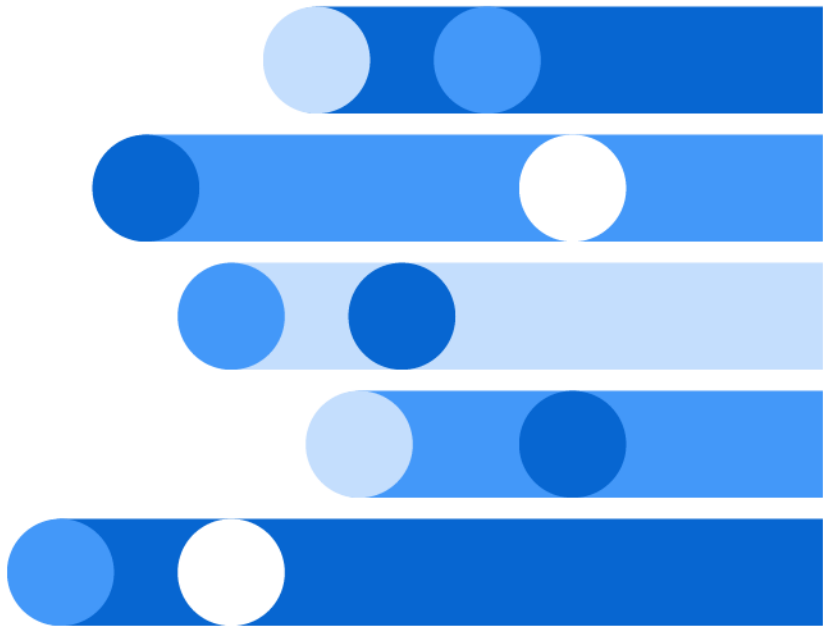
SAS® Viya® Platform: Deployment Notes

2021.1.1 - 2026.09*



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

SAS® Documentation
October 6, 2026

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

Copyright © 2021, SAS Institute Inc., Cary, NC, USA

All Rights Reserved. Produced in the United States of America.

For a hard copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication, or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a), and DFAR 227.7202-4, and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

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

October 2026

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

v_014-P1:dplynotes

Contents

Chapter 1 / ***Introduction*** **1**
Overview 1
How to Use the Deployment Notes 1

Chapter 2 / ***Long-Term Support Releases*** **3**
2026.03 (May 2026) 5
2025.09 (November 2025) 29
2025.03 (May 2025) 44
2024.09 (November 2024) 70
2024.03 (May 2024) 93
2023.10 (November 2023) 110
2023.03 (May 2023) 121
2022.09 (November 2022) 138

Chapter 3 / ***Stable Releases*** **151**
2026.09 (September 2026) 157
2026.08 (August 2026) 164
2026.07 (July 2026) 168
2026.06 (June 2026) 172
2026.05 (May 2026) 174
2026.04 (April 2026) 179
2026.03 (March 2026) 183
2026.02 (February 2026) 191
2026.01 (January 2026) 199
2025.12 (December 2025) 201
2025.11 (November 2025) 208
2025.10 (October 2025) 213
2025.09 (September 2025) 217
2025.08 (August 2025) 221
2025.07 (July 2025) 228
2025.06 (June 2025) 234
2025.05 (May 2025) 244
2025.04 (April 2025) 248
2025.03 (March 2025) 256
2025.02 (February 2025) 264
2025.01 (January 2025) 282
2024.12 (December 2024) 286
2024.11 (November 2024) 291
2024.10 (October 2024) 297
2024.09 (September 2024) 303
2024.08 (August 2024) 311
2024.07 (July 2024) 314
2024.06 (June 2024) 319
2024.05 (May 2024) 322

2024.04 (April 2024) 336
2024.03 (March 2024) 341
2024.02 (February 2024) 351
2024.01 (January 2024) 359
2023.12 (December 2023) 363
2023.11 (November 2023) 368
2023.10 (October 2023) 373
2023.09 (September 2023) 380
2023.08 (August 2023) 384
2023.07 (July 2023) 388
2023.06 (June 2023) 391
2023.05 (May 2023) 395
2023.04 (April 2023) 397
2023.03 (March 2023) 403
2023.02 (February 2023) 409
2023.01 (January 2023) 412
2022.12 (December 2022) 415
2022.11 (November 2022) 418
2022.10 (October 2022) 422
2022.09 (September 2022) 435
2022.1.4 (August 2022) 438
2022.1.3 (July 2022) 442
2022.1.2 (June 2022) 448
2022.1.1 (May 2022) 449
2021.2.6 (April 2022) 453
2021.2.5 (March 2022) 457
2021.2.4 (February 2022) 458
2021.2.3 (January 2022) 461
2021.2.2 (December 2021) 464
2021.2.1 (November 2021) 467
2021.1.6 (October 2021) 470
2021.1.5 (September 2021) 474
2021.1.4 (August 2021) 476
2021.1.3 (July 2021) 484
2021.1.2 (June 2021) 489
2021.1.1 (May 2021) 493
2020.1.5 (April 2021) 495
2020.1.4 (March 2021) 497
2020.1.3 (February 2021) 502
2020.1.2 (January 2021) 505
2020.1.1 (December 2020) 506

1

Introduction

Overview 1
How to Use the Deployment Notes 1

Overview

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 Tasks*.

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 2023.04 and you want to update to 2023.07, review the deployment notes for 2023.05, 2023.06, and 2023.07.

2

Long-Term Support Releases

2026.03 (May 2026) **5**
Additional Documentation 5
General Deployment 5
SAS Decisions Runtime Builder 22
SAS Detection Architecture 24
SAS Model Publish Service 25
SAS RFC Solution Configuration 26
SAS Risk Cirrus KRM Service 27
SAS Studio 28
SAS Visual Investigator 29

2025.09 (November 2025) **29**
Additional Documentation 29
General Deployment 30
Model Studio 36
SAS Data Quality 37
SAS Decisions Runtime Builder 38
SAS Detection Architecture 40
SAS Micro Analytic Service 41
SAS Model Publish Service 42
SAS SpeedyStore 43

2025.03 (May 2025) **44**
Additional Documentation 44
General Deployment 45
SAS Allowance for Credit Loss 52
SAS Asset and Liability Management 53
SAS Dynamic Actuarial Modeling 54
SAS Event Stream Processing 55
SAS Expected Credit Loss 57
SAS Insurance Capital Management 58
SAS Insurance Contract Valuation Foundation 59
SAS Integrated Regulatory Reporting 60
SAS Model Repository Service 61
SAS Model Risk Management 62
SAS Regulatory Capital Management 63
SAS Risk Cirrus Core 64
SAS Risk Factor Manager 65
SAS Risk Modeling 66
SAS Risk Offerings 67

SAS Stress Testing ... 68
SAS/ACCESS to Impala ... 68
SAS/ACCESS to Spark ... 69

2024.09 (November 2024) ... **70**
Additional Documentation ... 70
General Deployment ... 70
OpenSearch ... 78
SAS Allowance for Credit Loss ... 79
SAS Anti-Money Laundering ... 80
SAS Asset and Liability Management ... 80
SAS Dynamic Actuarial Modeling ... 81
SAS Event Stream Processing ... 82
SAS Expected Credit Loss ... 84
SAS Insurance Capital Management ... 85
SAS Model Risk Management ... 86
SAS Regulatory Capital Management ... 87
SAS RFC Solution Configuration ... 88
SAS Risk Cirrus Core ... 89
SAS Risk Factor Manager ... 90
SAS Risk Modeling ... 91
SAS Stress Testing ... 92

2024.03 (May 2024) ... **93**
General Deployment ... 93
Analytics Resources ... 96
OpenSearch ... 99
SAS Allowance for Credit Loss ... 99
SAS Anti-Money Laundering ... 100
SAS Climate Risk Stress Testing ... 100
SAS Conversation Designer ... 101
SAS Credit Risk Stress Testing ... 101
SAS Event Stream Processing ... 102
SAS Expected Credit Loss ... 104
SAS Insurance Capital Management ... 104
SAS Insurance Contract Valuation Foundation ... 105
SAS Intelligent Planning ... 106
SAS Regulatory Capital Management ... 106
SAS Stress Testing ... 107
SAS Viya with SingleStore ... 108
SAS/ACCESS Interface to MongoDB ... 109
SAS/ACCESS Interface to MySQL ... 109

2023.10 (November 2023) ... **110**
General Deployment ... 110
SAS Conversation Designer ... 116
SAS Dynamic Actuarial Modeling ... 116
SAS Event Stream Processing ... 118
SAS Intelligent Planning ... 118
SAS Risk Modeling ... 119

2023.03 (May 2023) ... **121**
General Deployment ... 121
SAS Dynamic Actuarial Modeling ... 133
SAS Event Stream Processing ... 133
SAS Inventory Optimization ... 135

SAS Model Risk Management ... 135
SAS Risk Engine ... 137

2022.09 (November 2022) ... **138**
General Deployment ... 138
SAS Visual Analytics ... 147
SAS Event Stream Processing ... 147

2026.03 (May 2026)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

New Architecture for SAS Workload Orchestrator

Previously, the manager and server Kubernetes objects for SAS Workload Orchestrator have been delivered as a single entity. With this release they are being delivered as separate entities, namely, sas-workload-orchestrator for the manager StatefulSet and sas-workload-orchestrator-server for the server DaemonSet. This change requires new configuration files. In addition, the existing files used to configure SAS Workload Orchestrator have been revised to use the new

configuration file guidelines and format. The new format will simplify future updates.

To find and use the new configuration files, follow the instructions at "Configure SAS Workload Orchestrator" in *SAS Viya Platform: Deployment Guide*. Here is a summary of the most common changes to be made:

> **IMPORTANT** Even if you have disabled your existing deployment of SAS Workload Orchestrator, you must revise the configuration files for the new architecture in order that SAS Workload Orchestrator remains disabled.

1 If SAS Workload Orchestrator is disabled in your existing deployment and you want to keep it disabled, you must disable it with the new configuration files.

   a Confirm that SAS Workload Orchestrator is disabled by searching for the following line in the base kustomization.yaml file (**`$deploy/kustomization.yaml`**):

   ```
   - sas-bases/overlays/sas-workload-orchestrator/enable-disable/sas-workload-orchestrator-disable-patch-transformer.yaml
   ```

   If the line is present, continue with Step 1b. If the line is not present, skip to Step 2.

   b Add the following line to the base kustomization.yaml file immediately after the line you just searched for.

   ```
   - sas-bases/overlays/sas-workload-orchestrator-server/enable-disable/sas-workload-orchestrator-server-disable-patch-transformer.yaml
   ```

   After you revise the base kustomization.yaml file, there are no further steps. You should skip the rest of this deployment note.

2 Determine if your current deployment configured cluster-roles.

   a Search the base kustomization.yaml file for the following line:

   ```
   - sas-bases/overlays/sas-workload-orchestrator
   ```

   If the line is present, continue with Step 2b. If the line is not present skip to Step 3.

   b In the base kustomization.yaml file, delete the following line:

   ```
   - sas-bases/overlays/sas-workload-orchestrator
   ```

   In the place of the line you just deleted, insert the following two lines:

   ```
   - sas-bases/overlays/sas-workload-orchestrator/cluster-role
   ```

   ```
   - sas-bases/overlays/sas-workload-orchestrator-server/cluster-role
   ```

3 Search for the presence of a configuration for SAS Workload Orchestrator pod requests and limits:

   a In the base kustomization.yaml file, search for the following string:

   ```
   sas-workload-orchestrator-statefulset-resources.yaml
   ```

If the string is present, note the location of the file and continue with this step. If it is not present, skip to Step 4.

b Copy the **/$deploy/sas-bases/examples/sas-workload-orchestrator/configure/configuration.env** file to your site-config directory as **/$deploy/site-config/sas-workload-orchestrator/configure/configuration.env**.

c In the `Statefulset (manager) pod resource configuration` section of the copied **/sas-workload-orchestrator/configure/configuration.env** file, modify the entries to match your current configuration. For each entry you want to modify, uncomment the line by removing the hashtag and then replacing the variable with the appropriate value. You can use the file in the location that you noted in Step 3a as a guide for the values used in the existing deployment.

d In the base kustomization.yaml file, delete the line in which you found the string you searched for in Step 3a. SAS recommends that you also delete the file described by that line since you have copied the information to the new configuration file.

4 Search for the presence of a custom initial SAS Workload Orchestrator configuration:

a In the base kustomization.yaml file, search for the following string:

```
sas-workload-orchestrator-initial-configuration-change.yaml
```

If the string is present, note the location of the file and continue with this step. If it is not present, skip to Step 5.

b If you have not already, copy the **/$deploy/sas-bases/examples/sas-workload-orchestrator/configure/configuration.env** file to your site-config directory as **/$deploy/site-config/sas-workload-orchestrator/configure/configuration.env**.

c In the `Initial Grid Configuration` section of the copied **/sas-workload-orchestrator/configure/configuration.env** file, uncomment the `initial-configuration-json` property and modify the value to match what is in the file from Step 4a.

d In the base kustomization.yaml file, delete the line in which you found the string you searched for in Step 4a. SAS recommends that you also delete the file described by that line since you have copied the information to the new configuration file.

5 If you modifed the configuration.env file in Step 3 or Step 4, continue with this step. If you did not modify the configuration.env file, skip to Step 6.

a In the base kustomization.yaml file, add a merge behavior to merge in the settings. Here is an example:

```
configMapGenerator:
...
- behavior: merge
  envs:
```

```
  - /site-config/sas-workload-orchestrator/configure/
configuration.env
  name: sas-workload-orchestrator-user-configuration
```

b Also in the base kustomization.yaml file, add a reference to the **/$deploy/sas-bases/overlays/sas-workload-orchestrator/configure/transformers.yaml** to the transformers block. Here is an example:

```
transformers:
...
- sas-bases/overlays/sas-workload-orchestrator/configure/
transformers.yaml
```

6 Search for the presence of a configuration for SAS Workload Orchestrator pod requests and limits:

a In the base kustomization.yaml file, search for the following string:

```
sas-workload-orchestrator-daemonset-resources.yaml
```

If the string is present, note the location of the file and continue with this step. If it is not present, skip to Step 7.

b Copy the **/$deploy/sas-bases/examples/sas-workload-orchestrator-server/configure/configuration.env** file to your site-config directory as **/$deploy/site-config/sas-workload-orchestrator-server/configure/configuration.env**.

c In the `Daemonset (server) pod resource configuration` section of the copied **/sas-workload-orchestrator-server/configure/configuration.env** file, modify the entries to match your current configuration. For each entry you want to modify, uncomment the line by removing the hashtag and then replacing the variable with the appropriate value. You can use the file in the location that you noted in Step 6a as a guide for the values used in the existing deployment.

d In the base kustomization.yaml file, delete the line in which you found the string you searched for in Step 6a. SAS recommends that you also delete the file described by that line since you have copied the information to the new configuration file.

7 Search for the presence of a custom initial custom Prometheus Pushgateway:

a In the base kustomization.yaml file, search for the following string:

```
sas-workload-orchestrator-prometheus-push-gateway-url.yaml
```

If the string is present, note the location of the file and continue with this step. If it is not present, skip to Step 8.

b If you have not already, copy the **/$deploy/sas-bases/examples/sas-workload-orchestrator-server/configure/configuration.env** file to your site-config directory as **/$deploy/site-config/sas-workload-orchestrator-server/configure/configuration.env**.

c In the `Initial Grid Configuration` section of the copied **/sas-workload-orchestrator-server/configure/configuration.env** file, uncomment the `prometheus-push-gateway-url` property and modify the value to match what is in the file from Step 7a.

d In the base kustomization.yaml file, delete the line in which you found the string you searched for in Step 7a. SAS recommends that you also delete the file described by that line since you have copied the information to the new configuration file.

8 If you modifed the configuration.env file in Step 6 or Step 7, continue with this step. If you did not modify the configuration.env file, skip to Step 9.

a In the base kustomization.yaml file, add a merge behavior to merge in the settings. Here is an example:

```
configMapGenerator:
...
- behavior: merge
  envs:
  - /site-config/sas-workload-orchestrator-server/configure/
configuration.env
  name: sas-workload-orchestrator-server-user-configuration
```

b Also in the base kustomization.yaml file, add a reference to the **/$deploy/sas-bases/overlays/sas-workload-orchestrator-server/configure/transformers.yaml** to the transformers block. Here is an example:

```
transformers:
...
- sas-bases/overlays/sas-workload-orchestrator-server/configure/
transformers.yaml
```

9 Determine if your current deployment has a user-defined script volume for the server pods.

a In the base kustomization.yaml file, search for the following string:

```
sas-workload-orchestrator-host-user-defined-resources-script-storage.yaml
```

If the string is present, note the location of the file and continue with this step. If it is not present, you have completed this deployment note.

b Copy the **$deploy/sas-bases/examples/sas-workload-orchestrator-server/configure/sas-workload-orchestrator-server-host-user-defined-resources-script-storage.yaml** file to **/$deploy/site-config/sas-workload-orchestrator-server/configure/sas-workload-orchestrator-server-host-user-defined-resources-script-storage.yaml**.

c In the copied file, replace the {{ NFS-SERVER-ADDR }} variable with the fully-qualified domain name of the server and replace the {{ NFS-SERVER-PATH }} variable with the path to the volume on the server. You can use the file in Step 9a as a guide for the infromation to insert into the file. Here is an example:

```
- op: replace
  path: /spec/template/spec/volumes/0
  value:
    name: scripts
    nfs:
      path: /path/to/my/scripts
      server: my.nfs.server.mydomain.com
```

For more information about this configuration, see the "User-Defined Scripts Volume for Server Pods" section of the README file located at **$deploy/sas-bases/examples/sas-workload-orchestrator-server/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_workload_orchestrator_server_service_examples.htm** (for HTML format).

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

```
- site-config/sas-workload-orchestrator-server/configure/sas-workload-orchestrator-server-host-user-defined-resources-script-storage.yaml
```

Here is an example:

```
transformers:
...
- site-config/sas-workload-orchestrator-server/configure/sas-workload-orchestrator-server
```

e In the base kustomization.yaml file, delete the line in which you found the string you searched for in Step 9a. SAS recommends that you also delete the file described by that line since you have copied the information to the new configuration file.

Upgrade to OpenSearch 3.x

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

SAS is upgrading the version of OpenSearch used in the internal instances of OpenSearch to version 3.3.2. OpenSearch 3.x contains a breaking change: indices created under Elasticsearch 7.x or OpenSearch 1.x are not compatible with OpenSearch 3.x and must be reindexed before the upgrade can proceed. The breaking change can affect deployments that were initially created on LTS 2023.03 or Stable 2023.05 or earlier, and which have not since reindexed application data. This deployment note contains the necessary steps to identify and remediate incompatible indices so that the OpenSearch upgrade can proceed.

Create List of Incompatible Indices

1 Determine the name of the pods that have OpenSearch running on them:

```
kubectl get pods -l app=sas-opendistro -n name-of-namespace
```

Save the list of pods for Step 3.

2 Retrieve the sas-opendistro-sasadmin-secret Kubernetes password:

```
kubectl -n name-of-namespace get secret sas-opendistro-sasadmin-secret -o json |jq -r .data.password| base64 -d
```

Save the password for Step 3.

3 Query the OpenSearch settings API for indices with created versions that begin with "1." or "7.".

- Perform the following command on deployments using full-stack TLS:

```
kubectl exec name-of-OpenDistro-pod -n name-of-namespace -- \
curl -k -s "https://localhost:9200/*/_settings?human" \
-u sasadmin:admin-password | \
jq 'to_entries[] | select((.key | startswith(".") | not) and
((.value.settings.index.version.created_string // "") | test("^(7|
1)\\."))) | .key'
```

- Perform the following command on deployments using front-door TLS or no TLS:

```
kubectl exec name-of-OpenDistro-pod -n name-of-namespace -- \
curl -k -s "http://localhost:9200/*/_settings?human" \
-u sasadmin:admin-password | \
jq 'to_entries[] | select((.key | startswith(".") | not) and
((.value.settings.index.version.created_string // "") | test("^(7|
1)\\."))) | .key'
```

In the command that you choose to use, replace *name-of-OpenDistro-pod* with the name of any OpenDistro pod that is in the output of the command in Step 1. Also, replace *admin-password* with the password found in Step 2.

4 The result of performing either of the commands is a list of indices. In that list, the following kinds of indices may be ignored:

- OpenSearch system indices prefixed by a period, such as `.opendistro_security`
- Indices that begin with `sand__entity_link__` or `sand__shape__`

If no indices or no other kind of indices are returned, no manual action is required and you should skip the rest of this deployment note. If other indices are returned, continue with the next section.

Reindex Incompatible Data Indices

1 The method for reindexing varies based on the SAS product. The associated product for an index is identified by its prefix. Use the appropriate method for each indexed identified in Table 2.1.

Table 2.1 *Indices and Their Reindexing Method*

Index Prefix	Associated Product	Note	Reindexing Method
`sand__`	SAS Visual Investigator	`sand__entity__link__` and `sand__shape__` indices are reindexed by the OpenSearch operator during the upgrade. Any other index that begins with `sand__` should use the described	See "Re-index All Entities" in *SAS Visual Investigator: Administrator's Guide*.

Index Prefix	Associated Product	Note	Reindexing Method
		reindexing method.	
`viya_catalog__`	SAS Data Governance (formerly known as SAS Information Catalog before 2026.06)		When you restart sas-catalog-services, re-create the search indices. The associated search indices should be rebuilt to ensure that they are fully populated. See the steps described in the table at "Respond to Notifications" in *SAS Data Governance: Administrator's Guide*.
`security-auditlog-` and `top_queries-`	OpenSearch Log Indices	By default, `security-auditlog-` and `top_queries-` indices are removed after seven days. However, a transformer can be applied to retain security auditlogs for longer. If your deployment uses this transformer and audit logs are present, logs created with OpenSearch 1.x must be removed or reindexed.	See Reindex data and Delete Index API.
`viya_cirrus__`	SAS Risk Cirrus	No version 1.x `viya_cirrus__` indices should be present. If you see these indices in	See "Get Support" in *Getting Started with SAS Viya Platform Operations*.

Index Prefix	Associated Product	Note	Reindexing Method
		the output, contact SAS Technical Support.	

2 If you are unsure how a product-specific index should be rebuilt, consult the product documentation or contact SAS Technical Support.

Validate That No Incompatible Indices Remain

1 Re-run the query from "Create List of Incompatible Indices" to confirm that no application data indices created under Elasticsearch 7.x or OpenSearch 1.x remain.

2 After you have confirmed that all application data indices have been reindexed, perform the deployment commands to update your software and then perform the tasks described at "OpenSearch 3 Post-Upgrade Validation" on page 22.

Revised odbc.ini File

Updates to SAS/ACCESS Interface to Amazon Redshift, SAS/ACCESS Interface to Greenplum, and SAS/ACCESS Interface to Microsoft SQL Server require the odbc.ini file in a deployment that includes those products to be revised. Here are the instructions to determine if your deployment is affected and to revise the file.

1 Perform the following command:

```
%put %quote(%sysget(ODBCINI));
```

If you get no output, you do not use the odbc.ini file in your deployment and you can continue to the next deployment note. If you get output, continue with Step 2.

2 Go to the location described in the output of the command from Step 1. Open the odbc.ini file there and delete the following entries:

- `SSLLibname`
- `CryptoLibname`

SAS/CONNECT Spawner Requires SCC for Red Hat OpenShift

Note: If your deployment is not running on Red Hat OpenShift, skip this deployment note.

All deployments on Red Hat OpenShift require binding and applying a security context constraint (SCC). For more information, including the required commands, see "Security Context Constraints and Service Accounts" in *SAS Viya Platform: Deployment Guide*.

New Configuration File Format for Backup and Restore

The configuration files for backup, restore, and migration have been revised. The new format simplifies future updates and the maintenance of backup and restore configurations.

Use the following steps to update the reformatted files.

1 In the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the following string:

```
sas-backup-pv-copy-cleanup-job-modify-resources-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 2.

a Note the location for the file described in the kustomization.yaml entry.

b Copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the values for the {{ PV-COPY-CLEANUP-CPU-LIMIT }} and {{ PV-COPY-CLEANUP-MEMORY-LIMIT }} variables. You can use the file you found in Step 1 as a guide for the values to use in the copied file.

d In the base kustomization.yaml file, add the following content to the configMapGenerator block:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e Add the following content to the transformers block of the base kustomization.yaml file:

```
transformers:
...
- sas-bases/overlays/backup/sas-backup-pv-copy-cleanup-job-modify-resources-transformer.yaml
...
```

f Delete the file described by the line you found in Step 1. In the base kustomization.yaml file, remove the line with the string that you found in Step 1.

2 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-scheduled-backup-all-sources-change-default-schedule
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 3.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the value for the {{ SCHEDULE-BACKUP-ALL-SOURCES-CRON-EXPRESSION }} variable. You can use the file you found in Step 2 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-scheduled-backup-all-sources-change-default-schedule.yaml
...
```

f Delete the file described by the line you found in Step 2. In the base kustomization.yaml file, remove the line with the string that you found in Step 2.

3 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-scheduled-backup-incr-job-change-default-schedule
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 4.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the value for the {{ SCHEDULE-BACKUP-INCR-CRON-EXPRESSION }} variable. You can use the file you found in Step 3 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-scheduled-backup-incr-job-change-default-schedule.yaml
...
```

f Delete the file described by the line you found in Step 3. In the base kustomization.yaml file, remove the line with the string that you found in Step 3.

4 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-backup-job-modify-resources-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 5.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the values for the {{ BACKUP-JOB-CPU-LIMIT }} and { BACKUP-JOB-MEMORY-LIMIT }} variables. You can use the file you found in Step 4 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-scheduled-backup-job-modify-resources-transformer.yaml
...
```

f Delete the file described by the line you found in Step 4. In the base kustomization.yaml file, remove the line with the string that you found in Step 4.

