS/CONNECT Lockdown Default Changed

Using the LOCKDOWN System Option and the LOCKDOWN Statement, you can limit access to files and to specific features in a SAS/CONNECT server session.

Prior to version 2020.1.1, LOCKDOWN was disabled by default. But with version 2020.1.1, LOCKDOWN has been enabled by default.

Note: For more information about the LOCKDOWN system option, see "LOCKDOWN System Option" in *SAS Viya Platform: Programming Run-Time Servers*.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2020.1.1, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2020.1.1, delete the existing ESPServer custom resource using the following command:

```
kubectl -n name-of-namespace delete espserver --all
```

To verify that the custom resource was deleted:

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

The following output indicates a successful deletion:

```
No resources found in name-of-namespace namespace.
```

SAS Model Publish Service

Before Deployment Commands

Deployment Assets Directory Structure Change

In version 2020.1.1, the SAS Model Publish Service has added content for Git, causing the directory structure to change. As a result, if you are updating to version 2020.1.1 and you previously configured sas-model-publish for Kaniko, you must modify your original .yaml files.

1 Delete the following files from your directory structure:

- **$deploy/site-config/examples/sas-model-publish/podtemplate.yaml**

- **$deploy/site-config/examples/sas-model-publish/storage.yaml**

2. Remove the following entries from your base kustomization.yaml file:
   - In the resources block, remove `- site-config/sas-model-publish`
   - In the transformers block, remove `- sas-bases/overlays/sas-model-publish/transformers.yaml`
3. Follow the instructions in the new README file, located at **$deploy/sas-bases/examples/sas-model-publish/kaniko/README.md** (for Markdown format) or **$deploy/sas-bases/docs/configure_kaniko_for_sas_model_publish_service.htm** (for HTML format). The README file contains the necessary steps to copy and modify the new .yaml files and add the appropriate references to the base kustomization.yaml file.