5 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-common-backup-data-storage-size-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 6.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the value for the {{ STORAGE-SIZE-IN-GB }} variable. You can use the file you found in Step 5 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-common-backup-data-storage-size-
transformer.yaml
...
```

f Delete the file described by the line you found in Step 5. In the base kustomization.yaml file, remove the line with the string that you found in Step 5.

6 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-common-backup-data-storage-class-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 7.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the value for the {{ STORAGE-CLASS-NAME }} variable. You can use the file you found in Step 6 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
```

```
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-common-backup-data-storage-class-
transformer.yaml
...
```

f Delete the file described by the line you found in Step 6. In the base kustomization.yaml file, remove the line with the string that you found in Step 6.

7 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-scheduled-backup-job-change-default-backup-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 8.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the value for the {{ SCHEDULE-BACKUP-CRON-EXPRESSION }} variable. You can use the file you found in Step 7 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-scheduled-backup-job-change-default-
backup-transformer.yaml
...
```

f Delete the file described by the line you found in Step 7. In the base kustomization.yaml file, remove the line with the string that you found in Step 7.

8 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-restore-job-modify-resources-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 9.

a Note the location for the file described in the kustomization.yaml entry.

b Copy the **$deploy/sas-bases/examples/restore/sas-restore-config.env** to **$deploy/site-config/restore/sas-restore-config.env**.

c In the copied file, update the values for the {{ RESTORE-JOB-CPU-LIMIT }} and {{ RESTORE-JOB-MEMORY-LIMIT }} variables. You can use the file you found in Step 8 as a guide for the values to use in the copied file.

d In the base kustomization.yaml file, add the following content to the configMapGenerator block:

```
configMapGenerator:
...
- name: sas-restore-config
  envs:
    - site-config/restore/sas-restore-config.env
...
```

e Add the following content to the transformers block of the base kustomization.yaml file:

```
transformers:
...
- sas-bases/overlays/restore/sas-restore-job-modify-resources-
transformer.yaml
...
```

f Delete the file described by the line you found in Step 8. In the base kustomization.yaml file, remove the line with the string that you found in Step 8.

9 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-migration-job-modify-resources-transformer
```

If the string is present, continue with the following substeps. If it is not present, you have completed this deployment note.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/restore/sas-restore-config.env** to **$deploy/site-config/restore/sas-restore-config.env**.

c In the copied file, update the values for the {{ MIGRATION-JOB-CPU-LIMIT }} and {{ MIGRATION-JOB-MEMORY-LIMIT }} variables. You can use the file you found in Step 9 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
```

```
- name: sas-restore-config
  envs:
    - site-config/restore/sas-restore-config.env
...
```

e Add the following content to the transformers block of the base kustomization.yaml file:

```
transformers:
...
- sas-bases/overlays/migration/sas-migration-job-modify-resources-
transformer.yaml
...
```

f Delete the file described by the line you found in Step 9. In the base kustomization.yaml file, remove the line with the string that you found in Step 9.

Revised SAS Configurator for Open Source Transformer

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised to manage the list of supported Python versions. The transformer is 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 at **$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 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.

5 After your revisions are complete, delete the change-configuration2.yaml file.

For more information about the change-configuration.yaml file, see "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

New PROC R Requires Revisions to Resource File

SAS is introducing PROC R for use with the R external language. To enable PROC R, you must revise a resource file before you update your SAS Viya platform deployment.

1 In the resources block of your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the following string:

```
- site-config/sas-open-source-config/r
```

If that string is not present, no further steps are required. If the string is present, continue with Step 2.

2 Go to **$deploy/site-config/sas-open-source-config/r** and change the kustomization.yaml file name to `kustomization2.yaml`.

3 Copy the file at **$deploy/sas-bases/examples/sas-open-source-config/r/kustomization.yaml** to **$deploy/site-config/sas-open-source-config/r**. You should now have a kustomization.yaml file and a kustomization2.yaml file in the same directory.

4 Compare the kustomization.yaml and kustomization2.yaml files. Use the kustomization2.yaml file as a guide for the revisions to be made to the kustomization.yaml file in order to match the configuration settings used in your existing deployment.

5 After your revisions are complete, delete the kustomization2.yaml file.

For more information about PROC R and configuring for it, see "Configure R Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

OpenSearch 3 Post-Upgrade Validation

Verify that the OpenSearch upgrade has completed successfully:
`kubectl get opendistrocluster -n name-of-namespace`

In the output of the command, the`UPGRADE-STATUS` column should display `up-to-date` and the `SEARCH-ENGINE-VERSION` column should display `opensearch 3.x`. Here is an example:

```
NAME             UPGRADE-STATUS   SEARCH-ENGINE-VERSION
sas-opendistro   up-to-date       opensearch 3.x
```

If the output displays `upgrade blocked (upgrade requires data reindex)`, incompatible indices still remain. Go to "Create List of Incompatible Indices" on page 10 to identify and reindex the remaining indices. The upgrade will resume automatically when all incompatible indices have been corrected.

SAS Decisions Runtime Builder

Before Deployment Commands

SAS Decisions Runtime Builder Service Merging with SAS Model Publish Service

The SAS Decisions Runtime Builder service is merging with the SAS Model Publish service. As a result, any references to SAS Decisions Runtime Builder in the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) must be revised to refer to SAS Model Publish in order to deploy BuildKit with SAS Model Publish. Furthermore, BuildKit deployments pull base layers from source registries and publish container images to target registries. If the source registry that contains SAS Viya platform deployment images or the destination registry uses self-signed certificates, then the certificates should be added to the BuildKit deployment via the certificate framework.

Note: Steps 4 through 8 duplicate steps that are included in the "Incorporating Additional CA Certificates" section of README file located 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). For more information, see that README.

1 In your deployment assets, search for the following directory:

```
$deploy/site-config/sas-decisions-runtime-builder
```

If the directory is present, continue with Step 2. If it is not present, skip the rest of this deployment note.

2 Configure BuildKit for use with the SAS Model Publish service by following the instructions in the README file located at **$deploy/sas-bases/examples/sas-model-publish/buildkit/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_buildkit_for_sas_model_publish_service.htm** (for HTML format).

Note: SAS recommends referring to the **$deploy/site-config/sas-decisions-runtime-builder/buildkit/storage.yaml** file while updating the new configuration.env file to ensure that the values used in your current deployment are used in the update.

3 In the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the following line:

```
- site-config/sas-decisions-runtime-builder/buildkit/certs
```

If the line is present, continue with Step 4. If it is not present, skip to Step 9.

4 Look in your deployment assets for the following directory:

```
$deploy/site-config/security/cacerts
```

If the directory does not exist, create it.

5 Copy the contents of **$deploy/site-config/sas-decisions-runtime-builder/buildkit/certs** to **$deploy/site-config/security/cacerts**.

6 Copy the **$deploy/sas-bases/examples/security/customer-provided-ca-certificates.yaml** file to **$deploy/site-config/security/**

7 Edit the **$deploy/site-config/security/customer-provided-ca-certificates.yaml** to include the names of the CA certificates.

8 Edit the base kustomization.yaml file to add the following line in the generators block:

```
- site-config/security/customer-provided-ca-certificates.yaml
```

9 Also in the base kustomization.yaml file, delete the following lines:

- In the resources block, delete

```
- site-config/sas-decisions-runtime-builder/buildkit/certs
```

```
- site-config/sas-decisions-runtime-builder/buildkit
```

- In the transformers block, delete

```
- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/
buildkit-certificates-transformer.yaml
```

```
- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/buildkit-transformer.yaml
```

10 Delete the **$deploy/site-config/sas-decisions-runtime-builder** directory from your deployment assets.

SAS Detection Architecture

Before Deployment Commands

New Configuration File Format

The files used to configure SAS Detection Architecture have been revised. The new format will simplify future updates.

1 In your deployment assets, search for the following subdirectory:

```
$deploy/site-config/sas-detection/
```

If the subdirectory is not present, you are not affected by the change and should skip the rest of this deployment note. If the subdirectory is present, continue with Step 2.

2 To use the new configuration files, follow the steps described in the README file at **$deploy/sas-bases/examples/sas-detection/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_detection_engine_configuration.htm** (for HTML format).

Note: SAS recommends referring to the **$deploy/site-config/sas-detection/detection-engine-deployment.yaml** file while updating the new configuration.env file. Using the detection-engine-deployment.yaml file ensures that the values used in your current deployment are used in the update.

3 After configuration.env has been revised, delete **$deploy/site-config/sas-detection/detection-engine-deployment.yaml** because it is no longer used in the deployment.

SAS Model Publish Service

Before Deployment Commands

BuildKit Adding Default Restrictions for Sizing

SAS Model Publish is adding default restrictions to the number of CPUs, the amount of memory, and the number of replicas for BuildKit. To use the new defaults:

1 In the resoures block of your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), look for the following line:

```
- site-config/sas-model-publish/buildkit
```

If the line is not present, you are not affected by the change and should skip the rest of the deployment note. If the line is present, continue with Step 2.

2 Rename the **`$deploy/site-config/sas-model-publish/buildkit/configuration.env`** file `configuration2.env`.

3 Copy the **`$deploy/sas-bases/examples/sas-model-publish/buildkit/configuration.env`** file to **`$deploy/site-config/sas-model-publish/buildkit/configuration.env`**. The **`$deploy/site-config/sas-model-publish/buildkit/`** directory should now contain a configuration.env and a configuration2.env file.

4 Modify the configuration.env file so that it contains the values you want for your deployment.

> **IMPORTANT** You must specify the same value for `buildkitStorageSize` and `buildkitStorageClass` that was used in the previous configuration. Use the configuration.env2 file as a guide.

If you do not change the values in configuration.env, they will be the following:

Table 2.2 *Default Restrictions for BuildKit*

Minimum CPU	1 core
Minimum memory	1Gi
Maximum memory	1Gi
Maximum replicas	1 replica

Note: For more information about the configuration.env file, see the README file located at **$deploy/sas-bases/examples/sas-model-publish/buildkit/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_buildkit_for_sas_model_publish_service.htm** (for HTML format).

5 After the configuration.env file has been modified, delete the configuration2.env file.

6 Copy **$deploy/sas-bases/examples/sas-model-publish/buildkit/kustomization.yaml** file to **$deploy/site-config/sas-model-publish/buildkit/kustomization.yaml**.

SAS RFC Solution Configuration

Before Deployment Commands

Reorganized Directory Structure

In 2026.03, SAS RFC Solution adds new functionality to its workloads that requires that the deployment assets for the product be reorganized. This note describes how to modify an existing deployment of SAS RFC Solution to accommodate the new functionality.

1 To determine if you have SAS RFC Solution workloads in your deployment, go to the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**). Search for the following line:

```
- site-config/sas-rfc-solution-workloads/install
```

If that line is present, then you have SAS RFC Solution workloads in your deployment and should continue with Step 2. If the line is not present, skip the rest of this deployment note.

2 To configure SAS RFC Solution workloads with the new configuration files, follow the instructions in the README file located at **$deploy/sas-bases/examples/sas-rfc-solution-workloads/ README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configuring_sas_rfc_solution_workloads.htm** (for HTML format).

Note: After you copy the configuration files to the **$deploy/site-config** directory, you will have two sets of nearly identical configuration files. The existing deployment uses the configuration files located at **$deploy/site-config/sas-rfc-solution-workloads/install**. The updated deployment will

use the configuration files located at **$deploy/site-config/sas-rfc-solution-workloads/install/main**.

The instructions in the README include modifying the files that you move to the **$deploy/site-config/sas-rfc-solution-workloads/install/main** directory. Because the old versions of the configuration files can still be found at **$deploy/site-config/sas-rfc-solution-workloads/install**, SAS recommends that you use the old version as a basis for any changes you make to the new version of a file. This ensures that the configuration properties in the updated deployment are the same as in the existing deployment.

For example, assume that the README instructs you to "Update the **$deploy/site-config/sas-rfc-solution-workloads/install/main/namespace.yaml** file by replacing the {{ NAMESPACE }} value." You can find the value for the {{ NAMESPACE }} variable in that file by referring to the **$deploy/site-config/sas-rfc-solution-workloads/install/namespace.yaml** file.

3 To prevent potential confusion in the future, SAS recommends that you delete the old versions of the files. Delete the files in **$deploy/sas-bases/examples/sas-rfc-solution-workloads/install** and **$deploy/site-config/sas-rfc-solution-workloads/install**.

SAS Risk Cirrus KRM Service

Before Deployment Commands

New Configuration File Format

The SAS Risk Cirrus KRM service provides a REST API for starting and managing KRM runs associated with Risk Cirrus analysis runs and comes with default settings that may be changed. The files used to configure SAS Risk Cirrus KRM Service have been revised. The new format will simplify future updates.

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

```
- site-config/sas-risk-cirrus-krm/resources/krm_transform.yaml
```

If the line is not present, you are not affected by the change and should skip the rest of the deployment note. If the line is present, continue with Step 2.

2 To use the new configuration files, follow the steps described in the README file at **$deploy/sas-bases/examples/sas-risk-cirrus-krm/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_the_sas_risk_cirrus_krm_service.htm** (for HTML format).

Note: SAS recommends referring to the **$deploy/site-config/sas-risk-cirrus-krm/resources/krm_transform.yaml** file while updating the new configuration.env file to ensure that the values used in your current deployment are used in the update.

3 The new configuration of the files means that the properties and values used in the old krm_transform.yaml file are now in the configuration.env file. Because the file is no longer needed, delete **$deploy/site-config/site-config/sas-risk-cirrus-krm/resources/krm_transform.yaml** from your deployment assets. Also, remove the reference to it from the transformers block of the base kustomization.yaml file.

SAS Studio

Before Deployment Commands

SSH Authentication with Git Requires New File

If you are leveraging SSH authentication with Git, you must include a `known_hosts` file in the **<userhome>/.ssh** directory along with your public and private SSH key files. If you are uncertain whether your deployment is leveraging SSH authentication with git, go to SAS Environment Manager, select the **Configuration** page, and look in the studio service for the `allowGitSSHPassword` property. If that property is not present or if it is present but not set to `true`, then you should skip the rest of this deployment note.

You can create a `known_hosts` file by scanning your SSH keys with the following command from either a UNIX or Linux terminal or with Git Bash:

```
ssh-keyscan <remote repo server> >> ~/.ssh/known_hosts
```

Here is an example:

```
ssh-keyscan github.com >> ~/.ssh/known_hosts
```

Upload your SSH key files and `known_hosts` file to the server file system by selecting the appropriate directory in the SAS Server section of the navigation pane and clicking . Click + and browse for the SSH key and `known_hosts` files. Select the files and click **Open**. Click **Upload** to upload the files to the directory on your SAS server.

To display hidden files and folders in the **SAS Server** and **SAS Content** sections of the navigation pane, click the user button on the application bar and select **Settings**. From the SAS Studio application settings, select **General**, and then select the **Display hidden folders and items on the file system** option.

Additionally, your environment must be configured so that each user has their own home directory. Otherwise, the SAS git functionality will not be able to find the

known_hosts file. For information about configuring user home directories, see this blog post.

SAS Visual Investigator

After Deployment Commands

Removal of System Table Storage for Context Records Replication

SAS Visual Investigator deprecated the option to store replication records in a system table in 2024.09. With this release, that functionality has been removed. For more information about how flows are affected and how to retain the ability to make updates to a flow that is using system replication, see "Removal of System Table Storage for Context Records Replication" in *What's New in SAS Visual Investigator*.

2025.09 (November 2025)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Changes for Launcher Service Transformers

The transformers used to configure the user and super-user process limits for the launcher service have been revised. If you have used these transformers in your deployment, you must replace the original transformers with the new ones. You must also ensure that any revisions that you made to the original transformers have been duplicated to the new ones.

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

   - launcher-user-process-limit.yaml
   - launcher-super-user-process-limit.yaml

   If either or both of the transformers are listed in the base kustomization.yaml file, continue with this deployment note. If neither is listed in the base kustomization.yaml file, your deployment is unaffected by this change, and you should skip the rest of this deployment note.

2 The full entry for the transformer or transformers is the location where those files can be found. Go to that location and rename the affected transformer file or files with a "2". For example, if you have **`$deploy/site-config/sas-launcher/configure/launcher-user-process-limit.yaml`**, rename that file launcher-user-process-limit2.yaml.

3 Copy the new version of the file or files from **`$deploy/sas-bases/examples/sas-launcher/configure/`** so that they are in the same directory location as the original file or files. Using the example from Step 2, you would have two files at **`$deploy/site-config/sas-launcher/configure/`**, launcher-user-process-limit.yaml and launcher-user-process-limit2.yaml.

4 Using the original version of the file or files as a guide, edit the new versions so that the original decisions you made are duplicated in the new files.

5 Delete the original version of the transformer files, thereby removing files that have a "2" in their name.

For more information about configuring the launcher service, see the README file located at **`$deploy/sas-bases/examples/sas-launcher/configure/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configuration_settings_for_sas_launcher_service.htm`** (for HTML format).

Updated CAS Host Launch Configuration Files

A property in the file to set up and configure CAS Host Launch and two security content constraints (SCCs) have been revised. If your deployment uses CAS Host Launch, you must replace the original configuration file with the new one before updating your software. Additionally, if you are using Red Hat OpenShift as your cloud provider and depending on the cas-server SCC you used in your deployment, you might also have to apply and bind the revised SCC.

Note: For details on CAS Host Launch, see "Enable Host Launch" in *SAS Viya Platform: Deployment Guide*.

1. To determine if your deployment uses CAS Host Launch, look at the base kustomization file (**$deploy/kustomization.yaml**). In the transformers block, search for the string "cas-enable-host". If the string is present, continue with the steps in this deployment note. If the string is not present, your deployment does not have CAS Host Launch enabled, and you should skip the rest of this deployment note.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `cas-enable-host2.yaml`.
3. Copy the **$deploy/sas-bases/examples/cas/configure/cas-enable-host.yaml** file to the location of your cas-enable-host2.yaml file. You should now have a cas-enable-host.yaml file and a cas-enable-host2.yaml file in the same directory
4. Compare the cas-enable-host.yaml and cas-enable-host2.yaml files. Use the cas-enable-host2.yaml file as a guide for the revisions to be made to the cas-enable-host.yaml file in order to match the configuration settings used in your existing deployment.
5. After your revisions are complete, delete the cas-enable-host2.yaml file.

   Note: If your deployment runs on Red Hat OpenShift, continue with the steps in this deployment note. Otherwise, you have completed the required steps, and you should move on to the next deployment note.

6. Determine the SCC that you used in the deployment with the following command:

   ```
   kubectl get scc | grep "cas-server"
   ```

7. Your next steps depend on the output of the command in Step 6.

   - If `sas-cas-server-host` is in the output of the command, apply the revised SCC and bind it to the service account with the following commands:

     ```
     kubectl apply -f sas-bases/examples/cas/configure/cas-server-scc-host-launch.yaml
     ```

```
oc -n name-of-namespace adm policy add-scc-to-user sas-cas-server-host -z
sas-cas-server
```

- If `sas-cas-server-sssd` is in the output of the command, apply the revised SCC and bind it to the service account with the following commands:

  ```
  kubectl apply -f sas-bases/examples/cas/configure/cas-server-scc-
  sssd.yaml
  ```

  ```
  oc -n name-of-namespace adm policy add-scc-to-user sas-cas-server-sssd -z
  sas-cas-server
  ```

- If neither of those SCCs is listed, skip to Step 9.

8 To ensure that the SCC used for backups is synchronized with the CAS server host SCC, perform the following commands:

```
kubectl apply -f sas-backup-pv-copy-cleanup-job-scc.yaml
```

```
oc create -f sas-backup-pv-copy-cleanup-job-scc.yaml
```

9 In the cluster, look for any namespaces that are using a release earlier than 2025.06. Perform the following commands for each namespace that is older than the 2025.06 release:

- Depending on the CAS server SCC that you use in the namespace, perform one of the following commands.

  ```
  kubectl patch scc sas-cas-server-host --type json -p '[{"op": "add",
  "path": "/allowedCapabilities/-", "value": "DAC_OVERRIDE"}]'
  ```

  ```
  kubectl patch scc sas-cas-server-sssd --type json -p '[{"op": "add",
  "path": "/allowedCapabilities/-", "value": "DAC_OVERRIDE"}]'
  ```

- Perform the following command to revise the backup SCC:

  ```
  kubectl patch scc sas-backup-pv-copy-cleanup-job --type json -p '[{"op":
  "add", "path": "/allowedCapabilities/-", "value": "DAC_OVERRIDE"}]'
  ```

Update Configuration for SAS Servers Using Kerberos on OpenShift

Note: This deployment note is repeated from version 2025.03 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.

Note: If your deployment is not running on Red Hat OpenShift, skip this deployment note.

Because of differences in how Red Hat OpenShift handles Kerberos, SAS is introducing new transformers specifically for deployments on Red Hat OpenShift. The new transformers take the place of transformers in use prior to the update.

1 Search the transformers block of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following entry:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-tls.yaml
```

If you find the entry, replace it with the following line:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-tls-openshift.yaml
```

2 Search the transformers block of the base kustomization.yaml file for the following entry:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-no-tls.yaml
```

If you find the entry, replace it with the following line:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-no-tls-openshift.yaml
```

3 If neither entry is in your base kustomization.yaml file, no further action is required.

Revised SAS Configurator for Open Source Transformer

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised to support a newer version of Python and is 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 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.

5 After your revisions are complete, delete the change-configuration2.yaml file.

For more information about the change-configuration.yaml file, see "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

JAVAOBJ Added to LOCKDOWN Default Blocked List

The JAVAOBJ access method has been added to the list of access methods that LOCKDOWN blocks by default. If your deployment uses the JAVAOBJ access

method, after the update your access will be blocked unless you add JAVAOBJ to the enabled list before the update.

To enable the JAVAOBJ access method, follow the instructions in the README file located at **$deploy/sas-bases/examples/sas-programming-environment/lockdown/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/lockdown_settings_for_the_sas_programming_environment.htm** (for HTML format). Ensure that you modify the enable-lockdown-access-methods.yaml file and also that you add a reference to that file in the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**).

Updated Storage Class Transformer

Note: This deployment note is repeated from version 2025.03 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-viya-volume-storage-class.yaml file has been modified to target only sas-programming-environment podTemplates that make use of the viya volume. 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 the transformers block of 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:

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      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
```

Note: For more information about using the change-viya-volume-storage-class.yaml file, see the README file located 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).

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

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

Guest Access

If you have enabled guest access for users of SAS Job Execution Web Application, a manual step is required. An authorization rule has changed, and it potentially affects guest access.

Users who are logged in to the Guest account will see errors from the file service when they attempt to run jobs. The errors state, "`Initial Files Service creation of file failed`." To avoid these errors, modify an authorization rule after you have deployed the updates. Use the SAS Viya platform command-line interface to run the following commands:

Note: The authorization plug-in to the SAS Viya platform CLI is required.

```
sas-viya --profile c1 --insecure auth login

sas-viya plugins install --repo SAS authorization

./sas-viya authorization update-rule --id jobexec-files-1 --condition
"clientId() == 'sas.SASJobExecution' || clientId() ==
'sas.jobExecution' || clientId() == 'sas.compute'" || clientId() ==
'sas.files'
```

Model Studio

Before Deployment Commands

JAVAOBJ Added to LOCKDOWN Default Blocked List

To run the Open Source Code node in Model Studio, the JAVAOBJ access method must be enabled. However, the JAVAOBJ access method has been added to the list of access methods that LOCKDOWN blocks by default. If your deployment uses the JAVAOBJ access method, after the update your access will be blocked unless you add JAVAOBJ to the enabled list before the update.

To enable the JAVAOBJ access method, follow the instructions in the README file located at **$deploy/sas-bases/examples/sas-programming-environment/lockdown/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/lockdown_settings_for_the_sas_programming_environment.htm** (for HTML format). Ensure that you modify the enable-lockdown-access-methods.yaml file and also that you add a reference to that file in the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**).

SAS Data Quality

Before Deployment Commands

New Configuration File Format

The configuration files for controlling the size of the persistent volume claim (PVC) used to store the Quality Knowledge Base have been completely revised. The new format will simplify future updates.

1. In the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the following line:

   ```
   - site-config/data-quality/storageclass/storage-class-transformer.yaml
   ```

   If the line is not present, you are not affected by the change and should skip the rest of the deployment note. If the line is present, continue with Step 2.

2. Delete the `- site-config/data-quality/storageclass/storage-class-transformer.yaml` line from the base kustomization.yaml.

3. To use the new configuration files, follow the steps described in the README file at **$deploy/sas-bases/examples/data-quality/storagesize/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/quality_knowledge_base_for_the_sas_viya_platform.htm** (for HTML format).

4. Delete the actual `$deploy/site-config/data-quality/storageclass/storage-class-transformer.yaml` file from the deployment assets.

SAS Decisions Runtime Builder

Before Deployment Commands

Kaniko No Longer Available for Container Images

With the update, Kaniko is no longer available for SAS Decisions Runtime Builder to publish models or decisions to container destinations. Perform the following tasks to clean Kaniko from your deployment.

1 Look at the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following entry:

```
- site-config/sas-decisions-runtime-builder/kaniko
```

If the line is not present, then you do not have the SAS Decisions Runtime Builder service with Kaniko deployed, and you should skip the rest of this deployment note. If the line is present, continue with Step 2.

2 In the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**), check for the following entry:

```
- site-config/sas-decisions-runtime-builder/buildkit
```

If the line is present, BuildKit has been installed and configured. You should skip to Step 4. Otherwise, continue with Step 3.

3 To install and configure BuildKit, perform the steps described in the README file located at **$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_buildkit_for_sas_decisions_runtime_builder_service.htm** (for HTML format).

4 Make the following changes to your base kustomization.yaml file (**$deploy/kustomization.yaml**):

- Search for `- site-config/sas-decisions-runtime-builder/kaniko` in the resources block. If it is present, remove it.
- Search for `- sas-bases/overlays/sas-decisions-runtime-builder/kaniko/kaniko-transformer.yaml` in the transformers block. If it is present, remove it.
- Search for `- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/buildkit-kaniko-transformer.yaml` in the transformers block. If it is present, replace it with `- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/buildkit-transformer.yaml`.

Revised Time-Sensitive Configuration Files

If your deployment includes SAS Decisions Runtime Builder running with BuildKit, your configuration contains two files that are time-sensitive. The update contains revised versions of those files to keep your deployment running. If the revised files are not placed correctly, your deployment will cease working because the configuration files being used by your deployment will expire.

1 In your base kustomization.yaml file, (**$deploy/kustomization.yaml**), search the resources block for the following line:

```
- site-config/sas-decisions-runtime-builder/buildkit
```

If the line is not present, then your deployment is unaffected by the change and you should skip this deployment note and the related one that this note links to. If the line is present, continue with Step 2.

2 Copy the following files to **$deploy/site-config/sas-decisions-runtime-builder/buildkit**:

- **$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/buildkit-client-certs.yaml**
- **$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/buildkit-daemon-certs.yaml**

Copying the files should overwrite files in the target location with the same name. This is the desired behavior.

3 After updating your deployment with the appropriate deployment commands, be sure to perform the task described in "Restart the Daemon Buildkit Pod".

Revised Configuration File

Note: If you are performing an update that includes installing and configuring BuildKit from Step 3 of the tasks associated with "Kaniko No Longer Available for Container Images" on page 38, skip this note.

The publish-job-template.yaml file has been revised and is replaced during update. Ensure that its changes are included in your deployment by performing the following steps.

1 Go to the **$deploy/site-config/sas-decisions-runtime-builder/buildkit** directory.

2 (Red Hat OpenShift only) In that directory, change the publish-job-template.yaml file name to publish-job-template2.yaml.

3 Copy the **$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/publish-job-template.yaml** file to **$deploy/site-config/sas-decisions-runtime-builder/buildkit**. If your deployment is not on Red Hat OpenShift, the new publish-job-template.yaml file is in place and you are finished with this deployment note. If your deployment is on Red Hat OpenShift,

you now have a publish-job-template.yaml file and a publish-job-template2.yaml file in the same directory.

4 (Red Hat OpenShift only) In the publish-job-template.yaml file, revise the {{ FSGROUP_VALUE }} to match the value used in the publish-job-template2.yaml file.

5 (Red Hat OpenShift only) Delete the publish-job-template2.yaml file.

For more information about the publish-job-template.yaml file, see the README file located at **$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_buildkit_for_sas_decisions_runtime_builder_service.htm** (for HTML format).

After Deployment Commands

Restart the Daemon Buildkit Pod

Note: If you did not copy and move the files described in "Revised Time-Sensitive Configuration Files" on page 39, skip this deployment note.

Restart your deployment's buildkit daemon pod with the following command:

```
kubectl -n name-of-namespace rollout restart deployments/buildkitd
```

SAS Detection Architecture

Before Deployment Commands

Revised Configuration YAML File

The update contains a revision to the primary configuration file, **/$deploy/site-config/sas-detection/detection-engine-deployment.yaml**. In order for SAS Detection Architecture to continue to work smoothly, you must revise that file before the update.

1 In your deployment assets, search for a **/$deploy/site-config/sas-detection/detection-engine-deployment.yaml** file. If that file is not present, then you do not have SAS Detection Architecture and should skip the rest of this deployment note. If the file is present, continue with this procedure.

2 In the **/$deploy/site-config/sas-detection/detection-engine-deployment.yaml** file, find the following code block.

```
### Only the NAMESPACE must be specified
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: reader
  namespace: {{ NAMESPACE }}
rules:
  - apiGroups: [""]
    resources: ["deployments", "deployments/scale", "pods"]
    verbs: ["get", "watch", "list"]
  - apiGroups: ["apps"]
    resources: ["deployments", "deployments/scale", "pods"]
    verbs: ["get", "watch", "list"]
---
```

In your file, the variable `{{ NAMESPACE }}` has been replaced with the name of your namespace.

3 In that code block, revise the `resources` line (highlighted in the example above) by adding `"services"` and `"endpoints"`. The revised code block should look like this:

```
### Only the NAMESPACE must be specified
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: reader
  namespace: {{ NAMESPACE }}
rules:
  - apiGroups: [""]
    resources: ["deployments", "deployments/scale", "pods", "services", "endpoints"]
    verbs: ["get", "watch", "list"]
  - apiGroups: ["apps"]
    resources: ["deployments", "deployments/scale", "pods", "services", "endpoints"]
    verbs: ["get", "watch", "list"]
---
```

SAS Micro Analytic Service

Before Deployment Commands

Revised ASTORES Transformer File

SAS Micro Analytic Service is a component used in a number of offerings in the SAS Viya platform, including SAS Event Stream Processing Studio, SAS Intelligent Decisioning, and SAS Model Manager. This update contains a revised transformer for SAS Micro Analytic Service, astores-transformer.yaml. For some deployments, the revised astores-transformer.yaml file might be in a different location than expected. Use the following steps to determine if your deployment is affected and

to ensure that you have the latest astores-transformer.yaml file in the correct location for the update.

1 Search the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for a reference to astores-transformer.yaml.

2 Based on the result of your search, perform one of the following tasks:

- If the full reference is `- sas-bases/overlays/sas-microanalytic-score/astores/astores-transformer.yaml`, the reference is correct. Do not take any further steps.
- If the full reference is `- site-config/sas-microanalytic-score/astores/astores-transformer.yaml`, revise the line to read as follows: `- sas-bases/overlays/sas-microanalytic-score/astores/astores-transformer.yaml`. Then save and close the base kustomization.yaml file.

  Note: With this revision, the **$deploy/site-config/sas-microanalytic-score/astores/astores-transformer.yaml** file will no longer be used. It can be safely deleted from your deployment assets.

SAS Model Publish Service

Before Deployment Commands

New Configuration File Format

The configuration files for controlling the size of the persistent volume claim (PVC) used by the SAS Model Publish service have been revised. The new format will simplify future updates.

1 In the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the following line:

```
- site-config/sas-model-publish/git
```

If the line is not present, you are not affected by the change and should skip the rest of the deployment note. If the line is present, continue with Step 2.

2 To use the new configuration files, follow the steps described in the README file at **$deploy/sas-bases/examples/sas-model-publish/git/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_git_for_sas_model_publish_service.htm** (for HTML format).

Note: SAS recommends referring to the **$deploy/site-config/sas-model-publish/git/storage.yaml** file while updating the new configuration.env file

to ensure that the values used in your current deployment are used in the update.

3 The new configuration of the files means that the properties and values used in the old storage.yaml file are now in the configuration.env file. Because the file is no longer needed, delete **$deploy/site-config/sas-model-publish/git/storage.yaml** from your deployment assets.

Remove Kaniko References

Kaniko is no longer being used by the SAS Model Publish service. Therefore, you must remove references to it from the base kustomization.yaml file. Take the following steps:

1 Search for `- site-config/sas-model-publish/kaniko` in the resources block. If it is present, remove it.

2 Search for `- sas-bases/overlays/sas-model-publish/kaniko/kaniko-transformer.yaml` in the transformers block. If it is present, remove it.

3 Delete the **$deploy/site-config/sas-model-publish/kaniko** directory from your deployment assets.

If you are using Red Hat OpenShift, ensure that you have bound and applied the security context constraint for sas-model-publish. For more information, see "sas-model-publish" in *SAS Viya Platform: Deployment Guide*.

SAS SpeedyStore

Before Deployment Commands

Updated Configuration Transformer File

Note: The tasks described in this deployment note only apply if your deployment is running on AWS. If your deployment is running on a different provider, you should skip this deployment note.

The update contains a revised transformer file, sas-singlestore-cluster-config.yaml. If you used the sas-singlestore-cluster-config.yaml file in your deployment, you must replace the existing version of the file with the new version.

1 In the transformers block of your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the string "site-config/sas-singlestore/sas-singlestore-cluster-config.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.

2 Go to **$deploy/site-config/sas-singlestore/** and change the file name to `sas-singlestore-cluster-config2.yaml`.

3 Copy **$deploy/sas-bases/examples/sas-singlestore/sas-singlestore-cluster-config.yaml** to **$deploy/site-config/sas-singlestore/**. You should now have a sas-singlestore-cluster-config.yaml file and a sas-singlestore-cluster-config2.yaml file in the same directory.

4 Using the sas-singlestore-cluster-config2.yaml file as a guide, revise the sas-singlestore-cluster-config.yaml file so that it includes the same values you used in the earlier version of the file. Notice that the revisions to the sas-singlestore-cluster-config.yaml can include user-supplied values for variables as well as commenting out and uncommenting lines within the file.

Note: For information about using the sas-singlestore-cluster-config.yaml file, see the README file located at **$deploy/sas-bases/examples/sas-singlestore/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_singlestore_cluster_operator.htm** (for HTML format).

5 After your revisions are complete, delete the sas-singlestore-cluster-config2.yaml file.

2025.03 (May 2025)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Ensure RabbitMQ Feature Flags Are Enabled

IMPORTANT The SAS Viya platform uses RabbitMQ as its messaging and streaming broker. The SAS Viya platform requires that RabbitMQ be updated periodically along with the rest of your SAS Viya platform software. To ensure that RabbitMQ is updated successfully and thus provide the framework for a successful SAS Viya platform update, run the following command:

```
kubectl -n name-of-namespace exec -it sas-rabbitmq-server-0 -- bash -c
"source ~/.bashrc ; rabbitmqctl enable_feature_flag all"
```

Required Ingress Controller Configuration

If you are using ingress-nginx 1.12.*x* and later for your Kubernetes cluster, a change to the default configuration from NGINX will affect your deployment. For more information, see "Required ingress-nginx Controller Configuration" in *SAS Viya Platform: Deployment Guide*.

Change to CAS Backing Store Enablement

The method by which the CAS backing store is enabled has been changed. Previously, the backing store was enabled by modifying the custom resource for the CAS operator. Now, you must enable the backing store in four different scenarios using patch transformer .yaml files. To determine whether you have enabled the backing store for your current deployment, run the following command:

```
kubectl -n name-of-namespace get cas default -
o=jsonpath={.spec.controllerTemplate.spec.containers[0].env} | grep
TK_BACKING_STORE_DIR > /dev/null;echo $?
```

If the output is `0`, then the CAS backing store is enabled in your current deployment, and you must use the new transformer files to enable it in your updated deployment. If the output is `1`, then the backing store has not been enabled and you can either skip the rest of this deployment note or enable the backing store for the first time.

There are four transformers to choose from:

- If you are using CAS auto-resourcing, **$deploy/sas-bases/examples/cas/configure/cas-enable-default-backing-store.yaml** causes the CAS operator to select an appropriate size for the backing store.
- **$deploy/sas-bases/examples/cas/configure/cas-enable-backing-store.yaml** selects a specific size for the backing store.
- **$deploy/sas-bases/examples/cas/configure/cas-enable-backing-store-with-priority-groups.yaml** selects a specific size for five separate backing stores, one for each priority group.
- **$deploy/sas-bases/examples/cas/configure/cas-enable-backing-store-with-priority-group-one.yaml** selects specific sizes for two backing stores, one for users in priority group one and a second for all other users. This is appropriate only when CAS resource management in enabled.

To use the new transformer files:

1. Copy the patch transformer that you want to use from **$deploy/sas-bases/examples/cas/configure/*file-name*** to **$deploy/site-config/cas/configure/*file-name***.
2. Follow the instructions in the comments of the copied patch transformer to replace variables with the appropriate values.
3. In the base kustomization.yaml file (**$deploy/kustomization.yaml**), add a line to the transformers block that points to the patch transformer you want to use. Here is an example using the transformer that uses a specific size for the backing store:

   ```
   transformers:
   ...
   - site-config/cas/configure/cas-enable-backing-store.yaml
   ```

   Replace `cas-enable-backing-store.yaml` with the name of the transformer you have chosen to use.

For more information about the new required .yaml files, see the "Enable a Backing Store for CAS" 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).

Updated Storage Class Transformer

The change-viya-volume-storage-class.yaml file has been modified to target only sas-programming-environment podTemplates that make use of the viya volume. 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 the transformers block of 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:

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      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
```

Note: For more information about using the change-viya-volume-storage-class.yaml file, see the README file located 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).

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

Red Hat OpenShift and Root Routes

Note: If your deployment is not using Red Hat OpenShift, skip this note.

Note: If your deployment is maintained with the SAS Viya Platform deployment operator or the sas-orchestration command, skip this deployment note. This note only applies to deployments maintained using Kubernetes commands.

Red Hat OpenShift uses routes to allow you to host your application at a public URL. It can either be secure or unsecured, depending on the network security configuration of your application. Prior to this version, the root routes redirected to **/SASDrive**. However, with this update, **/SASDrive** has been retired and the root routes redirect to **/SASLanding**. The change happens automatically with the update if the existing root routes are deleted. Therefore, deployments on Red Hat OpenShift must perform the following command in order to use the new root routes:

```
kubectl delete route sas-drive-app-root sas-drive-app-root-multitenant -n name-of-namespace
```

Update Configuration for SAS Servers Using Kerberos on OpenShift

Note: If your deployment is not running on Red Hat OpenShift, skip this deployment note.

Because of differences in how Red Hat OpenShift handles Kerberos, SAS is introducing new transformers specifically for deployments on Red Hat OpenShift. The new transformers take the place of transformers in use prior to the update.

1. Search the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following entry:

   ```
   - sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-tls.yaml
   ```

   If you find the entry, replace it with the following line:

   ```
   - sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-tls-openshift.yaml
   ```

2. Search the transformers block of the base kustomization.yaml file for the following entry:

   ```
   - sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-no-tls.yaml
   ```

   If you find the entry, replace it with the following line:

   ```
   - sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-no-tls-openshift.yaml
   ```

3. If neither entry is in your base kustomization.yaml file, no further action is required.

Revised SAS Configurator for Open Source Transformer

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised to support a newer version of Python and is 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. 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. Note that the `default_py.python_signature` and `default_py.python_tarball` properties have been modified to support a newer version of Python. SAS recommends that you keep the values for these properties.

For more information about the change-configuration.yaml file, see "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

After Deployment Commands

Configure the Multi-Language Python Library for the Enhanced SAS Compute Server

With this update, SAS has released enhancements to the SAS Compute Server (for details of the enhancements, see "Enhancements to the SAS Compute Server" in *SAS Programmer's Guide: Essentials*.) The enhancements take advantage of the configuration of Python. If you have not already configured Python and want to use it in your deployment, use the procedure described at "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*. If you have configured Python before this update, ensure that you have performed step 2d and step 2e in "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*. Both steps are new with this update.

Multiple Configuration Definitions for SAML Identity Providers

SAML libraries have been revised and no longer allow multiple configurations with duplicate entityID values. You should perform the following steps to determine if you have configurations with duplicate values. If you have duplicate values, the procedure also provides information for addressing the issue.

1 To list all the configuration properties under config/SASLogon that contain "saml":

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse config/SASLogon/" | grep saml
```

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

The output is a list of entries. Here is an example of an entry:

```
config/SASLogon/sas.logon.saml.providers/mysamlentityid/linkText=Use your corporate credenti
```

2 Evaluate the output of the command in Step 1 to determine if it contains multiple properties. If the output of the command is a single item, you are not affected by the issue and you can skip the rest of this note. If there is more than one item in the output, continue with the next step.

3 Review the contents for each idpMetadata property in the output list. Determine whether the entityID value is unique for each or if there are identical entityID values. If you do not have identical entityID values, you are not affected by the issue and you can skip the rest of this note. If you have identical entityID values, continue with the next steps.

Note: You might see XML returned for the idpMetadata property. You can view the XMl directly in the command output to check the entityID values. You might also see a URI returned for the idpMetadata property. If a URI is returned for the idpMetadata property, you must enter that URI in a browser to view the XML and check the entityID values.

4 Perform the following command:

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse configurationservice/configurations/" |
grep sas.logon.saml.providers
```

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

5 Create a case at the Customer Service Portal in order to receive assistance from SAS Technical Support for the rest of the procedure. In the case that you create, be sure that you include the output of the command in Step 4.

Multiple OIDC Configurations with Same Issuer Property

OIDC support has been revised and no longer allow multiple configurations with duplicate issuer values. You should perform the following steps to determine if you have configurations with duplicate values. If you have duplicate values, the procedure also provides information for addressing the issue.

1 To list all the configuration properties under config/SASLogon that contain "oidc":

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse config/SASLogon/" | grep oauth
```

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

2 Evaluate the output of the command in Step 1 to determine if it contains multiple issuer properties. If there is not more than one issuer property in the output or if no issuer properties are duplicated, you are not affected by the concern and you can skip the rest of this note. If there are duplicate issuer properties in the output, continue with the next step.

3 Perform the following command:

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse configurationservice/configurations/" |
grep sas.logon.oauth.providers
```

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

4 Create a case at the Customer Service Portal in order to receive assistance from SAS Technical Support for the rest of the procedure. In the case that you create, be sure that you include the output of the command in Step 3.

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 2025.03 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

Guest Access

If you have enabled guest access for users of SAS Job Execution Web Application, a manual step is required after an update. An authorization rule has changed, and it potentially affects guest access.

Users who are logged in to the Guest account will see errors from the file service when they attempt to run jobs. The errors state, "`Initial Files Service creation of file failed`." To avoid these errors, modify an authorization rule after you have deployed the updates. Use the SAS Viya platform command-line interface to run the following commands:

Note: The authorization plug-in to the SAS Viya platform CLI is required.

```
sas-viya --insecure auth login

sas-viya plugins install --repo SAS authorization

./sas-viya authorization update-rule --id jobexec-files-1 --condition
"clientId() == 'sas.SASJobExecution' || clientId() ==
'sas.jobExecution' || clientId() == 'sas.compute'"
```

SAS Allowance for Credit Loss

Before Deployment Commands

SAS Allowance for Credit Loss Configuration Changes

The process of configuring SAS Allowance for Credit Loss has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Allowance for Credit Loss, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-acl/resources/acl_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete steps 3-9 in the "Installation" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-acl/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_allowance_for_credit_loss_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Asset and Liability Management

Before Deployment Commands

SAS Asset and Liability Management Configuration Changes

The process of configuring SAS Asset and Liability Management has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Asset and Liability Management, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-alm/resources/alm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete steps 3-7 in the "Installation" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-alm/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_asset_and_liability_management_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Dynamic Actuarial Modeling

Before Deployment Commands

SAS Dynamic Actuarial Modeling Configuration Changes

The process of configuring SAS Dynamic Actuarial Modeling has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Dynamic Actuarial Modeling, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete steps 3-6 in the "Installation" section of the README file located at **`$deploy/sas-bases/examples/sas-risk-cirrus-pcpricing/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_sas_dynamic_actuarial_modeling_for_deployment.htm`** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Event Stream Processing

Before Deployment Commands

PVC Changes

SAS Event Stream Processing now creates a PersistentVolumeClaim (PVC) with a default storage capacity of 5 GB. Here are the steps to implement the new process and remove the files that supported the previous process:

1. In the base kustomization.yaml file (**$deploy/kustomization.yaml**), remove the following lines:
   - In the resources block, remove `- site-config/sas-event-stream-processing-studio-app/pvc/resources.yaml`
   - In the transformer block, remove the following:
     - `- sas-bases/overlays/sas-event-stream-processing-studio-app/data/data-transformer.yaml`
     - `- sas-bases/overlays/sas-event-stream-manager-app/data/data-transformer.yaml`
     - `- sas-bases/overlays/sas-event-stream-processing-studio-app/pvc/pvc-transformer.yaml`
     - `- sas-bases/overlays/sas-event-stream-manager-app/pvc/pvc-transformer.yaml`
2. Copy the files in the **$deploy/sas-bases/examples/sas-event-stream-processing-studio-app/storage** directory to the **$deploy/site-config/sas-event-stream-processing-studio-app/storage** directory. Create the destination directory if it does not exist.
3. Review the storage-capacity value in the current **$deploy/site-config/sas-event-stream-processing-studio-app/pvc/resources.yaml** file. Update the **$deploy/site-config/sas-event-stream-processing-studio-app/storage/esp-storage-size-transformer.yaml** file with that same value.
4. Revise the base kustomization.yaml file by adding `- site-config/sas-event-stream-processing-studio-app/storage/esp-storage-size-transformer.yaml` to the transformers block. The resulting kustomization.yaml should look like this:

```
transformers:
...
- site-config/sas-event-stream-processing-studio-app/storage/esp-storage-size-transformer.yaml
```

```
...
```

5 Also in the base kustomization.yaml file, in the patches block, add `sas-event-stream-processing-studio-app` to the `annotationSelector` line of the `site-config/storageclass.yaml` patch. 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,sas-cas-operator,sas-pyconfig,
sas-risk-cirrus-search,sas-risk-modeling-core,sas-event-stream-processing-studio-app)
```

Note: For more details about the SAS Event Stream Processing PVC, see the README file located at **`$deploy/sas-bases/examples/sas-event-stream-processing-studio-app/storage/README.md`** (for Markdown format) and at **`$deploy/sas-bases/docs/configuring_file_storage_for_sas_event_stream_processing_studio_projects.htm`** (for HTML format).

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects.

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken:

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in "Restore Projects After Update".

Delete espserver Before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS Expected Credit Loss

Before Deployment Commands

SAS Expected Credit Loss Configuration Changes

The process of configuring SAS Expected Credit Loss has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Expected Credit Loss, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-ecl/resources/ecl_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete steps 3-9 in the "Installation" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-ecl/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_expected_credit_loss_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Insurance Capital Management

Before Deployment Commands

SAS Insurance Capital Management Configuration Changes

The process of configuring SAS Insurance Capital Management has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Insurance Capital Management, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-icm/resources/icm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration for SAS Insurance Capital Management" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-icm/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_insurance_capital_management_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Insurance Contract Valuation Foundation

Before Deployment Commands

SAS Insurance Contract Valuation Foundation Configuration Changes

The process of configuring SAS Insurance Contract Valuation Foundation has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Insurance Contract Valuation Foundation, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-icv/resources/icv_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration Files for SAS Insurance Contract Valuation Foundation" section of the README file located at **`$deploy/sas-bases/examples/sas-risk-cirrus-icv/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_sas_insurance_contract_valuation_foundation_for_deployment.htm`** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Integrated Regulatory Reporting

Before Deployment Commands

SAS Integrated Regulatory Reporting Configuration Changes

The process of configuring SAS Integrated Regulatory Reporting has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Integrated Regulatory Reporting, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-eba/resources/eba_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration Files for SAS Integrated Regulatory Reporting" section of the README file located at **`$deploy/sas-bases/examples/sas-risk-cirrus-eba/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_sas_integrated_regulatory_reporting_for_deployment.htm`** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Model Repository Service

Before Deployment Commands

Revised Overlay Requires Revision to Base kustomization.yaml File

Note: If you are deploying on a platform other than Red Hat OpenShift, skip this deployment note.

Service accounts are created for some SAS services by default. Previously, an overlay was used to create these service accounts for the Model Repository service, enabling it to run with additional permissions in a Red Hat OpenShift environment. Two transformers in this overlay are no longer needed. To avoid failures when you perform an update on OpenShift, references to two YAML files must be removed.

Note: The Model Repository service is part of the SAS Viya Platform deployment for SAS Model Manager, SAS Intelligent Decisioning, and Model Studio.

Before you perform an update for the Model Repository service in a Red Hat OpenShift environment, perform these steps:

1. Use your preferred text editor to modify the base kustomization.yaml file (**`$deploy/kustomization.yaml`**).
2. From the resources block, remove the reference to **`sas-bases/overlays/sas-model-repository/service-account/sa.yaml`**.
3. From the transformers block, remove the reference to **`sas-bases/overlays/sas-model-repository/service-account/sa-transformer.yaml`**.
4. Save your changes to the file.

SAS Model Risk Management

Before Deployment Commands

SAS Model Risk Management Configuration Changes

The process of configuring SAS Model Risk Management has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Model Risk Management, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-mrm/resources/mrm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration Files Related to SAS Model Risk Management" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-mrm/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_model_risk_management_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Regulatory Capital Management

Before Deployment Commands

SAS Regulatory Capital Management Configuration Changes

The process of configuring SAS Regulatory Capital Management has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Regulatory Capital Management, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-rcm/resources/rcm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration Files Related to SAS Regulatory Capital Management" section of the README file located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rcm/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_sas_regulatory_capital_management_for_deployment.htm`** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Risk Cirrus Core

Before Deployment Commands

Updated Configuration File

SAS Risk Cirrus Core is a common layer that is used by all Cirrus solutions. The main configuration file for SAS Risk Cirrus Core, core_transform.yaml, has been revised with two new properties. While the new properties are optional, users of SAS Risk Solutions should be aware of their presence and, if desired, take steps to add the new properties to their deployment.

1 To determine if your deployment includes SAS Risk Cirrus Core, go to the base kustomization.yaml file (**$deploy/kustomization.yaml**). In the transformers block, search for "core_transform.yaml". If that phrase is present, then you have SAS Risk Cirrus Core in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.

2 Review the SAS Risk Cirrus Core README file for information about the two new variables, `SAS_RISK_CIRRUS_CODE_LIB_APPEND_NAMESPACE_TO_BRANCH_FLG` and `SAS_RISK_CIRRUS_CODE_LIB_SKIP_PUSH_IF_BRANCH_EXISTS`. Based on the information in the README, decide if you want to add either or both of the properties to your deployment. If you decide to add the properties, continue with these steps. If you do not want to add either of the new properties, skip the rest of this note.

   The SAS Risk Cirrus Core README file is located at **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

3 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 `core_transform2.yaml`. For example, if the file is located at **$deploy/site-config/sas-risk-cirrus-rcc/resources/core_transform.yaml**, change the file name to **$deploy/site-config/sas-risk-cirrus-rcc/resources/core_transform2.yaml**.

4 Copy the **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/core_transform.yaml** file to the directory you used in step 2. You should now have a core_transform.yaml file and a core_transform2.yaml file in the same directory.

5 Follow the steps in the SAS Risk Cirrus Core README file to add the values for the new variables, `SAS_RISK_CIRRUS_CODE_LIB_APPEND_NAMESPACE_TO_BRANCH_FLG` and

SAS_RISK_CIRRUS_CODE_LIB_SKIP_PUSH_IF_BRANCH_EXISTS, to the core_transform.yaml file.

6 Use the core_transform2.yaml file as a guide for filling in the rest of the values in the core_transform.yaml file.

7 When the core_transform.yaml file is complete, delete the core_transform2.yaml file.

SAS Risk Cirrus Core Configuration Changes

The process of configuring SAS Risk Cirrus Core has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Risk Cirrus Core, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-core/resources/core_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration for Risk Cirrus Core" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Risk Factor Manager

Before Deployment Commands

SAS Risk Factor Manager Configuration Changes

The process of configuring SAS Risk Factor Manager has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no

longer required for SAS Risk Factor Manager, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-rfm/resources/rfm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration Files Related to SAS Risk Factor Manager" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-rfm/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_factor_manager_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Risk Modeling

Before Deployment Commands

SAS Risk Modeling Configuration Changes

The process of configuring SAS Risk Modeling has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Risk Modeling, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-rm/resources/rm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete the steps in the "Installation" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-rm/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/**

preparing_and_configuring_sas_risk_modeling_for_deployment.htm (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS Risk Offerings

Before Deployment Commands

Enable Risk Cirrus Objects Features

The SAS Risk Cirrus Objects Microservice stores information related to business object definitions and business object instances such as analysis data, analysis runs, models or model reviews. Cirrus Objects also stores and retrieves items related to business objects, such as attachments. These business objects are associated with the Risk Cirrus Platform that underlies most SAS Risk offerings.

A new feature for enabling the promotion of Cirrus Objects across SAS Viya platform environments has been introduced. For more details, see *What's New in SAS Risk Cirrus*.

Perform the following steps to determine if your deployment is affected and, if so, how to enable the new feature Risk Cirrus Objects.

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

```
- site-config/risk-cirrus-builder/resources/sas_risk_cirrus_builder_transform.yaml
```

If the line is present, then continue with the steps in this deployment note. If it is not present, your deployment is unaffected and you can skip the rest of this note.

2 Perform the steps described in the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-objects/resources/README.md** (for Markdown format) and **$deploy/sas-bases/docs/configure_environment_id_settings_for_sas_risk_cirrus_builder_microservice.htm** (for HTML format).

SAS Stress Testing

Before Deployment Commands

SAS Stress Testing Configuration Changes

The process of configuring SAS Stress Testing has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Stress Testing, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-st/resources/st_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete steps 3-9 in the "Installation" section of the README file located at **`$deploy/sas-bases/examples/sas-risk-cirrus-st/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_sas_stress_testing_for_deployment.htm`** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features".

SAS/ACCESS to Impala

Before Deployment Commands

Impala ODBC Driver Client Change

Prior to this release, SAS/ACCESS to Impala used the Cloudera Impala ODBC driver client. With this release, SAS/ACCESS to Impala now uses the Simba Impala ODBC

driver client. The change requires users of SAS/ACCESS to Impala to evaluate their configuration and potentially make revisions to account for the new ODBC driver client.

1 Check the **$deploy/site-config/data-access/sas-access.properties** file for the IMPALA. If you find it, SAS recommends that it be removed from the file.

2 If you have set any variables in the Impala INI file, revise the **$deploy/site-config/data-access/sas-access.properties** to remove the CLOUDERAIMPALAINI property and replace it with a SIMBAIMPALAINI property. Therefore, if the properties file contains CLOUDERAIMPALAINI=*path-to-the-Impala-INI-file*, it should be replaced with SIMBAIMPALAINI=*path-to-the-Impala-INI-file*

   If you have not set any variables in the Impala INI file, SAS recommends that the CLOUDERAIMPALAINI property be removed from the **$deploy/site-config/data-access/sas-access.properties** file.

SAS/ACCESS to Spark

After Deployment Commands

Revise JREOPTION for Updated JDBC Data Connector

The Simba Databricks JDBC Data Connector uses the Apache Arrow library to efficiently transfer result sets from Databricks to the client. The Arrow library requires the JRE option JREOPTION --add-opens=java.base/java.nio=ALL-UNNAMED to be passed to the SAS Java Runtime Environment at startup. This property must be set in the SAS Studio compute context, the SAS batch server context, and the CAS compute context. When this property is not set, SAS returns an error when attempting to read data from Databricks.

For the steps to revise the property in the appropriate contexts, see "Resolution" in *SAS Viya Platform: Deployment Guide*.

2024.09 (November 2024)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Upgrade to PostgreSQL 16

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

IMPORTANT Before you attempt the procedure described in this deployment note, complete the procedures described in every other deployment note within your current update path that is listed in a "Before Deployment Commands" section. This deployment note should be the last

procedure you perform from a deployment note in a "Before Deploymnt Commands" section. After you have completed the procedure in this deployment note, perform the deployment/update commands and then proceed with any deployment notes listed in an "After Deployment Commands" section.

SAS is upgrading the version of PostgreSQL used in the internal instances of PostgreSQL (Crunchy Postgres from Crunchy Data) from version 12 to version 16. This deployment note contains the necessary steps to perform the PostgreSQL upgrade on a running SAS Viya platform deployment.

1 Take a full pgBackRest backup. For details, see "Take an Ad Hoc pgBackRest Backup " in *SAS Viya Platform: Infrastructure Servers*.

2 Consider changing the PostgreSQLdata storage to block storage. Block storage might improve PostgreSQL performance and, more importantly, it can help you avoid a potential PostgreSQL upgrade issue on NFS storage. For details, see "Change the Storage Class of the Data Pod" in *SAS Viya Platform: Infrastructure Servers*.

3 Check the current checksum setting for the database.

   To upgrade to Crunchy Postgres 16, the checksum must be `on`. If it is `off`, the upgrade process detects the current checksum setting and automatically enables it. Checksums can cause a slight Write overhead, but the overhead is considered worthwhile for the enhanced data integrity.

   a Determine the name of the cluster or clusters.

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

   The output of this command is one or two cluster names.

   b Use each of the cluster names in the following commands. If you have two cluster names as the output, perform the commands twice, once for each cluster name:

   ```
   POD1=$(kubectl -n name-of-namespace get pods --selector="postgres-operator.crunchydata.com/cluster=cluster-name,postgres-operator.crunchydata.com/role=master" -o name)
   ```

   ```
   kubectl exec -it $POD1 -c database -n name-of-namespace -- psql -c 'show data_checksums;'
   ```

   The output of the commands contains the term `on` or `off`.

   - If the current checksum setting is `on`, the upgrade should take 5-10 minutes for each PostgreSQL cluster.
   - If the current checksum setting is `off`, the upgrade may take some extra time in order to enable checksum across the database. You can check the database size with the following command:

     ```
     kubectl exec -it $POD1 -c database -n name-of-namespace -- psql -c '\l+'
     ```

     For each 100 Gi in the database, add approximately one hour to the time required for the upgrade.

Note: If you are using the SAS Viya Platform Deployment Operator or the sas-orchestration command to update your software, you should skip Step 4 through Step 8. Only perform those steps if you are using the Kubernetes command deployment method.

4 **(Kubernetes command deployment method only)** Create the site.yaml file.

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

5 **(Kubernetes command deployment method only)** As a Kubernetes cluster-level administrator, go to the upgrade script and change its permissions:

```
cd $deploy/sas-bases/examples/crunchydata/postgres-upgrade
```

```
chmod +x postgres-upgrade.sh
```

6 **(Kubernetes command deployment method only)** Using bash, run the upgrade script with the indicated parameters:

Note: The upgrade script requires the `yq` command (version 4) to be installed. For information about how to install the `yq` command, see https://github.com/mikefarah/yq.

```
./postgres-upgrade.sh \
location-of-admin-kubeconfig \
name-of-namespace \
$deploy/site.yaml \
debug-log \
2>&1 | tee ~/postgres-upgrade-$(date +'%m%d-%H%M%S').log
```

Running the script sends some log files to the pgupgrade-log directory under home.

7 **(Kubernetes command deployment method only)** After the PostgreSQL upgrade has completed, the vacuumdb task is performed and the PostgreSQL extensions are upgraded. The PostgreSQL cluster can be used normally while these actions are performed, including continuing to update your SAS Viya platform deployment.

8 **(Kubernetes command deployment method only)** Change the directory to the $deploy directory. Delete the site.yaml file that was created in Step 4.

9 After you perform the deployment commands to update your software, perform the tasks described at "PostgreSQL 16 Post-Upgrade Processing" on page 306.

CAUTION

The PostgreSQL upgrade process is not idempotent, so the manual script or the lifecycle operation should not be run more than once. If any issues occur during their execution, contact SAS Technical Support to remediate the incomplete upgrade instead of retrying the upgrade. Retrying or attempting to fix the issue might make the PostgreSQL cluster unrecoverable.

Required Ingress Controller Configuration

If you are using ingress-nginx 1.12.*x* and later for your Kubernetes cluster, a change to the default configuration from NGINX will affect your deployment. For more information, see Required Ingress Controller Configuration.

Job Definition Rule Change Affects Group-Managed Service Account Functionality

Note: This deployment note is repeated from version 2023.12 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.

A change in the rules for job definitions prevent job requests with an associated job definition from being run by users other than the job creator or a SAS Administrator. Therefore group-managed service account functionality requires that job definitions are created in such a way that they can be shared with the group-managed service account, such as being created in a folder and then shared. Alternatively, you can create a rule that grants permission to a specific job definition, to all job definitions, or to a collection of job definitions. Perform one of the following set of steps depending on the restrictions you want to place on your service account.

Have a service account able to run any job

1. As a SAS Administrator, go to SAS Environment Manager → Rules.
2. Create a new rule with the following settings:
   - Object URI: `/jobDefinitions/definitions/*`
   - Rule Type: `Grant`
   - Permission: `Read`
   - Principal Type: `User`
   - Principal: *`name-of-service-account`*

Have a service account able to run a specific job

1. Go to SAS Environment Manager → Jobs and Flows
2. Select the **Scheduling** tab.
3. Right-click the appropriate job and choose **Properties**.
4. Under the **Job Definition** tab, determine if the ID field is populated. If so, copy the ID. If the field is not populated, then the job definition is part of the job request object and no further steps are necessary.
5. As a SAS Administrator, go to SAS Environment Manager → Rules.
6. Create a rule with the following settings:
   - Object URI: `/jobDefinitions/definitions/`*`ID-from-step-4`*

- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

7 Repeat Step 3 through Step 6 for each job you want to run under the service account.

Have the service account able to run jobs with definitions stored in SAS Content

Perform one of the following tasks:

1 Use SAS Drive to "share" the job with the service account.

2 Ensure the service account has Read (convey) permission on the folder where the job definition is stored.

File Used to Identify fsGroup Has Changed

The fsGroup field defines a special supplemental group that assigns a GID for all containers in the pod. Volumes that support ownership management are modified to be owned and writable by the GID specified in fsGroup. For ease of maintenance, the fsGroup identification has been moved to a different file.

1 To determine if you have modified the fsGroup value in your current deployment, look for the following line in the transformers block of your base kustomization.yaml file (**`$deploy/kustomization.yaml`**):

```
- site-config/security/container-security/update-fsgroup.yaml
```

If the line is present, then complete the steps in this deployment note. If the line is not present, there is no work to be done and you should skip the rest of this deployment note.

2 Perform the steps described in the "Update the fsGroup Field" section of the README file at **`$deploy/sasbases/examples/security/container-security/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/modify_container_security_settings.htm`** (for HTML format).

3 In the transformer block of the base kustomization.yaml file, remove `- site-config/security/container-security/update-fsgroup.yaml`.

SAS Workload Orchestrator Configuration File Location Change

The transformer file that automatically disables SAS Workload Orchestrator has been moved. Therefore, the reference to its location in the base kustomization.yaml file ($deploy/kustomization.yaml) must be revised to account for the new location.

1 To determine whether your deployment is affected by this change, go to the base kustomization.yaml file. Search for the following line in the transformers block of that file:

```
- sas-bases/examples/sas-workload-orchestrator/enable-disable/sas-workload-orchestrator-
```

```
disable-patch-transformer.yaml
```

If the line is present, continue with the steps in this note. If the line is not present, then you are unaffected by this change and should skip the rest of this note.

2 In the transformers block of the base kustomization.yaml file, revise the line that you searched for by replacing "examples" with "overlays":

```
transformers:
...
- sas-bases/overlays/sas-workload-orchestrator/enable-disable/sas-workload-orchestrator-
disable-patch-transformer.yaml
```

3 Because the version of the transformer in the examples subdirectory is no longer being used by your deployment, you should feel free to remove it.

SAS Configurator for Open Source Requires SCC for OpenShift

If you are updating a deployment on Red Hat OpenShift, SAS Configurator for Open Source now requires that you apply and bind a Security Context Constraint (SCC). For details, see "sas-pyconfig" in *SAS Viya Platform: Deployment Guide*.

After Deployment Commands

PostgreSQL 16 Post-Upgrade Processing

Note: These steps for configuring PostgreSQL after it has been upgraded are applicable to all deployment methods. If you used the Kubernetes commands deployment method, and if you did not perform the steps described in "Upgrade to PostgreSQL 16" on page 303, do not perform the steps in this deployment note.

CAUTION
The PostgreSQL upgrade process is not idempotent, so the manual script or the lifecycle operation should not be run more than once. If any issues occurred during their execution, contact SAS Technical Support to remediate the incomplete upgrade instead of retrying the upgrade. Retrying or attempting to fix the issue might make the PostgreSQL cluster unrecoverable.

For more information about post-upgrade processing, see the following documentation from Crunchy Data: Step 6: Complete the Post-Upgrade Tasks.

Note: If applicable, repeat the following steps for both the platform PostgreSQL cluster and the CDS PostgreSQL cluster.

1 To determine the names of your PosgreSQL clusters, run the following command:

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

The output of this command is a cluster name that includes "platform" (such as sas-crunchy-platform-postgres) and, if you have a CDS PostgreSQL cluster, a cluster name that includes "cds" (such as sas-crunchy-cds-postgres).

In the remaining steps in this deployment note, the variable *cluster-name* should be replaced by the actual cluster name. If you have two clusters, run the command with the *cluster-name* variable twice, once for each cluster.

2 (Optional) Delete the old data directory.

a Exec into the primary pod of each cluster.

```
POD1=$(kubectl -n name-of-namespace get pods --selector="postgres-
operator.crunchydata.com/cluster=cluster-name,postgres-
operator.crunchydata.com/role=master" -o name)
```

```
kubectl exec -it $POD1 -c database -n name-of-namespace -- bash
```

After the second command, you will have a bash prompt, which you can use to perform additional steps.

b Check the PostgreSQL version:

```
psql -c "select version()"
```

If the version is not 16.*x*, the PostgreSQL upgrade might have failed. Stop and contact SAS Technical Support.

c Determine whether pg12 and pg12_wal directories are present in the leader pod:

```
ls -la /pgdata
```

If the output of the command does not include information about pg12 and pg12_wal directories, you might have had a failover. You should instead check replica nodes for the pg12 directories. Use the `patronictl list` command to show the list of replicas.

d Execute the following commands to rename or to remove the old data directory. Do so at your discretion:

```
mv -v /pgdata/pg12 /pgdata/pg12.old
```

```
mv -v /pgdata/pg12_wal /pgdata/pg12_wal.old
```

or

```
rm -rf /pgdata/pg12
```

```
rm -rf /pgdata/pg12_wal
```

e Close the shell session by entering the following command:

```
exit
```

3 (Optional) Clean up objects that are no longer needed:

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

```
kubectl -n name-of-namespace annotate postgrescluster cluster-name postgres-
operator.crunchydata.com/allow-upgrade-
```

Note: The "-" at the end of the second command is not a typographical error and should be included.

IMPORTANT If you have upgraded PostgreSQL in a namespace that is in a cluster that contains other namespaces still at PostgreSQL 12, update those namespaces with the latest patches available for the version of the SAS Viya platform that is deployed in each namespace. For example, if the cluster contains a namespace at version 2024.07, find the latest patch for 2024.07 and update the namespace with that patch. This task might require coordination among namespace administrators.

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 2024.09 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

Guest Access

If you have enabled guest access for users of SAS Job Execution Web Application, a manual step is required after an update to 2024.09 LTS. An authorization rule has changed, and it potentially affects guest access.

Users who are logged in to the Guest account will see errors from the file service when they attempt to run jobs. The errors state, "`Initial Files Service creation of file failed`." To avoid these errors, modify an authorization rule after you have deployed the updates. Use the SAS Viya platform command-line interface to run the following commands:

Note: The authorization plug-in to the SAS Viya platform CLI is required.

```
sas-viya --insecure auth login

sas-viya plugins install --repo SAS authorization

./sas-viya authorization update-rule --id jobexec-files-1 --condition
"clientId() == 'sas.SASJobExecution' || clientId() ==
'sas.jobExecution' || clientId() == 'sas.compute' || clientId() ==
'sas.files'"
```

OpenSearch

Before Deployment Commands

OpenSearch Adds Configurable Parameters for Retaining Audit Log Indices

Note: This topic applies to internal instances of OpenSearch only.

OpenSearch is changing its audit log indices retention policy so that the indices are no longer retained indefinitely. Instead, the audit log indices are retained only for seven days by default. If you want to use a different retention period, OpenSearch has added a new transformer file, audit-log-retention-transformer.yaml, to configure the retention of audit log indices. For details about using this new file, see the README file at **`$deploy/sas-bases/examples/configure-elasticsearch/internal/security-audit-logs/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/opensearch_security_audit_logs.htm`** (for HTML format).

When the software is updated for the first time after the introduction of this new policy, any audit log indices that are older than the defined retention period are deleted.

SAS Allowance for Credit Loss

Before Deployment Commands

Revised acl_transform.yaml File

The acl_transform.yaml file for SAS Allowance for Credit Loss has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the acl_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Allowance for Credit Loss, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "acl_transform.yaml". If that phrase is present, then you have SAS Allowance for Credit Loss in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-acl/resources/acl_transform.yaml`** and change the file name to `acl_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-acl/resources/acl_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-acl/resources/acl_transform.yaml`**. You should now have an acl_transform.yaml file and an acl_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Allowance for Credit Loss README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the acl_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-acl/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_allowance_for_credit_loss.htm`** (for HTML format).
5. Use the acl_transform2.yaml file as a guide for filling in the rest of the values in the acl_transform.yaml file.
6. When the acl_transform.yaml file is complete, delete the acl_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Anti-Money Laundering

After Deployment Commands

New Functionality for SAS Anti-Money Laundering

The latest update for SAS Anti-Money Laundering contains new functionality in several areas. After you have completed the update, refer to the *SAS Anti-Money Laundering: Administrator's Guide* for descriptions of the new areas of functionality and how to configure them:

- Chapter 2, Provision and Configure.
- Chapter 4, Databases.
- Chapter 5, Roles and Capabilities for Users and Groups.

SAS Asset and Liability Management

Before Deployment Commands

Revised alm_transform.yaml File

The alm_transform.yaml file for SAS Asset and Liability Management has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the alm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Asset and Liability Management, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "alm_transform.yaml". If that phrase is present, then you have SAS Asset and Liability Management in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-alm/resources/alm_transform.yaml`** and change the file name to `alm_transform2.yaml`.

3 Copy the **$deploy/sas-bases/examples/sas-risk-cirrus-alm/resources/alm_transform.yaml** file to **$deploy/site-config/sas-risk-cirrus-alm/resources/alm_transform.yaml**. You should now have an alm_transform.yaml file and an alm_transform2.yaml file in the same directory.

4 Follow the steps in the SAS Asset and Liability Management README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the alm_transform.yaml file. The README file is located at **$deploy/sas-bases/examples/sas-risk-cirrus-alm/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_connection_settings_for_sas_asset_and_liability_management.htm** (for HTML format).

5 Use the alm_transform2.yaml file as a guide for filling in the rest of the values in the alm_transform.yaml file.

6 When the alm_transform.yaml file is complete, delete the alm_transform2.yaml file.

7 If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Dynamic Actuarial Modeling

Before Deployment Commands

Revised pcpricing_transform.yaml File

The pcpricing_transform.yaml file for SAS Dynamic Actuarial Modeling has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the pcpricing_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1 To determine if your deployment includes SAS Dynamic Actuarial Modeling, go to the base kustomization.yaml file (**$deploy/kustomization.yaml**). In the transformers block, search for "pcpricing_transform.yaml". If that phrase is present, then you have SAS Dynamic Actuarial Modeling in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.

2 Go to **$deploy/site-config/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml** and change the file name to `pcpricing_transform2.yaml`.

3 Copy the **$deploy/sas-bases/examples/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml** file to **$deploy/site-config/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml**. You should now have a pcpricing_transform.yaml file and a pcpricing_transform2.yaml file in the same directory.

4 Follow the steps in the SAS Dynamic Actuarial Modeling README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the pcpricing_transform.yaml file. The README file is located at **$deploy/sas-bases/examples/sas-risk-cirrus-pcpricing/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_dynamic_actuarial_modeling_for_deployment.htm** (for HTML format).

5 Use the pcpricing_transform2.yaml file as a guide for filling in the rest of the values in the pcpricing_transform.yaml file.

6 When the pcpricing_transform.yaml file is complete, delete the pcpricing_transform2.yaml file.

7 If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Event Stream Processing

Before Deployment Commands

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in "Restore Projects After Update".

Delete espserver Before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS Expected Credit Loss

Before Deployment Commands

Revised ecl_transform.yaml File

The ecl_transform.yaml file for SAS Expected Credit Loss has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the ecl_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Expected Credit Loss, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "ecl_transform.yaml". If that phrase is present, then you have SAS Expected Credit Loss in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-ecl/resources/ecl_transform.yaml`** and change the file name to `ecl_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-ecl/resources/ecl_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-ecl/resources/ecl_transform.yaml`**. You should now have an ecl_transform.yaml file and an ecl_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Expected Credit Loss README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the ecl_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-ecl/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_expected_credit_loss.htm`** (for HTML format).
5. Use the ecl_transform2.yaml file as a guide for filling in the rest of the values in the ecl_transform.yaml file.
6. When the ecl_transform.yaml file is complete, delete the ecl_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Insurance Capital Management

Before Deployment Commands

Revised icm_transform.yaml File

The icm_transform.yaml file for SAS Insurance Capital Management has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the icm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Insurance Capital Management, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "icm_transform.yaml". If that phrase is present, then you have SAS Insurance Capital Management in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-icm/resources/icm_transform.yaml`** and change the file name to `icm_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-icm/resources/icm_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-icm/resources/icm_transform.yaml`**. You should now have an icm_transform.yaml file and an icm_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Insurance Capital Management README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the icm_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-icm/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_insurance_capital_management.htm`** (for HTML format).
5. Use the icm_transform2.yaml file as a guide for filling in the rest of the values in the icm_transform.yaml file.
6. When the icm_transform.yaml file is complete, delete the icm_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Model Risk Management

Before Deployment Commands

Revised mrm_transform.yaml File

The mrm_transform.yaml file for SAS Model Risk Management has been modified with a new variable for increased functionality with Git repositories, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`. Use the following steps to get the new version of the mrm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Model Risk Management, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "mrm_transform.yaml". If that phrase is present, then you have SAS Model Risk Management in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-mrm/resources/mrm_transform.yaml`** and change the file name to `mrm_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-mrm/resources/mrm_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-mrm/resources/mrm_transform.yaml`**. You should now have a mrm_transform.yaml file and a mrm_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Model Risk Management README file to add the value for the new variable, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`, to the mrm_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-mrm/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_model_risk_management.htm`** (for HTML format).
5. Use the mrm_transform2.yaml file as a guide for filling in the rest of the values in the mrm_transform.yaml file.
6. When the mrm_transform.yaml file is complete, delete the mrm_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Regulatory Capital Management

Before Deployment Commands

Revised rcm_transform.yaml File

The rcm_transform.yaml file for SAS Regulatory Capital Management has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the rcm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Regulatory Capital Management, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "rcm_transform.yaml". If that phrase is present, then you have SAS Regulatory Capital Management in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-rcm/resources/rcm_transform.yaml`** and change the file name to `rcm_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-rcm/resources/rcm_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-rcm/resources/rcm_transform.yaml`**. You should now have an rcm_transform.yaml file and an rcm_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Regulatory Capital Management README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the rcm_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rcm/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_regulatory_capital_management.htm`** (for HTML format).
5. Use the rcm_transform2.yaml file as a guide for filling in the rest of the values in the rcm_transform.yaml file.
6. When the rcm_transform.yaml file is complete, delete the rcm_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS RFC Solution Configuration

Before Deployment Commands

Revised kafka_configuration_patch.yaml File

The kafka_configuration_patch.yaml file for SAS RFC Solution Configuration has been modified with new variables. Use the following steps to get the new version of the kafka_configuration_patch.yaml file in order to add the new variables while maintaining the rest of the content in that file.

1. To determine whether your deployment includes SAS RFC Solution Configuration, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the patches block, search for "kafka_configuration_patch.yaml". If that phrase is present, then you have SAS RFC Solution Configuration in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-rfc-solution-config/kafka-configuration-patch.yaml`** and change the file name to `kafka_configuration_patch2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-rfc-solution-config/configure-kafka/kafka-configuration-patch.yaml`** file to **`$deploy/site-config/sas-rfc-solution-config/kafka-configuration-patch.yaml`**. You should now have a kafka_configuration_patch.yaml file and a kafka_configuraon_patch2.yaml file in the same directory.
4. Follow the steps in the SAS RFC Solution Configuration README file to add the values for the new variables. The README file is located at **`$deploy/sas-bases/examples/sas-rfc-solution-config/configure-kafka/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/sas_rfc_solution_configuration_patch_to_communicate_with_apache_kafka.htm`** (for HTML format).
5. Use the kafka-configuration-patch2.yaml file as a guide for filling in the rest of the values in the kafka-configuration-patch.yaml file.
6. When the kafka-configuration-patch.yaml file is complete, delete the kafka-configuration-patch2.yaml file.

SAS Risk Cirrus Core

Before Deployment Commands

Revised core_transform.yaml File

The core_transform.yaml file for SAS Risk Cirrus Core has been modified with a new variable for increased functionality with Git repositories, `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the core_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Risk Cirrus Core, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "core_transform.yaml". If that phrase is present, then you have SAS Risk Cirrus Core in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `core_transform2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-risk-cirrus-rcc/resources/core_transform.yaml`**, change the file name to **`$deploy/site-config/sas-risk-cirrus-rcc/resources/core_transform2.yaml`**.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/core_transform.yaml`** file to the directory you used in step 2. You should now have a core_transform.yaml file and a core_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Risk Cirrus Core README file to add the values for the new variable, `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the core_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm`** (for HTML format).
5. Use the core_transform2.yaml file as a guide for filling in the rest of the values in the core_transform.yaml file.
6. When the core_transform.yaml file is complete, delete the core_transform2.yaml file.

SAS Risk Factor Manager

Before Deployment Commands

Revised rfm_transform.yaml File

The rfm_transform.yaml file for SAS Risk Factor Manager has been modified with a new variable for increased functionality with Git repositories, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`. Use the following steps to get the new version of the rfm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Risk Factor Manager, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "rfm_transform.yaml". If that phrase is present, then you have SAS Risk Factor Manager in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-rfm/resources/rfm_transform.yaml`** and change the file name to `rfm_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-rfm/resources/rfm_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-rfm/resources/rfm_transform.yaml`**. You should now have an rfm_transform.yaml file and an rfm_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Risk Factor Manager README file to add the value for the new variable, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`, to the rfm_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rfm/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_risk_factor_manager.htm`** (for HTML format).
5. Use the rfm_transform2.yaml file as a guide for filling in the rest of the values in the rfm_transform.yaml file.
6. When the rfm_transform.yaml file is complete, delete the rfm_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Risk Modeling

Before Deployment Commands

Revised rm_transform.yaml File

The rm_transform.yaml file for SAS Risk Modeling has been modified with a new variable for increased functionality with Git repositories, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`. Use the following steps to get the new version of the rm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Risk Modeling, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "rm_transform.yaml". If that phrase is present, then you have SAS Risk Modeling in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-rm/resources/rm_transform.yaml`** and change the file name to `rm_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-rm/resources/rm_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-rm/resources/rm_transform.yaml`**. You should now have an rm_transform.yaml file and an rm_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Risk Modeling README file to add the value for the new variable, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`, to the rm_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rm/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_risk_modeling.htm`** (for HTML format).
5. Use the rm_transform2.yaml file as a guide for filling in the rest of the values in the rm_transform.yaml file.
6. When the rm_transform.yaml file is complete, delete the rm_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Stress Testing

Before Deployment Commands

Revised st_transform.yaml File

The st_transform.yaml file for SAS Stress Testing has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the rm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Stress Testing, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "st_transform.yaml". If that phrase is present, then you have SAS Stress Testing in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-st/resources/st_transform.yaml`** and change the file name to `st_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-st/resources/st_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-st/resources/st_transform.yaml`**. You should now have an st_transform.yaml file and an st_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Stress Testing README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the st_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-st/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_stress_testing.htm`** (for HTML format).
5. Use the st_transform2.yaml file as a guide for filling in the rest of the values in the st_transform.yaml file.
6. When the st_transform.yaml file is complete, delete the st_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

2024.03 (May 2024)

General Deployment

Before Deployment Commands

Required Ingress Controller Configuration

If you are using ingress-nginx 1.12.*x* and later for your Kubernetes cluster, a change to the default configuration from NGINX will affect your deployment. For more information, see Required Ingress Controller Configuration.

Job Definition Rule Change Affects Group-Managed Service Account Functionality

Changes in the rules for job definitions prevent job requests with an associated job definition from being run by users other than the job creator or a SAS Administrator. Therefore group-managed service account functionality requires that job definitions are created in such a way that they can be shared with the group-managed service account, such as being created in a folder and then shared. Alternatively, you can create a rule that grants permission to a specific job definition, to all job definitions, or to a collection of job definitions. Perform one of the following set of steps depending on the restrictions you want to place on your service account.

Have a service account able to run any job

1. As a SAS Administrator, go to SAS Environment Manager → Rules.
2. Create a new rule with the following settings:
   - Object URI: `/jobDefinitions/definitions/*`
   - Rule Type: `Grant`
   - Permission: `Read`
   - Principal Type: `User`
   - Principal: *`name-of-service-account`*

Have a service account able to run a specific job

1. Go to SAS Environment Manager → Jobs and Flows
2. Select the **Scheduling** tab.

3 Right-click the appropriate job and choose **Properties**.

4 Under the **Job Definition** tab, determine if the ID field is populated. If so, copy the ID. If the field is not populated, then the job definition is part of the job request object and no further steps are necessary.

5 As a SAS Administrator, go to SAS Environment Manager → Rules.

6 Create a rule with the following settings:

- Object URI: `/jobDefinitions/definitions/`*`ID-from-step-4`*
- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

7 Repeat Step 3 through Step 6 for each job you want to run under the service account.

Have the service account able to run jobs with definitions stored in SAS Content

Perform one of the following tasks:

1 Use SAS Drive to "share" the job with the service account.

2 Ensure the service account has Read (convey) permission on the folder where the job definition is stored.

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 2024.03 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
```

Delete Crunchy Data Operator Upgrade Deployment

Note: If your deployment does not use an internal instance of PostgreSQL, skip this note.

Note: This deployment note is repeated from version 2023.10 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.

Version 2023.10 contains an upgrade to the PostgreSQL operator from Crunchy Data. While the upgrade happens without user intervention, in some cases the obsolete pgo-upgrade deployment fails to remove itself from the SAS Viya platform deployment. To determine if the pgo-upgrade deployment is present after the SAS Viya platform update:

```
kubectl -n name-of-namespace get deployment sas-crunchy5-postgres-operator-upgrade
```

If the output of the command indicates that the pgo-upgrade deployment is present, delete it:

```
kubectl -n name-of-namespace delete deployment sas-crunchy5-postgres-operator-upgrade
```

Crunchy Data seccompProfile Patch

Note: This deployment note is repeated from version 2023.10 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.

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' |
xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec":
{"template": {"spec":{"containers":[{"name": "database", "securityContext":
{"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

Analytics Resources

Before Deployment Commands

Scale Down Pods Before Update

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or the sas-orchestration tool, do not perform the steps in this note. Proceed with this note only if you are updating with the Kubernetes commands deployment method.

A number of offerings use the sas-analytics-resources service. That service requires that pods be scaled down to zero before the update.

1 Determine if your deployment is affected by this change.

   a If your deployment contains sas-analytics-resources, it is affected by this change. The change could also affect any sas-analytics-services pods in your deployment. Perform the following commands to determine if either type of pod is in your deployment:

   ```
   kubectl get pods -n name-of-namespace | grep sas-analytics-resources
   ```

   ```
   kubectl get pods -n name-of-namespace | grep sas-analytics-services
   ```

If the sas-analytics-resources command has no output, you do not need to perform the tasks described in this note. Otherwise, continue with Step 1b.

b To determine the version of your sas-analytics-resources pods, perform the following command:

```
kubectl get pods --no-headers --output=jsonpath='{range.items[*]}
{.metadata.labels.app} {":"} {.metadata.annotations.sas\.com/version}
{"\n"} {end} ' --selector 'app in (sas-analytics-resources)' -n name-of-
namespace
```

The output of this command takes this format:

```
sas-analytics-resources:x.y.z
```

The version number for the sas-analytics-resources pods is represented by `x.y.z`, which is three integers separated by decimal points. If the integer `x` is 22 or greater, you do not need to complete the tasks in this note. If `x` is less than 22, continue with Step 2.

2 For each type of pod returned in Step 1a, determine if Horizontal Pod Autoscaling (HPA) is being used.

a Perform the following commands, depending on which type of pod was found to be in your deployment in Step 1a on page 354:

```
kubectl get hpa --field-selector metadata.name=sas-analytics-resources -n
name-of-namespace
```

```
kubectl get hpa --field-selector metadata.name=sas-analytics-services -n
name-of-namespace
```

b The output looks like this:

```
NAME                     REFERENCE                           TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
sas-analytics-resources  Deployment/sas-analytics-resources  2%/80%    2         2         2          25h
```

Note the values for `MINPODS`, `MAXPODS`, and `REPLICAS` for each type of pod in your deployment.

- If the values for `MINPODS`, `MAXPODS`, or `REPLICAS` are greater than 1, HPA has been turned on for that type of pod and must be turned off. Use the following command:

  ```
  kubectl patch hpa pod-type --patch '{"spec": {"minReplicas": 1,
  "maxReplicas": 1}}' -n name-of-namespace
  ```

  Replace *pod-type* with `sas-analytics-resources` or `sas-analytics-services`, depending on which command produced the output. If both types of pod produce this output, you must run the command twice, once for each type.

- If the values for `MINPODS`, `MAXPODS`, and `REPLICAS` are each one, then HPA has not been turned on.

3 Scale the sas-analytics-resources pods down.

```
kubectl -n name-of-namespace scale deployment/sas-analytics-resources --
replicas=0
```

If you have sas-analytics-services pods, scale them down also.

```
kubectl -n name-of-namespace scale deployment/sas-analytics-services --
replicas=0
```

The sas-analytics-services service will be replaced by the sas-analytics-gateway service during the update. After the update, you will scale up the sas-analytics-resources service when you perform the task described in "Scale Up Pods After Update".

After Deployment Commands

Scale Up Pods After Update

Note: If you did not perform any of the steps in "Scale Down Pods Before Update", do not perform the procedure in this note.

After the update is completed, you must return the sas-analytics pods to their original state.

1 For each type of pod that had Horizontal Pod Autoscaling (HPA) turned on, perform the following command to turn HPA back on:

```
kubectl patch hpa pod-type --patch '{"spec": {"minReplicas": previous-value-
for-MINPODS, "maxReplicas": previous-value-for-MAXPODS}}' -n name-of-
namespace
```

For *pod-type*, use either `sas-analytics-resources` or `sas-analytics-gateway`, in place of sas-analytics-services pods. For *previous-value-for-MINPODS*, use the value for `MINPODS` that you noted in Step 2a of "Scale Down Pods Before Update". For *previous-value-for-MAXPODS*, use the value for `MAXPODS` that you noted in that same step.

2 Using the value you noted in Step 2a of "Scale Down Pods Before Update", scale the sas-analytics-resources service up to the number of replicas you had before the update.

```
kubectl -n name-of-namespace scale deployment/sas-analytics-resources --
replicas=number-of-replicas-before-update
```

If your deployment had sas-analytics-services pods before the update, scale the sas-analytics-gateway service up to the number of sas-analytics-service replicas you had before the update.

```
kubectl -n name-of-namespace scale deployment/sas-analytics-gateway --
replicas=number-of-sas-analytics-services-replicas-before-update
```

3 To confirm that the restarted pods are running, perform the following commands:

```
kubectl get pods -n name-of-namespace | grep sas-analytics-resources
```

```
kubectl get pods -n name-of-namespace | grep sas-analytics-gateway
```

You will get non-zero output for each command if that type of pod is running.

OpenSearch

Before Deployment Commands

OpenSearch Adds Configurable Parameters for Retaining Audit Log Indices

Note: This topic applies to internal instances of OpenSearch only.

OpenSearch is changing its audit log indices retention policy so that the indices are no longer retained indefinitely. Instead, the audit log indices are retained only for seven days by default. If you want to use a different retention period, OpenSearch has added a new transformer file, audit-log-retention-transformer.yaml, to configure the retention of audit log indices. For details about using this new file, see the README file at **`$deploy/sas-bases/examples/configure-elasticsearch/internal/security-audit-logs/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/opensearch_security_audit_logs.htm`** (for HTML format).

When the software is updated for the first time after the introduction of this new policy, any audit log indices that are older than the defined retention period are deleted.

SAS Allowance for Credit Loss

Before Deployment Commands

Support for Apache Airflow

SAS Allowance for Credit Loss now supports using Apache Airflow for Process Orchestration with an instance of PostgreSQL. For details about configuring Apache Airflow, see the README located at **`$deploy/sas-bases/examples/sas-airflow/metadata/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm`** (for HTML format).

SAS Anti-Money Laundering

After Deployment Commands

Remove Row-Level Bypass Privileges

SAS Anti-Money Laundering removes row-level bypass privileges during the update. After the update is complete, you should consider modifying your permissions to match the updated configuration. For details, see Updating from a Previous Cadence on the SAS Viya Platform

SAS Climate Risk Stress Testing

Before Deployment Commands

Support for Apache Airflow

SAS Climate Risk Stress Testing now supports using Apache Airflow for Process Orchestration with an instance of PostgreSQL. For details about configuring Apache Airflow, see the README located at **`$deploy/sas-bases/examples/sas-airflow/metadata/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm`** (for HTML format).

SAS Conversation Designer

Before Deployment Commands

Remove Transformer from kustomization yaml File

Note: This deployment note is repeated from version 2023.10 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.

SAS Conversation Designer has reached end of life. If you formerly had SAS Conversation Designer included in your SAS Viya platform deployment, it is no longer available to you after this update. Additionally, you must remove a reference to a SAS Conversation Designer transformer from your base kustomization.yaml file (**`$deploy/kustomization.yaml`**).

1. In the transformers block of the base kustomization.yaml file, search for a reference to `sas-bases/overlays/sas-natural-language-understanding/resources/disable-parser-transformer.yaml`. If the line is not present, no further actions are required.
2. Remove the `sas-bases/overlays/sas-natural-language-understanding/resources/disable-parser-transformer.yaml` line.
3. Save and close the base kustomization.yaml file.

SAS Credit Risk Stress Testing

Before Deployment Commands

Support for Apache Airflow

SAS Credit Risk Stress Testing now supports using Apache Airflow for Process Orchestration with an instance of PostgreSQL. For details about configuring Apache Airflow, see the README located at **`$deploy/sas-bases/examples/sas-airflow/metadata/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm`** (for HTML format).

SAS Event Stream Processing

Before Deployment Commands

Required Transformer for SAS Event Stream Manager

A transformer for SAS Event Stream Manager that enables direct file access to the persistent volume file system is required. To apply the transformer, modify the transformers block of the base kustomization.yaml ($deploy/kustomization.yaml) by adding the following highlighted line:

```
transformers:
- sas-bases/overlays/sas-event-stream-manager-app/data/data-transformer.yaml
...
```

For more information, see the README file located at **$deploy/sas-bases/overlays/sas-event-stream-manager-app/data/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configuring_file_storage_for_sas_event_stream_manager_projects.htm** (for HTML format).

Required Transformer for SAS Event Stream Processing Studio

A transformer for SAS Event Stream Processing Studio that enables direct file access to the persistent volume file system is required. To apply the transformer, modify the transformers block of the base kustomization.yaml file ($deploy/kustomization.yaml) by adding the following highlighted line:

```
transformers:
- sas-bases/overlays/sas-event-stream-processing-studio-app/data/data-transformer.yaml
...
```

For more information, see the README file located at **$deploy/sas-bases/overlays/sas-event-stream-processing-studio-app/data/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configuring_file_storage_for_sas_event_stream_processing_studio_projects.htm** (for HTML format).

Save espserver Resources

Before you update to version 2024.03, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to updaate your software, be sure to perform the tasks described in “Restore Projects After Update”.

Delete espserver before the Update

When you update to version 2024.03, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2024.03, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in “Save espserver Resources”, skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS Expected Credit Loss

Before Deployment Commands

Support for Apache Airflow

SAS Expected Credit Loss now supports using Apache Airflow for Process Orchestration with an instance of PostgreSQL. For details about configuring Apache Airflow, see the README located at **$deploy/sas-bases/examples/sas-airflow/metadata/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm** (for HTML format).

SAS Insurance Capital Management

Before Deployment Commands

Support for Apache Airflow

SAS Insurance Capital Management now supports using Apache Airflow for Process Orchestration for both internal and external instances of PostgreSQL.

If your current deployment uses Process Orchestration for an external instance of PostgreSQL, you must perform the following steps:

1 Run the following commands to delete resources that are being replaced with updated versions:

```
kubectl delete deployment sas-airflow-scheduler -n name-of-namespace
kubectl delete deployment sas-airflow-triggerer -n name-of-namespace
kubectl delete statefulset sas-airflow-worker -n name-of-namespace
kubectl delete job sas-airflow-create-nonadmin-user -n name-of-namespace
kubectl delete job sas-airflow-run-airflow-migrations -n name-of-namespace
```

2 Add the following line to top of the components block base kustomization.yaml file (**$deploy/kustomization.yaml**):

```
components:
- sas-bases/components/sas-airflow/external-airflow
```

Changing Process Orchestration from an external instance of PostgreSQL to an internal instance is not supported.

For details about Process Orchestration and Apache Airflow, see the README located at **$deploy/sas-bases/examples/sas-airflow/metadata/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm** (for HTML format).

SAS Insurance Contract Valuation Foundation

Before Deployment Commands

Support for Apache Airflow

SAS Insurance Contract Valuation Foundation now supports using Apache Airflow for Process Orchestration for both internal and external instances of PostgreSQL.

If your current deployment uses Process Orchestration for an external instance of PostgreSQL, you must perform the following steps:

1 Run the following commands to delete resources that are being replaced with updated versions:

```
kubectl delete deployment sas-airflow-scheduler -n name-of-namespace
kubectl delete deployment sas-airflow-triggerer -n name-of-namespace
kubectl delete statefulset sas-airflow-worker -n name-of-namespace
kubectl delete job sas-airflow-create-nonadmin-user -n name-of-namespace
kubectl delete job sas-airflow-run-airflow-migrations -n name-of-namespace
```

2 Add the following line to top of the components block base kustomization.yaml file (**$deploy/kustomization.yaml**):

```
components:
- sas-bases/components/sas-airflow/external-airflow
```

Changing Process Orchestration from an external instance of PostgreSQL to an internal instance is not supported.

For details about Process Orchestration and Apache Airflow, see the README located at **$deploy/sas-bases/examples/sas-airflow/metadata/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm** (for HTML format).

SAS Intelligent Planning

Before Deployment Commands

Configure qpStartupOptions for SAS Environment Manager

Note: This deployment note is repeated from version 2023.09 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.

SAS Environment Manager includes changes that require that the correct qpStartupOptions are in place before the update is performed.

1. From the applications menu (☰), click **Manage Environment**.
2. Click 🔧 to navigate to the Configuration workspace.
3. From the **View** list on the left, click **Definitions**.
4. In the **Filter** text box in the right pane, click **sas.planning.cas**.
5. Click the edit icon (✎).
6. In the Edit sas.planning.cas Configuration window, navigate to **qpStartupOptions** and verify that the **-XX:+UseParallelGC** option is present. If it is not present, add it.
7. Save your changes.

SAS Regulatory Capital Management

Before Deployment Commands

Support for Apache Airflow

SAS Regulatory Capital Management now supports using Apache Airflow for Process Orchestration for both internal and external instances of PostgreSQL.

If your current deployment uses Process Orchestration for an external instance of PostgreSQL, you must perform the following steps:

1 Run the following commands to delete resources that are being replaced with updated versions:

```
kubectl delete deployment sas-airflow-scheduler -n name-of-namespace
kubectl delete deployment sas-airflow-triggerer -n name-of-namespace
kubectl delete statefulset sas-airflow-worker -n name-of-namespace
kubectl delete job sas-airflow-create-nonadmin-user -n name-of-namespace
kubectl delete job sas-airflow-run-airflow-migrations -n name-of-namespace
```

2 Add the following line to top of the components block base kustomization.yaml file (**$deploy/kustomization.yaml**):

```
components:
- sas-bases/components/sas-airflow/external-airflow
```

Changing Process Orchestration from an external instance of PostgreSQL to an internal instance is not supported.

For details about Process Orchestration and Apache Airflow, see the README located at **$deploy/sas-bases/examples/sas-airflow/metadata/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm** (for HTML format).

SAS Stress Testing

Before Deployment Commands

Support for Apache Airflow

SAS Stress Testing now supports using Apache Airflow for Process Orchestration with an instance of PostgreSQL. For details about configuring Apache Airflow, see the README located at **$deploy/sas-bases/examples/sas-airflow/metadata/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm** (for HTML format).

SAS Viya with SingleStore

Before Deployment Commands

Change to SingleStore Cluster Definition

With version 2023.12 support for Kustomize moved to v5.1.1. This change requires revision to the base kustomization.yaml file (**$deploy/kustomization.yaml**) .

1 If you are unsure whether your deployment includes SAS Viya with SingleStore, search the base kustomization.yaml file for the string `singlestore`. If your search finds the string, continue with the steps in this deployment note. If your search does not find the string, then your deployment does not include SAS Viya with SingleStore and you should skip the rest of this note.

2 Remove the following line from your base kustomization.yaml file:

```
...
generators:
- site-config/sas-singlestore/sas-singlestore-secret.yaml
...
```

3 Also in the base kustomization.yaml file, add the following highlighted line to the components block:

```
...
components:
- sas-bases/components/sas-singlestore
- site-config/sas-singlestore
...
```

Ensure that the `- sas-bases/components/sas-singlestore` entry comes before any TLS components. Also ensure that the `- site-config/sas-singlestore` entry is after the `- sas-bases/components/sas-singlestore` entry. The `- site-config/sas-singlestore` entry merges the license/secret into the deployment when the update commands are performed.

The revised deployment step is located in Step 6 of the README file located at **$deploy/sas-bases/examples/sas-singlestore/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_singlestore_cluster_operator.htm** (for HTML format).

SAS/ACCESS Interface to MongoDB

Before Deployment Commands

Changes to Client Requirements

Because OpenSSL 1.0.2 libraries no longer exist in the images that SAS ships, the SAS Viya platform no longer supports executing versions of MongoDB that require OpenSSL 1.0.2. Therefore, ensure that your version of MongoDB supports a version of OpenSSL that is later than version 1.0.2.

SAS/ACCESS Interface to MySQL

Before Deployment Commands

Changes to Client Requirements

Because OpenSSL 1.0.2 libraries no longer exist in the images that SAS ships, the SAS Viya platform no longer supports executing versions of MySQL that require OpenSSL 1.0.2. Therefore, ensure that your version of MySQL supports a version of OpenSSL that is later than version 1.0.2.

2023.10 (November 2023)

General Deployment

Before Deployment Commands

SAS Workload 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.

IMPORTANT The SAS Workload Orchestrator requires that at least one node in the cluster contain the label `workload.sas.com/class=compute`. Not setting the label has critical repercussions. For example:

- Compute sessions and jobs cannot be launched.
- SAS Studio does not function.
- Batch and scheduled jobs do not run.

If your cluster does not support label or taints, you must disable SAS Workload Orchestrator before you perform an update.

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).

If you choose to keep SAS Workload Orchestrator enabled, its daemons require information about resources on the nodes that can be used to run jobs. In order to obtain accurate resource information, you must add a ClusterRole and a ClusterRoleBinding to the SAS Workload Orchestrator service account. For more information about the required ClusterRoles, see the README file located at **`$deploy/sas-bases/overlays/sas-workload-orchestrator/README.md`** (for

Markdown format) or at **$deploy/sas-bases/docs/cluster_privileges_for_sas_workload_orchestrator_service.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 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.

Updated Security Resource File

The openssl-generated-ingress-certificate.yaml file is used to generate the server identity certificate for the ingress controller and store it in a secret. Its use is optional in deployments of SAS Viya platform. However, its contents were revised in version 2023.10, so if you did use the file, it must be replaced with the new version.

1 In the base kustomization.yaml file (**`$deploy/kustomization.yaml`**), look in the resources block for a reference to the openssl-generated-ingress-certificate.yaml file. If there is no reference, you did not use the file and no further steps are necessary. If there is a reference, 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 `openssl-generated-ingress-certificate2.yaml`. For example, if the file is located at **`site-config/security/openssl-generated-ingress-certificate.yaml`**, change the file name to **`site-config/security/openssl-generated-ingress-certificate2.yaml`**.

3 Copy **`$deploy/sas-bases/examples/security/openssl-generated-ingress-certificate.yaml`** to the directory in step 2. You should now have a openssl-generated-ingress-certificate.yaml file and a openssl-generated-ingress-certificate2.yaml file in the same directory.

4 The file includes variables that can be modified for your deployment. Using the openssl-generated-ingress-certificate2.yaml file as a guide, revise the openssl-generated-ingress-certificate.yaml file so that it includes any values you used in the earlier version of the file.

5 After your revisions are complete, delete the openssl-generated-ingress-certificate2.yaml file.

6 Delete the sas-create-openssl-ingress-certificate Job:

```
kubectl delete job sas-create-openssl-ingress-certificate -n name-of-namespace
```

New File Required for OpenShift Container Registry

Note: If you are using a mirror registry other than OpenShift Container Registry, skip this note.

If you are deploying on Red Hat OpenShift using the OpenShift Container Registry, you should add a reference to a new file in the base kustomization.yaml file (**$deploy/kustomization.yaml**). In the transformers block in the base kustomization.yaml file, add a reference to the **sas-bases/overlays/sas-prepull/change-relpath.yaml** file. Here is an example:

```
transformers:
...
- sas-bases/overlays/sas-prepull/change-relpath.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 2023.10 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
```

Delete Crunchy Data Operator Upgrade Deployment

Note: If your deployment does not use an internal instance of PostgreSQL, skip this note.

Version 2023.10 contains an upgrade to the PostgreSQL operator from Crunchy Data. While the upgrade happens without user intervention, in some cases the obsolete pgo-upgrade deployment fails to remove itself from the SAS Viya platform deployment. To determine if the pgo-upgrade deployment is present after the SAS Viya platform update:

```
kubectl -n name-of-namespace get deployment sas-crunchy5-postgres-operator-upgrade
```

If the output of the command indicates that the pgo-upgrade deployment is present, delete it:

```
kubectl -n name-of-namespace delete deployment sas-crunchy5-postgres-operator-upgrade
```

Crunchy Data seccompProfile Patch

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' | xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec": {"template": {"spec":{"containers":[{"name": "database", "securityContext": {"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

SAS Conversation Designer

Before Deployment Commands

Remove Transformer from kustomization yaml File

SAS Conversation Designer has reached end of life. If you formerly had SAS Conversation Designer included in your SAS Viya platform deployment, it is no longer available to you after this update. Additionally, you must remove a reference to a SAS Conversation Designer transformer from your base kustomization.yaml file (**$deploy/kustomization.yaml**).

1. In the transformers block of the base kustomization.yaml file, search for a reference to `sas-bases/overlays/sas-natural-language-understanding/resources/disable-parser-transformer.yaml`. If the line is not present, no further actions are required.
2. Remove the `sas-bases/overlays/sas-natural-language-understanding/resources/disable-parser-transformer.yaml` line.
3. Save and close the base kustomization.yaml file.

SAS Dynamic Actuarial Modeling

Before Deployment Commands

Update Transformer File

In 2023.10, one of the SAS Dynamic Actuarial Modeling transformer files has been modified, requiring you to change a key and the associated value. Prior to 2023.10, 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 2023.10, 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 2023.10, 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 2023.10, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.10, 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.
```

SAS Intelligent Planning

Before Deployment Commands

Configure qpStartupOptions for SAS Environment Manager

SAS Environment Manager includes changes that require that the correct qpStartupOptions are in place before the update is performed.

1 From the applications menu (☰), click **Manage Environment**.

2 Click 🔧 to navigate to the Configuration workspace.

3 From the **View** list on the left, click **Definitions**.

4 In the **Filter** text box in the right pane, click **sas.planning.cas**.

5 Click the edit icon ( ).

6 In the Edit sas.planning.cas Configuration window, navigate to **qpStartupOptions** and verify that the **-XX:+UseParallelGC** option is present. If it is not present, add it.

7 Save your changes.

SAS Risk Modeling

Before Deployment Commands

Update Transformer File

In 2023.10, one of the SAS Risk Modeling transformer files has been modified, requiring you to change a key and the associated value. Prior to 2023.10, 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 2023.10, 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.

SAS Risk Modeling Requires a Git Repository

Starting with version 2023.10, SAS Risk Modeling requires a Git repository. The requirements for the repository are described at "Requirements for SAS® Risk Modeling" in *System Requirements for the SAS Viya Platform*.

Additionally, parameters in the **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/core_transform.yaml** file should be revised to account for the addition of the Git repository. `SAS_RISK_CIRRUS_CODE_LIB_GIT_SETUP_NEEDED` should be set to `Y`. The following parameters should be set to the appropriate values for your Git repository:

- `SAS_RISK_CIRRUS_CODE_LIB_REPO_NAME`
- `SAS_RISK_CIRRUS_CODE_LIB_REPO_USER`
- `SAS_RISK_CIRRUS_CODE_LIB_REPO_EMAIL`
- `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_FULL_URL`

For the details about these parameters, see the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

Configuration for Workflow Service Tasks

Starting with version 2023.10, SAS Risk Modeling requires that certain values in the SAS Risk Cirrus Core configuration file be set so that the offering can take advantage of the workflow service tasks. In the **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/core_transform.yaml** file, the following parameters must be set to the appropriate values:

- `SAS_RISK_CIRRUS_SET_WORKFLOW_SERVICE_ACCOUNT_FLG`
- `SAS_RISK_CIRRUS_WORKFLOW_DEFAULT_SERVICE_ACCOUNT`

For the details about these parameters, see the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

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 Tasks*.

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 131.

   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*.

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.

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 121. 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 "Requirements for SAS Infrastructure Data Server" 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. SAS Event Stream Processing Studio must be configured to use the PV. For more information, see "Managing Persistent Volumes".

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 "Requirements for SAS Infrastructure Data Server" 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 "Requirements for SAS Infrastructure Data Server" 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 "Configure the Certificate Generator" 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: General Management of 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 141), 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 445 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 143.

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 443. 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.

3

Stable Releases

2026.09 (September 2026) **157**
Additional Documentation 157
General Deployment 158

2026.08 (August 2026) **164**
Additional Documentation 164
General Deployment 164
SAS Model Repository Service 167

2026.07 (July 2026) **168**
Additional Documentation 168
General Deployment 168

2026.06 (June 2026) **172**
Additional Documentation 172
General Deployment 172
SAS/ACCESS Interface to Microsoft SQL Server 174

2026.05 (May 2026) **174**
Additional Documentation 174
General Deployment 175

2026.04 (April 2026) **179**
Additional Documentation 179
General Deployment 179
SAS Detection Architecture 180

2026.03 (March 2026) **183**
Additional Documentation 183
General Deployment 183
SAS RFC Solution Configuration 189
SAS Visual Investigator 190

2026.02 (February 2026) **191**
Additional Documentation 191
General Deployment 191

2026.01 (January 2026) **199**
Additional Documentation 199
General Deployment 200

2025.12 (December 2025) **201**
Additional Documentation 201
General Deployment 202

2025.11 (November 2025) **208**
Additional Documentation 208
General Deployment 209
SAS Detection Architecture 210
SAS Model Publish Service 211
SAS Studio 212

2025.10 (October 2025) **213**
Additional Documentation 213
General Deployment 214
SAS Decisions Runtime Builder 215
SAS Risk Cirrus KRM Service 216

2025.09 (September 2025) **217**
Additional Documentation 217
General Deployment 218
SAS Decisions Runtime Builder 219
SAS Model Publish Service 220

2025.08 (August 2025) **221**
Additional Documentation 221
General Deployment 221
Model Studio 223
SAS Data Quality 224
SAS Decisions Runtime Builder 225
SAS SpeedyStore 227

2025.07 (July 2025) **228**
Additional Documentation 228
General Deployment 228
SAS Decisions Runtime Builder 229
SAS Micro Analytic Service 233

2025.06 (June 2025) **234**
Additional Documentation 234
General Deployment 235
SAS Decisions Runtime Builder 240
SAS Micro Analytic Service 243

2025.05 (May 2025) **244**
Additional Documentation 244
General Deployment 244

2025.04 (April 2025) **248**
Additional Documentation 248
General Deployment 248
SAS Detection Architecture 254

2025.03 (March 2025) **256**
Additional Documentation 256
General Deployment 256
SAS Event Stream Processing 262
SAS/ACCESS to Impala 263
SAS/ACCESS to Spark 264

2025.02 (February 2025) **264**
Additional Documentation 264
General Deployment 265
SAS Allowance for Credit Loss 268

SAS Asset and Liability Management 269
SAS Dynamic Actuarial Modeling 270
SAS Event Stream Processing 270
SAS Expected Credit Loss 272
SAS Insurance Capital Management 273
SAS Insurance Contract Valuation Foundation 274
SAS Integrated Regulatory Reporting 275
SAS Model Risk Management 276
SAS Regulatory Capital Management 277
SAS Risk Cirrus Core 278
SAS Risk Factor Manager 279
SAS Risk Modeling 280
SAS Risk Offerings 280
SAS Stress Testing 281

2025.01 (January 2025) **282**
Additional Documentation 282
General Deployment 282
SAS Event Stream Processing 285

2024.12 (December 2024) **286**
Additional Documentation 286
General Deployment 287
SAS Event Stream Processing 289
SAS Model Repository Service 290

2024.11 (November 2024) **291**
Additional Documentation 291
General Deployment 291
SAS Event Stream Processing 294

2024.10 (October 2024) **297**
Additional Documentation 297
General Deployment 298
SAS Event Stream Processing 300
SAS Risk Cirrus Core 301

2024.09 (September 2024) **303**
Additional Documentation 303
General Deployment 303
SAS Event Stream Processing 309

2024.08 (August 2024) **311**
General Deployment 311
SAS Event Stream Processing 312

2024.07 (July 2024) **314**
General Deployment 314
OpenSearch 316
SAS Event Stream Processing 316
SAS RFC Solution Configuration 318

2024.06 (June 2024) **319**
General Deployment 319
SAS Event Stream Processing 320
SAS Regulatory Capital Management 321

2024.05 (May 2024) **322**
General Deployment 322

SAS Allowance for Credit Loss 324
SAS Anti-Money Laundering 325
SAS Asset and Liability Management 325
SAS Dynamic Actuarial Modeling 326
SAS Event Stream Processing 327
SAS Expected Credit Loss 329
SAS Insurance Capital Management 330
SAS Model Risk Management 331
SAS Risk Cirrus Core 332
SAS Risk Factor Manager 333
SAS Risk Modeling 334
SAS Stress Testing 335

2024.04 (April 2024) **336**
General Deployment 336
SAS Event Stream Processing 339

2024.03 (March 2024) **341**
General Deployment 341
OpenSearch 344
SAS Allowance for Credit Loss 344
SAS Climate Risk Stress Testing 345
SAS Credit Risk Stress Testing 345
SAS Event Stream Processing 346
SAS Expected Credit Loss 348
SAS Insurance Capital Management 348
SAS Insurance Contract Valuation Foundation 349
SAS Regulatory Capital Management 350
SAS Stress Testing 351

2024.02 (February 2024) **351**
General Deployment 351
Analytics Resources 354
SAS Anti-Money Laundering 357
SAS Event Stream Processing 357
SAS/ACCESS Interface to MongoDB 358
SAS/ACCESS Interface to MySQL 358

2024.01 (January 2024) **359**
General Deployment 359
SAS Event Stream Processing 362

2023.12 (December 2023) **363**
General Deployment 363
SAS Event Stream Processing 366
SAS Viya with SingleStore 367

2023.11 (November 2023) **368**
General Deployment 368
SAS Conversation Designer 370
SAS Event Stream Processing 371
SAS Intelligent Planning 372

2023.10 (October 2023) **373**
General Deployment 373
SAS Conversation Designer 377
SAS Event Stream Processing 378
SAS Intelligent Planning 378

SAS Risk Modeling 379

2023.09 (September 2023) **380**
General Deployment 380
SAS Event Stream Processing 382
SAS Intelligent Planning 382
SAS Risk Modeling 383

2023.08 (August 2023) **384**
General Deployment 384
SAS Event Stream Processing 387
SAS Risk Modeling 387

2023.07 (July 2023) **388**
General Deployment 388
SAS Event Stream Processing 390
SAS Risk Modeling 390

2023.06 (June 2023) **391**
General Deployment 391
SAS Event Stream Processing 393
SAS Risk Modeling 393

2023.05 (May 2023) **395**
General Deployment 395
SAS Event Stream Processing 397

2023.04 (April 2023) **397**
General Deployment 397
SAS Business Orchestration Services 401
SAS Dynamic Actuarial Modeling 401
SAS Event Stream Processing 403

2023.03 (March 2023) **403**
General Deployment 403
SAS Event Stream Processing 407
SAS Model Risk Management 408

2023.02 (February 2023) **409**
General Deployment 409
SAS Event Stream Processing 412

2023.01 (January 2023) **412**
General Deployment 412
SAS Event Stream Processing 414
SAS Risk Engine 414

2022.12 (December 2022) **415**
General Deployment 415
SAS Event Stream Processing 417
SAS Inventory Optimization 418

2022.11 (November 2022) **418**
General Deployment 418
SAS Dynamic Actuarial Modeling 420
SAS Event Stream Processing 421
SAS Model Risk Management 421

2022.10 (October 2022) **422**
General Deployment 422

SAS Event Stream Processing 433

2022.09 (September 2022) **435**
General Deployment 435
SAS Event Stream Processing 438

2022.1.4 (August 2022) **438**
General Deployment 438
SAS Event Stream Processing 441

2022.1.3 (July 2022) **442**
General Deployment 442
SAS Event Stream Processing 446

2022.1.2 (June 2022) **448**
General Deployment 448
SAS Event Stream Processing 449

2022.1.1 (May 2022) **449**
General Deployment 449
SAS Event Stream Processing 452

2021.2.6 (April 2022) **453**
General Deployment 453
SAS Visual Analytics 456
SAS Event Stream Processing 456

2021.2.5 (March 2022) **457**
General Deployment 457
SAS Event Stream Processing 458

2021.2.4 (February 2022) **458**
General Deployment 458
SAS Event Stream Processing 461

2021.2.3 (January 2022) **461**
General Deployment 461
SAS Event Stream Processing 464

2021.2.2 (December 2021) **464**
General Deployment 464
SAS Event Stream Processing 466

2021.2.1 (November 2021) **467**
General Deployment 467
SAS Event Stream Processing 470

2021.1.6 (October 2021) **470**
General Deployment 470
SAS Event Stream Processing 473

2021.1.5 (September 2021) **474**
General Deployment 474
SAS Event Stream Processing 476

2021.1.4 (August 2021) **476**
General Deployment 476
SAS Event Stream Processing 484

2021.1.3 (July 2021) **484**
General Deployment 484
SAS Event Stream Processing 488

2021.1.2 (June 2021) **489**
General Deployment 489
SAS Event Stream Processing 491
SAS Model Publish 492
SAS Risk Engine 492

2021.1.1 (May 2021) **493**
General Deployment 493
SAS Event Stream Processing 494
SAS Information Catalog 495

2020.1.5 (April 2021) **495**
General Deployment 495
SAS Event Stream Processing 497

2020.1.4 (March 2021) **497**
General Deployment 497
SAS Event Stream Processing 502

2020.1.3 (February 2021) **502**
General Deployment 502
SAS Event Stream Processing 505

2020.1.2 (January 2021) **505**
General Deployment 505
SAS Event Stream Processing 506

2020.1.1 (December 2020) **506**
General Deployment 507
SAS Event Stream Processing 508
SAS Model Publish Service 509

2026.09 (September 2026)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Upgrade to PostgreSQL 17

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

IMPORTANT Before you attempt the procedure described in this deployment note, complete the procedures described in every other deployment note within your current update path that is listed in a "Before Deployment Commands" section. This deployment note should be the last procedure you perform from a deployment note in a "Before Deployment Commands" section. After you have completed the procedure in this deployment note, perform the deployment/update commands and then proceed with any deployment notes listed in an "After Deployment Commands" section.

SAS is upgrading the version of PostgreSQL used in the internal instances of PostgreSQL (Crunchy Postgres from Crunchy Data) from version 16 to version 17. This deployment note contains the necessary steps to perform the PostgreSQL upgrade on a running SAS Viya platform deployment.

Note: The amount of time the upgrade takes varies based on the amount of data you have and the underlying storage speed. For each 100 Gi in the database, add approximately one hour to the time required for the upgrade.

1 Take a full pgBackRest backup. For more information, see "Take an Ad Hoc pgBackRest Backup " in *SAS Viya Platform: Infrastructure Servers*.

2 Consider changing the PostgreSQLdata storage to block storage. Block storage might improve PostgreSQL performance and, more importantly, it can help you avoid a potential PostgreSQL upgrade issue on NFS storage. For details, see "Change the Storage Class of the Data Pod" in *SAS Viya Platform: Infrastructure Servers*.

Note: If you are using the SAS Viya Platform Deployment Operator or the sas-orchestration command to update your software, you should skip Step 3 through Step 7. Only perform those steps if you are using the Kubernetes command deployment method.

3 **(Kubernetes command deployment method only)** Create the site.yaml file.

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

4 **(Kubernetes command deployment method only)** Copy the upgrade script from the examples directory and insure it has execute permissions:

```
cp $deploy/sas-bases/examples/crunchydata/postgres-upgrade/postgres-upgrade.sh ./
```

```
chmod +x postgres-upgrade.sh
```

5 **(Kubernetes command deployment method only)** Using bash, run the upgrade script with the indicated parameters:

Note: The upgrade script requires the `yq` command (version 4) to be installed. For information about how to install the `yq` command, see https://github.com/mikefarah/yq.

```
./postgres-upgrade.sh \
location-of-admin-kubeconfig \
name-of-namespace \
$deploy/site.yaml \
log-debug \
2>&1
```

Running the script sends log files to the **`postgresupgrade-logs-16-to-17`** in your `$HOME` directory.

6 **(Kubernetes command deployment method only)** After the PostgreSQL upgrade has completed, the vacuumdb task is performed and the PostgreSQL extensions are upgraded. The PostgreSQL cluster can be used normally while these actions are performed, including continuing to update your SAS Viya platform deployment.

7 **(Kubernetes command deployment method only)** Change the directory to the $deploy directory. Delete the site.yaml file that was created in Step 3.

8 After you perform the deployment commands to update your software, perform the tasks described at "PostgreSQL 17 Post-Upgrade Processing".

CAUTION

The PostgreSQL upgrade process is not idempotent, so the manual script or the lifecycle operation should not be run more than once. If any issues occur during their execution, contact SAS Technical Support to remediate the incomplete upgrade instead of retrying the upgrade. Retrying or attempting to fix the issue might make the PostgreSQL cluster unrecoverable.

Revised database-transformer.yaml File

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

The update includes an updated transformer file that might be required to replace the existing version of the same transformer in your deployment. Perform the following steps to determine if you need to replace your existing transformer and to perform that replacement as required.

1 In the transformers block of your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the following string:

```
dataserver-transformer.yaml
```

You should have one or two instances of the string. Make a list of the file name and path of any file that includes that string in its name.

2 Go to the location described by the string you located in Step 1. If you found more than one string in Step 1., you will be directed to return to this step for each string you found.

3 Open the file on your list and search for the following string:

```
- op: add
  path: "/metadata/annotations/sas.com~1database-server-location"
  value: external
```

If that string is present, then you do not need to get a new version of the file. If additional files remain in your list, return to Step 2 and continue with the next file in the list. If the string is not present, continue to Step 4.

4 Change the name of the file at that location by adding a "2" before the period. For example, if you found a file named `platform-postrgres-dataserver-transformer.yaml`, rename it `platform-postrgres-dataserver-transformer2.yaml`.

5 Copy the file at **`$deploy/sas-bases/examples/`*`oracle-or-postgres`*`/dataserver-transformer.yaml`** to the location of the file you renamed in Step 4. For the variable, use `oracle` if your SAS Infrastructure Data Server is running on Oracle. Use `postgres` if your SAS Infrastructure Data Server is running on PostgreSQL.

6 Rename the copied file with the original name of the file. For example, if the file you found in Step 1 is `platform-postgres-dataserver-transformer.yaml`, use that name for the copied file. You should have two files with similar names, the only difference being the number "2", such as `platform-postrgres-dataserver-transformer.yaml` and `platform-postrgres-dataserver-transformer2.yaml`.

7 Compare the copied file and the original file. Use the version of the file with a "2" in the name as a guide for the revisions to be made to the copied file in order to match the configuration settings used in your existing deployment. Replace the variables in the copied file with the values used in the original file.

8 When the revisions are complete, delete the version of the file with a "2" in the file name.

9 If additional files remain in the list you created in Step 1, return to Step 2 and repeat the process for the next file on the list. When all the files on the list have been processed, continue to the next deployment note.

After Deployment Commands

PostgreSQL 17 Post-Upgrade Processing

Note: These steps for configuring PostgreSQL after it has been upgraded are applicable to all deployment methods. If you used the Kubernetes commands deployment method, and if you did not perform the steps described in "Upgrade to PostgreSQL 17". do not perform the steps in this deployment note.

CAUTION
The PostgreSQL upgrade process is not idempotent, so the manual script or the lifecycle operation should not be run more than once. If any issues occurred during their execution, contact SAS Technical Support to remediate the incomplete upgrade instead of retrying the upgrade. Retrying or attempting to fix the issue might make the PostgreSQL cluster unrecoverable.

For more information about post-upgrade processing, see the following documentation from Crunchy Data: Step 6: Complete the Post-Upgrade Tasks.

Note: If applicable, repeat the following steps for both the platform PostgreSQL cluster and the CDS PostgreSQL cluster.

1 To determine the names of your PostgreSQL clusters, run the following command:

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

The output of this command is a cluster name that includes "platform" (such as sas-crunchy-platform-postgres) and, if you have a CDS PostgreSQL cluster, a cluster name that includes "cds" (such as sas-crunchy-cds-postgres).

In the remaining steps in this deployment note, the variable *cluster-name* should be replaced by the actual cluster name. If you have two clusters, run the command with the *cluster-name* variable twice, once for each cluster.

2 (Optional) Delete the old data directory.

a Exec into the primary pod of each cluster.

```
POD1=$(kubectl -n name-of-namespace get pods --selector="postgres-
operator.crunchydata.com/cluster=cluster-name,postgres-
operator.crunchydata.com/role=master" -o name)
```

```
kubectl exec -it $POD1 -c database -n name-of-namespace -- bash
```

After the second command, you will have a bash prompt, which you can use to perform additional steps.

b Check the PostgreSQL version:

```
psql -c "select version()"
```

If the version is not 17.*x*, the PostgreSQL upgrade might have failed. Stop and contact SAS Technical Support.

c Determine if pg16 and pg16_wal directories are present in the leader pod:

```
ls -la /pgdata
```

If the output of the command does not include information about pg16 and pg16_wal directories, you might have had a failover. You should instead check replica nodes for the pg16 directories. Use the `patronictl list` command to show the list of replicas.

d Execute the following commands to rename or to remove the old data directory. Do so at your discretion:

```
mv -v /pgdata/pg16 /pgdata/pg16.old
```

```
mv -v /pgdata/pg16_wal /pgdata/pg16_wal.old
```

or

```
rm -rf /pgdata/pg16
```

```
rm -rf /pgdata/pg16_wal
```

e Close the shell session by entering the following command:

```
exit
```

3 (Optional) Clean up objects that are no longer needed:

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

```
kubectl -n name-of-namespace annotate postgrescluster cluster-name postgres-operator.crunchydata.com/allow-upgrade-
```

Note: The "-" at the end of the second command is not a typographical error and should be included.

IMPORTANT If you have upgraded PostgreSQL in a namespace that is in a cluster that contains other namespaces still at PostgreSQL 16, update those namespaces with the latest patches available for the version of the SAS Viya platform that is deployed in each namespace. For example, if the cluster contains a namespace at version 2026.08, find the latest patch for 2026.08 and update the namespace with that patch. This task might require coordination among namespace administrators.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

2026.08 (August 2026)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Revised odbc.ini File

Note: This deployment note is repeated from version 2026.06 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.

Updates to SAS/ACCESS Interface to Amazon Redshift, SAS/ACCESS Interface to Greenplum, and SAS/ACCESS Interface to Microsoft SQL Server require the odbc.ini file in a deployment that includes those products to be revised. Here are the instructions to determine if your deployment is affected and to revise the file.

1 Perform the following command:

```
%put %quote(%sysget(ODBCINI));
```

If you get no output, you do not use the odbc.ini file in your deployment and you can continue to the next deployment note. If you get output, continue with Step 2.

2 Go to the location described in the output of the command from Step 1. Open the odbc.ini file there and delete the following entries:

- `SSLLibname`
- `CryptoLibname`

Fast Execution Resource Management Infrastructure Changes

Fast Execution Resource Management Infrastructure (FERMI) is a process orchestration tool used primarily by the Risk solutions. The Redis service associated with FERMI is moving from a single-instance topology to a cluster topology, aligning with the Redis service associated with the SAS Viya platform. Cluster mode requires a headless service that provides stable per-pod DNS so that nodes can discover each other during cluster formation. The move to a cluster topology requires changes to Kubernetes files that are immutable. As a result, the StatefulSet, service, and PVC for the single-instance topology must be deleted so that it can be created again as a cluster by the update.

1 Determine the topology of your FERMI deployment:

```
kubectl get statefulset sas-fermi-redis-server -n name-of-namespace -o
jsonpath='{.spec.serviceName}{"\n"}'
```

- If the output includes the following error, FERMI service is not deployed and you should skip the rest of this deployment note.

  ```
  Error from server (NotFound): statefulsets.apps "sas-fermi-redis-server" not found
  ```

- If the output from the command includes `sas-fermi-redis-server-cluster`, your deployment already uses the clustered topology, and you should skip the rest of this deployment note.
- If the output includes *sas-fermi-redis-server* or if there is no output, continue with Step 2.

2 Perform the following commands to delete the single-instance artifacts:

```
kubectl delete statefulset sas-fermi-redis-server -n name-of-namespace --
ignore-not-found
```

```
kubectl delete service sas-fermi-redis-server -n name-of-namespace --ignore-
not-found
```

```
kubectl delete pvc sas-viya-fermi-redis-data-volume-sas-fermi-redis-server-0
-n name-of-namespace --ignore-not-found
```

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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 Model Repository Service

Before Deployment Commands

Apply and Bind New Security Context Constraint

Note: If your deployment is not running on Red Hat OpenShift, skip this deployment note.

The SAS Model Repository (sas-model-manager) pod requires the nfs volume type if the Python or R environment is made available through an NFS mount. NFS volumes are not permitted by the default restricted SCC, so an SCC that includes nfs in the allowed volumes section is required. The SCC used for this purpose before this update has been replaced. Users who had applied and bound the previous SCC to an existing deployment must apply and bind the new SCC. Optionally, they can also remove the previous SCC.

1 To determine whether you are using the original SCC, run the following command:

```
kubectl get scc sas-model-repository
```

If the output does not include `sas-model-repository`, you do not have the original SCC applied and you can skip the rest of this note. If `sas-model-repository` is in the output, continue with Step 2.

2 Apply the new SCC:

```
kubectl apply -f sas-bases/overlays/sas-model-repository/service-account/
sas-model-manager-scc.yaml
```

3 Bind the SCC to the sas-model-manager service account:

```
oc -n name-of-namespace adm policy add-scc-to-user sas-model-manager -z sas-
model-manager
```

4 (Optional) To remove the obsolete SCC and binding, first confirm that no other workload depends on them. Then perform the following commands:

```
oc -n name-of-namespace adm policy remove-scc-from-user sas-model-repository
-z sas-model-repository
```

```
oc delete scc sas-model-repository
```

2026.07 (July 2026)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Revised odbc.ini File

Note: This deployment note is repeated from version 2026.06 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.

Updates to SAS/ACCESS Interface to Amazon Redshift, SAS/ACCESS Interface to Greenplum, and SAS/ACCESS Interface to Microsoft SQL Server require the odbc.ini file in a deployment that includes those products to be revised. Here are the instructions to determine if your deployment is affected and to revise the file.

1 Perform the following command:

```
%put %quote(%sysget(ODBCINI));
```

If you get no output, you do not use the odbc.ini file in your deployment and you can continue to the next deployment note. If you get output, continue with Step 2.

2 Go to the location described in the output of the command from Step 1. Open the odbc.ini file there and delete the following entries:

- `SSLLibname`
- `CryptoLibname`

Revised SAS Configurator for Open Source Transformer

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised to manage the list of supported Python versions. The transformer is 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 at **`$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 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.

5 After your revisions are complete, delete the change-configuration2.yaml file.

For more information about the change-configuration.yaml file, see "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

Removal of Support for Process Orchestration and Its Internal Apache Airflow

SAS Viya platform might contain an internal Apache Airflow that was used by Process Orchestration and is no longer available to your deployment. When you perform an update, any YAML files for Process Orchestration will be removed from your deployment assets. To ensure that your base kustomization.yaml file does not contain any references to files that are no longer in your deployment:

1 Search your base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for any instances of "airflow".

2 Delete any line that appears in your search results.

Note: This step might require a little more thought than just removing lines that meet the criteria. You must also consider the context. For example, assume you have the following content in your kustomization.yaml file:

```
secretGenerator:
- name: sas-airflow-metadata
  envs:
  - site-config/sas-airflow/metadata/metadata.env
...
```

If there is more content in the sas-airflow-metadata section of the secretGenerator block than the lines that contain "airflow", you cannot simply delete those two lines because doing so breaks the code. Instead, you should remove the entire sas-airflow-metadata block to keep the code functional while removing references to "airflow".

3 After you perform the deployment commands to update your software, perform the tasks in "Removal of Support for Process Orchestration and Associated Apache Airflow Artifacts".

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

Removal of Support for Process Orchestration and Associated Apache Airflow Artifacts

Note: There are steps for the removal of Process Orchestration and associated Apache Airflow pieces in the "Before Deployment Commands" portion of the deployment notes for 2026.07. If you have not already, see "Removal of Support for Process Orchestration and Its Internal Apache Airflow".

If you removed any lines from the base kustomization.yaml file when performing the tasks described in "Removal of Support for Process Orchestration and Its Internal Apache Airflow", you can also remove the Apache Airflow database and user role:

1. Drop the Apache Airflow database:

   ```
   drop database airflow-database-name;
   ```

   For *airflow-database-name*, use the following:

   internal instance of PostgreSQL: `airflowdb`

   external instance of PostgreSQL: `airflow_db`

2. Drop the Apache Airflow user role:

   ```
   drop user airflow-database-user-name;
   ```

   For *airflow-database-user-name*, use the following:

   internal instance of PostgreSQL: `airflowuser`

   external instance of PostgreSQL: `airflow_user`

3. Delete any remaining PVCs:

   a. Get the names of any PVCs that remain:

      ```
      kubectl -n name-of-namespace get pvc | grep sas-redis-db-sas-airflow
      ```

   b. Delete the PVCs that are listed by the last command

      ```
      kubectl -n name-of-namespace delete pvc name-of-PVC-from-step-3a
      ```

      Repeat this command for each PVC that is returned by the command in step 3a.

2026.06 (June 2026)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Revised odbc.ini File

Updates to SAS/ACCESS Interface to Amazon Redshift, SAS/ACCESS Interface to Greenplum, and SAS/ACCESS Interface to Microsoft SQL Server require the odbc.ini file in a deployment that includes those products to be revised. Here are the instructions to determine if your deployment is affected and to revise the file.

1 Perform the following command:

```
%put %quote(%sysget(ODBCINI));
```

If you get no output, you do not use the odbc.ini file in your deployment and you can continue to the next deployment note. If you get output, continue with Step 2.

2 Go to the location described in the output of the command from Step 1. Open the odbc.ini file there and delete the following entries:

   - `SSLLibname`
   - `CryptoLibname`

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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/ACCESS Interface to Microsoft SQL Server

After Deployment Commands

ACCELERATOR Procedure Requires New Properties in the LIBNAME Connection to Microsoft SQL Server

If you run PROC ACCELERATOR in Azure Synapse, changes are required due to an update in the DataDirect Microsoft SQL Server ODBC driver. Additional properties are required for connections using SAS/ACCESS Interface to Microsoft SQL Server. For the details, see "Publish, Run, and Delete a Model in Azure Synapse" in *SAS In-Database Products: User's Guide*.

2026.05 (May 2026)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

After Deployment Commands

CAS REST API Authentication

By default, the CAS server now uses OAuth for calls from external resources to CAS REST. Previously, HTTP Basic Authentication was enabled by default for the CAS REST port.

> **IMPORTANT** After the update, Basic Authentication will no longer be available for calls to CAS REST. For OAuth to authorize REST API calls to CAS, you must create an OAuth token for those calls. The steps to create an OAuth token are described at CAS REST.

OAuth is more secure than Basic Authentication, so SAS recommends it for calls to CAS REST. However, to give you time to adjust existing code to a new authentication method, you have the option to enable Basic Authentication so that existing code will continue to work. Basic Authentication can then be disabled when code is converted to use OAuth.

Here are the steps to enable Basic Authentication:

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

```
- sas-bases/overlays/sas-programming-environment/enable-admin-script-access.yaml
```

If the line is present, skip to Step 4. If the line is not present, continue with Step 2.

2 Add a reference to `- sas-bases/overlays/sas-programming-environment/enable-admin-script-access.yaml` to the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**). Here is an example:

```
transformers:
...
- sas-bases/overlays/sas-programming-environment/enable-admin-script-access.yaml
...
```

Note: For more information about the enable-admin-script-access.yaml file, see the README file located at **$deploy/sas-bases/overlays/sas-programming-environment/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_programming_environment_configuration_tasks.htm** (for HTML format).

3 Update the software to pick up the change in the base kustomization.yaml. Use the same commands that you use to update the software. For the specific steps, see "General Process" in *SAS Viya Platform: Deployment Guide*.

4 Use SAS Environment Manager to set the CAS option `cas.ALLOWRESTBASICAUTH` to `true` (case-sensitive) for the service name that matches the name of your CAS deployment instance (the default name is `default: cas-shared-default`). In the configuration instance, change the setting to `sas.cas.instance.config: config`.

   For more information about the `cas.ALLOWRESTBASICAUTH` option, see "cas.ALLOWRESTBASICAUTH='FALSE'" in *SAS Viya Platform: SAS Cloud Analytic Services*. For more information about using SAS Environment Manager to set options, see "Edit Configuration Instances" in *SAS Viya Platform: SAS Cloud Analytic Services*.

5 Restart the CAS servers with the following command:

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

Note: In any update where you perform the tasks described in this deployment note, you do not need to perform the tasks described in "Update CAS Servers" on page 176. Step 5 applies the same command as that deployment note.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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 Security Certificate Framework Can Generate RSA Key Pair and Certificate

The ability to use the SAS Security Certificate Framework (sas-certframe) to generate the RSA key pair and certificate used for signing JSON web tokens (JWTs) and SAML 2.0 protocol messages is now available. The new functionality adds a second sas-certframe init container to the deployment for sas-logon-app. The new init container is configured to reuse the same key across all replicas of sas-logon-app. Furthermore, the key is not linked to the lifecycle of the pod, so it does not change on pod restarts. If you want to use your own key and certificate, you can still do so.

New deployments of SAS Viya platform automatically have this new functionality. Existing deployments continue to use the signing key and certificate stored in the SAS Configuration Server (Consul) key/values. However, you can choose to convert to sas-certframe to manage them. To convert to the new functionality:

Note: The following steps require an outage for your deployment because stopping and restarting your services is required. Be sure to account for the outage in your planning if you choose to perform these steps.

1. All deployments should remove the existing signing key and certificate by performing the steps described at "(OIDC) Remove Signing Key and Certificate from SAS Configuration Server" in *SAS Viya Platform: Authentication*.
2. If you are using SAML for logins, remove the existing signing key and certificate by performing the steps described at "(SAML) Remove Signing Key and Certificate from SAS Configuration Server" in *SAS Viya Platform: Authentication*.
3. If you have created a configuration for `sas.logon.jwt` and provided values for `signingKey` and `signingCert`, you must remove the existing values from the configuration so that they are removed from the Consul key/values.
    a. Log on to SAS Environment Manager as an administrator, and navigate to the Configuration page. For more information, see "Navigation" in *SAS Environment Manager: User's Guide*.
    b. On the **Configuration** page, select **Definitions** from the drop-down list.
    c. In the **Definitions** list, select **sas.logon.jwt**.
    d. Delete the entries in the **signingCert** and **signingKey** fields.

e Save the configuration. The sas-logon-app pods should restart automatically.

f All services must be restarted to pick up the new JSON web keys

1 .Stop the servers with the following command:

```
kubectl create job sas-stop-all-`date +%s` --from cronjobs/sas-stop-all -
n name-of-namespace
```

2 Start the servers with the following command:

```
kubectl create job sas-start-all-`date +%s` --from cronjobs/sas-start-all
-n name-of-namespace
```

SAS Logon Manager uses the key and certificate generated by sas-certframe to sign JWTs. To override these in order to use your own key and certificate, see Step 4. If you do not have your own key and certificate, you have completed the deployment note.

Note: If you are using OIDC logins through an external provider and have enabled the **jwtclientAuthentication** option in the **sas.logon.oauth.providers** configuration, you might need to update the public key or certificate the external OIDC provider has for SAS Viya. Obtain the signing key and/or certificate using the URIs in the following table:

Table 3.1 *URIs for certificate and Keys*

Type	URI	Description
JSON web keys	**`/SASLogon/token_keys`**	JSON contains the RSA public key components used to validate JWTs, and associated x5c certificate. See RFC 7517.
OIDC Provider Configuration	**`/SASLogon/.well-known/openid-configuration`**	JSON containing the OIDC configuration information, including the jwks_uri. See OIDC Connect Discovery.
SAML Metadata	**`/SASLogon/saml/metadata`**	XML containing the SAML 2.0 service provider (SP) metadata, including the X509Certificate that can be used to validate SAML messages.

4 By default, sas-certframe generates the RSA key pair and certificate that are used for signing JWTs and SAML protocol messages. However, you can replace

the key pair and certificate that the SAS Viya platform uses for signing JWTs and SAML messages with your own keys and certificate. For the instructions to replace the RSA key pair and certificate with your own secure signing key and signing certificate, see the README file located at **$deploy/sas-bases/examples/sas-logon-app/jwt/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_jwt_signing_key_and_certificate.htm** file (for HTML format).

2026.04 (April 2026)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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 Detection Architecture

Before Deployment Commands

Password Properties Moved to Secrets

SAS Detection Engine, which is part of SAS Detection Architecture, previously kept password information in the configuration.env file. With this release, the information is moving to a new secrets.env file because of the new Profile/Signature storage in Oracle feature. Keeping the deployment running after the update requires a revised kustomization.yaml file and a new secrets.env file.

1 To determine if your deployment uses SAS Detection Architecture, look in your deployment assets for the following file:

```
$deploy/site-config/sas-detection/kustomization.yaml
```

If the file is present, continue with Step 2. If the file is not present, then SAS Deployment Architecture is not included in your deployment, and you can skip the rest of this deployment note.

2 In **$deploy/site-config/sas-detection**, rename the kustomization.yaml file to kustomization2.yaml. Also rename the configration.env file to configuration2.env.

3 Copy **$deploy/sas-bases/examples/sas-detection/kustomization.yaml**, **$deploy/sas-bases/examples/sas-detection/configuration.env**, and **$deploy/sas-bases/examples/sas-detection/secrets.env** to **$deploy/site-config/sas-detection**.

4 The **$deploy/site-config/sas-detection** directory should now contain a kustomization.yaml file and a kustomization2.yaml file. Compare the kustomization.yaml and kustomization2.yaml files. Use the kustomization2.yaml file as a guide for the revisions to be made to the kustomization.yaml file in order to match the configuration settings used in your existing deployment.

5 The **$deploy/site-config/sas-detection** directory should now contain a configuration.env file and a configuration2.env file. Compare the configuration.env and configuration2.env files. Use the configuration2.env file as a guide for the revisions to be made to the configuration.env file in order to match the configuration settings used in your existing deployment.

Notice that the properties in the SAS Container Runtime Properties section of the file have increased in the update. For example, the configuration.env file now contains Oracle MEH Properties that must be set in order to use Oracle for profile storage. For descriptions of the new properties and values appropriate for them, see the README file located at **$deploy/sas-bases/examples/sas-detection/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_detection_engine_configuration.htm** (for HTML format).

Note: Properties previously used to define the Redis connection for signature storage have been expanded for use with either list data or connection to the Redis MEH. The following table describes the connection between the previous properties and the new ones included in the update. If a property in the first column is present in your configuration2.env file, replace it with the corresponding property from the second or third column, depending on if you are using MEH or list data.

Table 3.2 *Updated Property Names*

Property Name Before 2026.04	New Property Name for MEH	New Property Name for List Data
SAS_REDIS_HOST	SAS_REDIS_MEH_HOST	SAS_REDIS_LIST_HOST
SAS_REDIS_PORT	SAS_REDIS_MEH_PORT	SAS_REDIS_LIST_PORT
SAS_REDIS_AUTH_USER	SAS_REDIS_MEH_AUTH_USER	SAS_REDIS_LIST_AUTH_USER

Property Name Before 2026.04	New Property Name for MEH	New Property Name for List Data
SAS_REDIS_AUTH_PASS	SAS_REDIS_MEH_AUTH_PASS	SAS_REDIS_LIST_AUTH_PASS
SAS_REDIS_TLS	SAS_REDIS_MEH_TLS	SAS_REDIS_LIST_TLS
SAS_REDIS_CA_CERT	SAS_REDIS_MEH_CA_CERT	SAS_REDIS_LIST_CA_CERT
SAS_REDIS_TRUST_CERT_PATH	SAS_REDIS_MEH_TRUST_CERT_PATH	SAS_REDIS_LIST_TRUST_CERT_PATH
SAS_REDIS_CLIENT_CERT_FILE	SAS_REDIS_MEH_CLIENT_CERT_FILE	SAS_REDIS_LIST_CLIENT_CERT_FILE
SAS_REDIS_CLIENT_PRIV_KEY_FILE	SAS_REDIS_MEH_CLIENT_PRIV_KEY_FILE	SAS_REDIS_LIST_CLIENT_PRIV_KEY_FILE
SAS_REDIS_SERVER_DOMAIN	SAS_REDIS_MEH_SERVER_DOMAIN	SAS_REDIS_LIST_SERVER_DOMAIN

6 In the secrets.env file that you copied to this directory in Step 3, replace the variables with the appropriate values. The variables are identified by the doubled curly brackets and spaces (such as {{ SAS-REDIS-MEH-AUTH-PASS }}). For more information about the variables in the secrets.env file and the appropriate values for those variables, see the README file located at **$deploy/sas-bases/examples/sas-detection/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_detection_engine_configuration.htm** (for HTML format).

7 After your revisions are complete, delete the kustomization2.yaml file and the configuration2.env file.

8 Because the deployment of SAS Detection Engine is independent from the SAS Viya platform, the usual deployment steps will not complete the update. Instead, you must perform the deployment commands described in the "Installation" section of the README file located at **$deploy/sas-bases/examples/sas-detection/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_detection_engine_configuration.htm** (for HTML format).

2026.03 (March 2026)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Upgrade to OpenSearch 3.x

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

SAS is upgrading the version of OpenSearch used in the internal instances of OpenSearch to version 3.3.2. OpenSearch 3.x contains a breaking change: indices created under Elasticsearch 7.x or OpenSearch 1.x are not compatible with OpenSearch 3.x and must be reindexed before the upgrade can proceed. The breaking change can affect deployments that were initially created on LTS 2023.03 or Stable 2023.05 or earlier, and which have not since reindexed application data. This deployment note contains the necessary steps to identify and remediate incompatible indices so that the OpenSearch upgrade can proceed.

Create List of Incompatible Indices

1 Determine the name of the pods that have OpenSearch running on them:

```
kubectl get pods -l app=sas-opendistro -n name-of-namespace
```

Save the list of pods for Step 3.

2 Retrieve the sas-opendistro-sasadmin-secret Kubernetes password:

```
kubectl -n name-of-namespace get secret sas-opendistro-sasadmin-secret -o
json |jq -r .data.password| base64 -d
```

Save the password for Step 3.

3 Query the OpenSearch settings API for indices with created versions that begin with "`1.`" or "`7.`".

- Perform the following command on deployments using full-stack TLS:

```
kubectl exec name-of-OpenDistro-pod -n name-of-namespace -- \
curl -k -s "https://localhost:9200/*/_settings?human" \
-u sasadmin:admin-password | \
jq 'to_entries[] | select((.key | startswith(".") | not) and
((.value.settings.index.version.created_string // "") | test("^(7|
1)\\.")))  | .key'
```

- Perform the following command on deployments using front-door TLS or no TLS:

```
kubectl exec name-of-OpenDistro-pod -n name-of-namespace -- \
curl -k -s "http://localhost:9200/*/_settings?human" \
-u sasadmin:admin-password | \
jq 'to_entries[] | select((.key | startswith(".") | not) and
((.value.settings.index.version.created_string // "") | test("^(7|
1)\\.")))  | .key'
```

In the command that you choose to use, replace *name-of-OpenDistro-pod* with the name of any OpenDistro pod that is in the output of the command in Step 1. Also, replace *admin-password* with the password found in Step 2.

4 The result of performing either of the commands is a list of indices. In that list, the following kinds of indices may be ignored:

- OpenSearch system indices prefixed by a period, such as `.opendistro_security`
- Indices that begin with `sand__entity_link__` or `sand__shape__`

If no indices or no other kind of indices are returned, no manual action is required and you should skip the rest of this deployment note. If other indices are returned, continue with the next section.

Reindex Incompatible Data Indices

1 The method for reindexing varies based on the SAS product. The associated product for an index is identified by its prefix. Use the appropriate method for each indexed identified in Table 3.3.

Table 3.3 *Indices and Their Reindexing Method*

Index Prefix	Associated Product	Note	Reindexing Method
`sand__`	SAS Visual Investigator	`sand__entity__link__` and `sand__shape__` indices are reindexed by the OpenSearch operator during the upgrade. Any other index that begins with `sand__` should use the described reindexing method.	See "Re-index All Entities" in *SAS Visual Investigator: Administrator's Guide*.
`viya_catalog__`	SAS Data Governance (formerly known as SAS Information Catalog before 2026.06)		When you restart sas-catalog-services, re-create the search indices. The associated search indices should be rebuilt to ensure that they are fully populated. See the steps described in the table at "Respond to Notifications" in *SAS Data Governance: Administrator's Guide*.
`security-auditlog-` and `top_queries-`	OpenSearch Log Indices	By default, `security-auditlog-` and `top_queries-` indices are removed after seven days. However, a transformer can be applied to retain security auditlogs for longer. If your	See Reindex data and Delete Index API.

Index Prefix	Associated Product	Note	Reindexing Method
		deployment uses this transformer and audit logs are present, logs created with OpenSearch 1.x must be removed or reindexed.	
`viya_cirrus__`	SAS Risk Cirrus	No version 1.x `viya_cirrus__` indices should be present. If you see these indices in the output, contact SAS Technical Support.	See "Get Support" in *Getting Started with SAS Viya Platform Operations*.

2 If you are unsure how a product-specific index should be rebuilt, consult the product documentation or contact SAS Technical Support.

Validate That No Incompatible Indices Remain

1 Re-run the query from "Create List of Incompatible Indices" to confirm that no application data indices created under Elasticsearch 7.x or OpenSearch 1.x remain.

2 After you have confirmed that all application data indices have been reindexed, perform the deployment commands to update your software and then perform the tasks described at "OpenSearch 3 Post-Upgrade Validation" on page 188.

Revised SAS Configurator for Open Source Transformer

Note: The deployment notes for 2026.01 and 2026.02 include content nearly identical to this note for SAS Configurator for Open Source. If your update includes either of those releases and you perform the SAS Configurator for Open Source deployment note in them, you can skip this deployment note because an update will include all the changes to the affected transformer file.

If you have performed the SAS Configurator for Open Source deployment note for 2026.01 or 2026.02 in an earlier update, you must perform the deployment note for 2026.03 since the update is different even though the steps to include it are similar.

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised to manage the list of supported Python versions. The transformer is 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 at **$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 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.

5 After your revisions are complete, delete the change-configuration2.yaml file.

For more information about the change-configuration.yaml file, see "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

New PROC R Requires Revisions to Resource File

SAS is introducing PROC R for use with the R external language. To enable PROC R, you must revise a resource file before you update your SAS Viya platform deployment.

1 In the resources block of your base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the following string:

```
- site-config/sas-open-source-config/r
```

If that string is not present, no further steps are required. If the string is present, continue with Step 2.

2 Go to **$deploy/site-config/sas-open-source-config/r** and change the kustomization.yaml file name to `kustomization2.yaml`.

3 Copy the file at **$deploy/sas-bases/examples/sas-open-source-config/r/kustomization.yaml** to **$deploy/site-config/sas-open-source-config/r**. You should now have a kustomization.yaml file and a kustomization2.yaml file in the same directory.

4 Compare the kustomization.yaml and kustomization2.yaml files. Use the kustomization2.yaml file as a guide for the revisions to be made to the kustomization.yaml file in order to match the configuration settings used in your existing deployment.

5 After your revisions are complete, delete the kustomization2.yaml file.

For more information about PROC R and configuring for it, see "Configure R Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

OpenSearch 3 Post-Upgrade Validation

Verify that the OpenSearch upgrade has completed successfully:

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

In the output of the command, the`UPGRADE-STATUS` column should display `up-to-date` and the `SEARCH-ENGINE-VERSION` column should display `opensearch 3.x`. Here is an example:

```
NAME             UPGRADE-STATUS   SEARCH-ENGINE-VERSION
sas-opendistro   up-to-date       opensearch 3.x
```

If the output displays `upgrade blocked (upgrade requires data reindex)`, incompatible indices still remain. Go to "Create List of Incompatible Indices" on page 184 to identify and reindex the remaining indices. The upgrade will resume automatically when all incompatible indices have been corrected.

SAS RFC Solution Configuration

Before Deployment Commands

Reorganized Directory Structure

In 2026.03, SAS RFC Solution adds new functionality to its workloads that requires that the deployment assets for the product be reorganized. This note describes how to modify an existing deployment of SAS RFC Solution to accommodate the new functionality.

1 To determine if you have SAS RFC Solution workloads in your deployment, go to the resources block of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). Search for the following line:

```
- site-config/sas-rfc-solution-workloads/install
```

If that line is present, then you have SAS RFC Solution workloads in your deployment and should continue with Step 2. If the line is not present, skip the rest of this deployment note.

2 To configure SAS RFC Solution workloads with the new configuration files, follow the instructions in the README file located at **`$deploy/sas-bases/examples/sas-rfc-solution-workloads/ README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configuring_sas_rfc_solution_workloads.htm`** (for HTML format).

Note: After you copy the configuration files to the **`$deploy/site-config`** directory, you will have two sets of nearly identical configuration files. The existing deployment uses the configuration files located at **`$deploy/site-config/sas-rfc-solution-workloads/install`**. The updated deployment will use the configuration files located at **`$deploy/site-config/sas-rfc-solution-workloads/install/main`**.

The instructions in the README include modifying the files that you move to the **$deploy/site-config/sas-rfc-solution-workloads/install/main** directory. Because the old versions of the configuration files can still be found at **$deploy/site-config/sas-rfc-solution-workloads/install**, SAS recommends that you use the old version as a basis for any changes you make to the new version of a file. This ensures that the configuration properties in the updated deployment are the same as in the existing deployment.

For example, assume that the README instructs you to "Update the **$deploy/site-config/sas-rfc-solution-workloads/install/main/namespace.yaml** file by replacing the {{ NAMESPACE }} value." You can find the value for the {{ NAMESPACE }} variable in that file by referring to the **$deploy/site-config/sas-rfc-solution-workloads/install/namespace.yaml** file.

3 To prevent potential confusion in the future, SAS recommends that you delete the old versions of the files. Delete the files in **$deploy/sas-bases/examples/sas-rfc-solution-workloads/install** and **$deploy/site-config/sas-rfc-solution-workloads/install**.

SAS Visual Investigator

After Deployment Commands

Removal of System Table Storage for Context Records Replication

SAS Visual Investigator deprecated the option to store replication records in a system table in 2024.09. With this release, that functionality has been removed. For more information about how flows are affected and how to retain the ability to make updates to a flow that is using system replication, see "Removal of System Table Storage for Context Records Replication" in *What's New in SAS Visual Investigator*.

2026.02 (February 2026)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

New Configuration File Format for Backup and Restore

The configuration files for backup, restore, and migration have been revised. The new format simplifies future updates and the maintenance of backup and restore configurations.

Use the following steps to update the reformatted files.

1 In the transformers block of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the following string:

```
sas-backup-pv-copy-cleanup-job-modify-resources-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 2.

a Note the location for the file described in the kustomization.yaml entry.

b Copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the values for the {{ PV-COPY-CLEANUP-CPU-LIMIT }} and {{ PV-COPY-CLEANUP-MEMORY-LIMIT }} variables. You can use the file you found in Step 1 as a guide for the values to use in the copied file.

d In the base kustomization.yaml file, add the following content to the configMapGenerator block:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e Add the following content to the transformers block of the base kustomization.yaml file:

```
transformers:
...
- sas-bases/overlays/backup/sas-backup-pv-copy-cleanup-job-modify-resources-transformer.yaml
...
```

f Delete the file described by the line you found in Step 1. In the base kustomization.yaml file, remove the line with the string that you found in Step 1.

2 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-scheduled-backup-all-sources-change-default-schedule
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 3.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the value for the {{ SCHEDULE-BACKUP-ALL-SOURCES-CRON-EXPRESSION }} variable. You can use the file you found in Step 2 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-scheduled-backup-all-sources-change-default-schedule.yaml
...
```

f Delete the file described by the line you found in Step 2. In the base kustomization.yaml file, remove the line with the string that you found in Step 2.

3 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-scheduled-backup-incr-job-change-default-schedule
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 4.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the value for the {{ SCHEDULE-BACKUP-INCR-CRON-EXPRESSION }} variable. You can use the file you found in Step 3 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-scheduled-backup-incr-job-change-default-schedule.yaml
...
```

f Delete the file described by the line you found in Step 3. In the base kustomization.yaml file, remove the line with the string that you found in Step 3.

4 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-backup-job-modify-resources-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 5.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the values for the {{ BACKUP-JOB-CPU-LIMIT }} and { BACKUP-JOB-MEMORY-LIMIT }} variables. You can use the file you found in Step 4 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-scheduled-backup-job-modify-
resources-transformer.yaml
...
```

f Delete the file described by the line you found in Step 4. In the base kustomization.yaml file, remove the line with the string that you found in Step 4.

5 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-common-backup-data-storage-size-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 6.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the value for the {{ STORAGE-SIZE-IN-GB }} variable. You can use the file you found in Step 5 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
```

```
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-common-backup-data-storage-size-
transformer.yaml
...
```

f Delete the file described by the line you found in Step 5. In the base kustomization.yaml file, remove the line with the string that you found in Step 5.

6 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-common-backup-data-storage-class-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 7.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the value for the {{ STORAGE-CLASS-NAME }} variable. You can use the file you found in Step 6 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-common-backup-data-storage-class-
transformer.yaml
...
```

f Delete the file described by the line you found in Step 6. In the base kustomization.yaml file, remove the line with the string that you found in Step 6.

7 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-scheduled-backup-job-change-default-backup-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 8.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/backup/sas-backup-config.env** to **$deploy/site-config/backup/sas-backup-config.env**.

c In the copied file, update the value for the {{ SCHEDULE-BACKUP-CRON-EXPRESSION }} variable. You can use the file you found in Step 7 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-backup-config
  envs:
    - site-config/backup/sas-backup-config.env
...
```

e In the transformers block of the base kustomization.yaml file, add the following content:

```
transformers:
...
- sas-bases/overlays/backup/sas-scheduled-backup-job-change-default-backup-transformer.yaml
...
```

f Delete the file described by the line you found in Step 7. In the base kustomization.yaml file, remove the line with the string that you found in Step 7.

8 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-restore-job-modify-resources-transformer
```

If the string is present, continue with the following substeps. If it is not present, skip to Step 9.

a Note the location for the file described in the kustomization.yaml entry.

b Copy the **$deploy/sas-bases/examples/restore/sas-restore-config.env** to **$deploy/site-config/restore/sas-restore-config.env**.

c In the copied file, update the values for the {{ RESTORE-JOB-CPU-LIMIT }} and {{ RESTORE-JOB-MEMORY-LIMIT }} variables. You can use the file you found in Step 8 as a guide for the values to use in the copied file.

d In the base kustomization.yaml file, add the following content to the configMapGenerator block:

```
configMapGenerator:
...
```

```
- name: sas-restore-config
  envs:
    - site-config/restore/sas-restore-config.env
...
```

e Add the following content to the transformers block of the base kustomization.yaml file:

```
transformers:
...
- sas-bases/overlays/restore/sas-restore-job-modify-resources-
transformer.yaml
...
```

f Delete the file described by the line you found in Step 8. In the base kustomization.yaml file, remove the line with the string that you found in Step 8.

9 In the transformers block of the base kustomization.yaml file, search for the following string:

```
sas-migration-job-modify-resources-transformer
```

If the string is present, continue with the following substeps. If it is not present, you have completed this deployment note.

a Note the location for the file described in the kustomization.yaml entry.

b If you have not already, copy the **$deploy/sas-bases/examples/restore/sas-restore-config.env** to **$deploy/site-config/restore/sas-restore-config.env**.

c In the copied file, update the values for the {{ MIGRATION-JOB-CPU-LIMIT }} and {{ MIGRATION-JOB-MEMORY-LIMIT }} variables. You can use the file you found in Step 9 as a guide for the values to use in the copied file.

d If you have not already, add the following content to the configMapGenerator block in the base kustomization.yaml file:

```
configMapGenerator:
...
- name: sas-restore-config
  envs:
    - site-config/restore/sas-restore-config.env
...
```

e Add the following content to the transformers block of the base kustomization.yaml file:

```
transformers:
...
- sas-bases/overlays/migration/sas-migration-job-modify-resources-
transformer.yaml
...
```

f Delete the file described by the line you found in Step 9. In the base kustomization.yaml file, remove the line with the string that you found in Step 9.

Revised SAS Configurator for Open Source Transformer

Note: This deployment note is repeated from 2026.01 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 to support new R packages. The change-configuration.yaml file is 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. 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.
5. After your revisions are complete, delete the change-configuration2.yaml file.

For more information about the change-configuration.yaml file, see "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

2026.01 (January 2026)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Revised SAS Configurator for Open Source Transformer

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised to support new R packages. The change-configuration.yaml file is 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. 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.
5. After your revisions are complete, delete the change-configuration2.yaml file.

For more information about the change-configuration.yaml file, see "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

2025.12 (December 2025)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

New Architecture for SAS Workload Orchestrator

Previously, the manager and server Kubernetes objects for SAS Workload Orchestrator have been delivered as a single entity. With this release they are being delivered as separate entities, namely, sas-workload-orchestrator for the manager StatefulSet and sas-workload-orchestrator-server for the server DaemonSet. This change requires new configuration files. In addition, the existing files used to configure SAS Workload Orchestrator have been revised to use the new configuration file guidelines and format. The new format will simplify future updates.

To find and use the new configuration files, follow the instructions at "Configure SAS Workload Orchestrator" in *SAS Viya Platform: Deployment Guide*. Here is a summary of the most common changes to be made:

> **IMPORTANT** Even if you have disabled your existing deployment of SAS Workload Orchestrator, you must revise the configuration files for the new architecture in order that SAS Workload Orchestrator remains disabled.

1 If SAS Workload Orchestrator is disabled in your existing deployment and you want to keep it disabled, you must disable it with the new configuration files.

   a Confirm that SAS Workload Orchestrator is disabled by searching for the following line in the base kustomization.yaml file (**`$deploy/kustomization.yaml`**):

   ```
   - sas-bases/overlays/sas-workload-orchestrator/enable-disable/sas-workload-orchestrator-disable-patch-transformer.yaml
   ```

   If the line is present, continue with Step 1b. If the line is not present, skip to Step 2.

   b Add the following line to the base kustomization.yaml file immediately after the line you just searched for.

   ```
   - sas-bases/overlays/sas-workload-orchestrator-server/enable-disable/sas-workload-orchestrator-server-disable-patch-transformer.yaml
   ```

   After you revise the base kustomization.yaml file, there are no further steps. You should skip the rest of this deployment note.

2 Determine if your current deployment configured cluster-roles.

   a Search the base kustomization.yaml file for the following line:

   ```
   - sas-bases/overlays/sas-workload-orchestrator
   ```

   If the line is present, continue with Step 2b. If the line is not present skip to Step 3.

   b In the base kustomization.yaml file, delete the following line:

   ```
   - sas-bases/overlays/sas-workload-orchestrator
   ```

   In the place of the line you just deleted, insert the following two lines:

   ```
   - sas-bases/overlays/sas-workload-orchestrator/cluster-role
   ```

   ```
   - sas-bases/overlays/sas-workload-orchestrator-server/cluster-role
   ```

3 Search for the presence of a configuration for SAS Workload Orchestrator pod requests and limits:

   a In the base kustomization.yaml file, search for the following string:

   ```
   sas-workload-orchestrator-statefulset-resources.yaml
   ```

   If the string is present, note the location of the file and continue with this step. If it is not present, skip to Step 4.

   b Copy the **/$deploy/sas-bases/examples/sas-workload-orchestrator/configure/configuration.env** file to your site-config directory as **/$deploy/site-config/sas-workload-orchestrator/configure/configuration.env**.

   c In the `Statefulset (manager) pod resource configuration` section of the copied **/sas-workload-orchestrator/configure/configuration.env** file, modify the entries to match your current configuration. For each entry you want to modify, uncomment the line by removing the hashtag and then replacing the variable with the appropriate value. You can use the file in the location that you noted in Step 3a as a guide for the values used in the existing deployment.

   d In the base kustomization.yaml file, delete the line in which you found the string you searched for in Step 3a. SAS recommends that you also delete the file described by that line since you have copied the information to the new configuration file.

4 Search for the presence of a custom initial SAS Workload Orchestrator configuration:

   a In the base kustomization.yaml file, search for the following string:

   ```
   sas-workload-orchestrator-initial-configuration-change.yaml
   ```

   If the string is present, note the location of the file and continue with this step. If it is not present, skip to Step 5.

   b If you have not already, copy the **/$deploy/sas-bases/examples/sas-workload-orchestrator/configure/configuration.env** file to your site-config directory as **/$deploy/site-config/sas-workload-orchestrator/configure/configuration.env**.

c In the `Initial Grid Configuration` section of the copied **/sas-workload-orchestrator/configure/configuration.env** file, uncomment the `initial-configuration-json` property and modify the value to match what is in the file from Step 4a.

d In the base kustomization.yaml file, delete the line in which you found the string you searched for in Step 4a. SAS recommends that you also delete the file described by that line since you have copied the information to the new configuration file.

5 If you modifed the configuration.env file in Step 3 or Step 4, continue with this step. If you did not modify the configuration.env file, skip to Step 6.

a In the base kustomization.yaml file, add a merge behavior to merge in the settings. Here is an example:

```
configMapGenerator:
...
- behavior: merge
  envs:
  - /site-config/sas-workload-orchestrator/configure/
configuration.env
  name: sas-workload-orchestrator-user-configuration
```

b Also in the base kustomization.yaml file, add a reference to the **/$deploy/sas-bases/overlays/sas-workload-orchestrator/configure/transformers.yaml** to the transformers block. Here is an example:

```
transformers:
...
- sas-bases/overlays/sas-workload-orchestrator/configure/
transformers.yaml
```

6 Search for the presence of a configuration for SAS Workload Orchestrator pod requests and limits:

a In the base kustomization.yaml file, search for the following string:

```
sas-workload-orchestrator-daemonset-resources.yaml
```

If the string is present, note the location of the file and continue with this step. If it is not present, skip to Step 7.

b Copy the **/$deploy/sas-bases/examples/sas-workload-orchestrator-server/configure/configuration.env** file to your site-config directory as **/$deploy/site-config/sas-workload-orchestrator-server/configure/configuration.env**.

c In the `Daemonset (server) pod resource configuration` section of the copied **/sas-workload-orchestrator-server/configure/configuration.env** file, modify the entries to match your current configuration. For each entry you want to modify, uncomment the line by removing the hashtag and then replacing the variable with the appropriate value. You can use the file in the location that you noted in Step 6a as a guide for the values used in the existing deployment.

d In the base kustomization.yaml file, delete the line in which you found the string you searched for in Step 6a. SAS recommends that you also delete the

file described by that line since you have copied the information to the new configuration file.

7 Search for the presence of a custom initial custom Prometheus Pushgateway:

   a In the base kustomization.yaml file, search for the following string:

   ```
   sas-workload-orchestrator-prometheus-push-gateway-url.yaml
   ```

   If the string is present, note the location of the file and continue with this step. If it is not present, skip to Step 8.

   b If you have not already, copy the **/$deploy/sas-bases/examples/sas-workload-orchestrator-server/configure/configuration.env** file to your site-config directory as **/$deploy/site-config/sas-workload-orchestrator-server/configure/configuration.env**.

   c In the `Initial Grid Configuration` section of the copied **/sas-workload-orchestrator-server/configure/configuration.env** file, uncomment the `prometheus-push-gateway-url` property and modify the value to match what is in the file from Step 7a on page 205.

   d In the base kustomization.yaml file, delete the line in which you found the string you searched for in Step 7a on page 205. SAS recommends that you also delete the file described by that line since you have copied the information to the new configuration file.

8 If you modifed the configuration.env file in Step 6 or Step 7, continue with this step. If you did not modify the configuration.env file, skip to Step 9.

   a In the base kustomization.yaml file, add a merge behavior to merge in the settings. Here is an example:

   ```
   configMapGenerator:
   ...
   - behavior: merge
     envs:
     - /site-config/sas-workload-orchestrator-server/configure/
   configuration.env
     name: sas-workload-orchestrator-server-user-configuration
   ```

   b Also in the base kustomization.yaml file, add a reference to the **/$deploy/sas-bases/overlays/sas-workload-orchestrator-server/configure/transformers.yaml** to the transformers block. Here is an example:

   ```
   transformers:
   ...
   - sas-bases/overlays/sas-workload-orchestrator-server/configure/
   transformers.yaml
   ```

9 Determine if your current deployment has a user-defined script volume for the server pods.

   a In the base kustomization.yaml file, search for the following string:

   ```
   sas-workload-orchestrator-host-user-defined-resources-script-storage.yaml
   ```

   If the string is present, note the location of the file and continue with this step. If it is not present, you have completed this deployment note.

b Copy the **$deploy/sas-bases/examples/sas-workload-orchestrator-server/configure/sas-workload-orchestrator-server-host-user-defined-resources-script-storage.yaml** file to **/$deploy/site-config/sas-workload-orchestrator-server/configure/sas-workload-orchestrator-server-host-user-defined-resources-script-storage.yaml**.

c In the copied file, replace the {{ NFS-SERVER-ADDR }} variable with the fully-qualified domain name of the server and replace the {{ NFS-SERVER-PATH }} variable with the path to the volume on the server. You can use the file in Step 9a as a guide for the infromation to insert into the file. Here is an example:

```
- op: replace
  path: /spec/template/spec/volumes/0
  value:
    name: scripts
    nfs:
      path: /path/to/my/scripts
      server: my.nfs.server.mydomain.com
```

For more information about this configuration, see the "User-Defined Scripts Volume for Server Pods" section of the README file located at **$deploy/sas-bases/examples/sas-workload-orchestrator-server/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configuration_settings_for_sas_workload_orchestrator_servers.htm** (for HTML format).

Note: In the 2026.01 release, the HTML version of the README moved to **$deploy/sas-bases/docs/sas_workload_orchestrator_server_service_examples.htm**.

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

```
- site-config/sas-workload-orchestrator-server/configure/sas-workload-orchestrator-server-host-user-defined-resources-script-storage.yaml
```

Here is an example:

```
transformers:
...
- site-config/sas-workload-orchestrator-server/configure/sas-workload-orchestrator-serve
```

e In the base kustomization.yaml file, delete the line in which you found the string you searched for in Step 9a. SAS recommends that you also delete the file described by that line since you have copied the information to the new configuration file.

SAS/CONNECT Spawner Requires SCC for Red Hat OpenShift

Note: If your deployment is not running on Red Hat OpenShift, skip this deployment note.

All deployments on Red Hat OpenShift require binding and applying a security context constraint (SCC). For more information, including the required commands, see "Security Context Constraints and Service Accounts" in *SAS Viya Platform: Deployment Guide*.

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. 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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

2025.11 (November 2025)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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 Detection Architecture

Before Deployment Commands

New Configuration File Format

The files used to configure SAS Detection Architecture have been revised. The new format will simplify future updates.

1 In your deployment assets, search for the following subdirectory:

```
$deploy/site-config/sas-detection/
```

If the subdirectory is not present, you are not affected by the change and should skip the rest of this deployment note. If the subdirectory is present, continue with Step 2 on page 210.

2 To use the new configuration files, follow the steps described in the README file at **`$deploy/sas-bases/examples/sas-detection/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/sas_detection_engine_configuration.htm`** (for HTML format).

Note: SAS recommends referring to the **`$deploy/site-config/sas-detection/detection-engine-deployment.yaml`** file while updating the new configuration.env file. Using the detection-engine-deployment.yaml file ensures that the values used in your current deployment are used in the update.

3 After configuration.env has been revised, delete **`$deploy/site-config/sas-detection/detection-engine-deployment.yaml`** because it is no longer used in the deployment.

SAS Model Publish Service

Before Deployment Commands

BuildKit Adding Default Restrictions for Sizing

Note: If your update includes versions of the software from before 2025.10, and if the update includes performing the tasks described in "SAS Decisions Runtime Builder Service Merging with SAS Model Publish Service" on page 215 in the 2025.10 deployment notes, skip this deployment note.

SAS Model Publish is adding default restrictions to the number of CPUs, the amount of memory, and the number of replicas for BuildKit. To use the new defaults:

1 In the resoures block of your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), look for the following line:

```
- site-config/sas-model-publish/buildkit
```

If the line is not present, you are not affected by the change and should skip the rest of the deployment note. If the line is present, continue with Step 2.

2 Rename the **`$deploy/site-config/sas-model-publish/buildkit/configuration.env`** file `configuration2.env`.

3 Copy the **`$deploy/sas-bases/examples/sas-model-publish/buildkit/configuration.env`** file to **`$deploy/site-config/sas-model-publish/buildkit/configuration.env`**. The **`$deploy/site-config/sas-model-publish/buildkit/`** directory should now contain a configuration.env and a configuration2.env file.

4 Modify the configuration.env file so that it contains the values you want for your deployment.

> **IMPORTANT** You must specify the same value for `buildkitStorageSize` and `buildkitStorageClass` that was used in the previous configuration. Use the configuration.env2 file as a guide.

If you do not change the values in configuration.env, they will be the following:

Table 3.4 *Default Restrictions for BuildKit*

Minimum CPU	1 core

Minimum memory	1Gi
Maximum memory	1Gi
Maximum replicas	1 replica

Note: For more information about the configuration.env file, see the README file located at **$deploy/sas-bases/examples/sas-model-publish/buildkit/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_buildkit_for_sas_model_publish_service.htm** (for HTML format).

5 After the configuration.env file has been modified, delete the configuration2.env file.

6 Copy **$deploy/sas-bases/examples/sas-model-publish/buildkit/kustomization.yaml** file to **$deploy/site-config/sas-model-publish/buildkit/kustomization.yaml**.

SAS Studio

Before Deployment Commands

SSH Authentication with Git Requires New File

If you are leveraging SSH authentication with git, you must include a `known_hosts` file in the **<userhome>/.ssh** directory along with your public and private SSH key files. If you are uncertain whether your deployment is leveraging SSH authentication with git, go to SAS Environment Manager, select the **Configuration** page, and look in the studio service for the `allowGitSSHPassword` property. If that property is not present or if it is present but not set to `true`, then you should skip the rest of this deployment note.

You can create a `known_hosts` file by scanning your SSH keys with the following command from either a UNIX or Linux terminal or with Git Bash:

```
ssh-keyscan <remote repo server> >> ~/.ssh/known_hosts
```

Here is an example:

```
ssh-keyscan github.com >> ~/.ssh/known_hosts
```

Upload your SSH key files and `known_hosts` file to the server file system by selecting the appropriate directory in the SAS Server section of the navigation pane and clicking ⤒. Click + and browse for the SSH key and `known_hosts` files. Select

the files and click **Open**. Click **Upload** to upload the files to the directory on your SAS server.

To display hidden files and folders in the **SAS Server** and **SAS Content** sections of the navigation pane, click the user button on the application bar and select **Settings**. From the SAS Studio application settings, select **General**, and then select the **Display hidden folders and items on the file system** option.

Additionally, your environment must be configured so that each user has their own home directory. Otherwise, the SAS git functionality will not be able to find the `known_hosts` file. For information about configuring user home directories, see this blog post.

2025.10 (October 2025)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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 Decisions Runtime Builder

Before Deployment Commands

SAS Decisions Runtime Builder Service Merging with SAS Model Publish Service

The SAS Decisions Runtime Builder service is merging with the SAS Model Publish service. As a result, any references to SAS Decisions Runtime Builder in the base kustomization.yaml file (**$deploy/kustomization.yaml**) must be revised to refer to SAS Model Publish in order to deploy BuildKit with SAS Model Publish. Furthermore, BuildKit deployments pull base layers from source registries and publish container images to target registries. If the source registry that contains SAS Viya platform deployment images or the destination registry uses self-signed certificates, then the certificates should be added to the BuildKit deployment via the certificate framework.

Note: Steps 4 through 8 duplicate steps that are included in the "Incorporating Additional CA Certificates" section of README file located 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). For more information, see that README.

1 In your deployment assets, search for the following directory:

```
$deploy/site-config/sas-decisions-runtime-builder
```

If the directory is present, continue with Step 2. If it is not present, skip the rest of this deployment note.

2 Configure BuildKit for use with the SAS Model Publish service by following the instructions in the README file located at **$deploy/sas-bases/examples/sas-model-publish/buildkit/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_buildkit_for_sas_model_publish_service.htm** (for HTML format).

Note: SAS recommends referring to the **$deploy/site-config/sas-decisions-runtime-builder/buildkit/storage.yaml** file while updating the new configuration.env file to ensure that the values used in your current deployment are used in the update.

3 In the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the following line:

```
- site-config/sas-decisions-runtime-builder/buildkit/certs
```

If the line is present, continue with Step 4. If it is not present, skip to Step 9.

4 Look in your deployment assets for the following directory:

```
$deploy/site-config/security/cacerts
```

If the directory does not exist, create it.

5 Copy the contents of **$deploy/site-config/sas-decisions-runtime-builder/buildkit/certs** to **$deploy/site-config/security/cacerts**.

6 Copy the **$deploy/sas-bases/examples/security/customer-provided-ca-certificates.yaml** file to **$deploy/site-config/security/**

7 Edit the **$deploy/site-config/security/customer-provided-ca-certificates.yaml** to include the names of the CA certificates.

8 Edit the base kustomization.yaml file to add the following line in the generators block:

```
- site-config/security/customer-provided-ca-certificates.yaml
```

9 Also in the base kustomization.yaml file, delete the following lines:

- In the resources block, delete

```
- site-config/sas-decisions-runtime-builder/buildkit/certs
```

```
- site-config/sas-decisions-runtime-builder/buildkit
```

- In the transformers block, delete

```
- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/
buildkit-certificates-transformer.yaml
```

```
- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/buildkit-transformer.yaml
```

10 Delete the **$deploy/site-config/sas-decisions-runtime-builder** directory from your deployment assets.

SAS Risk Cirrus KRM Service

Before Deployment Commands

New Configuration File Format

The SAS Risk Cirrus KRM service provides a REST API for starting and managing KRM runs associated with Risk Cirrus analysis runs and comes with default settings that may be changed. The files used to configure SAS Risk Cirrus KRM Service have been revised. The new format will simplify future updates.

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

```
- site-config/sas-risk-cirrus-krm/resources/krm_transform.yaml
```

If the line is not present, you are not affected by the change and should skip the rest of the deployment note. If the line is present, continue with Step 2.

2 To use the new configuration files, follow the steps described in the README file at **$deploy/sas-bases/examples/sas-risk-cirrus-krm/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_the_sas_risk_cirrus_krm_service.htm** (for HTML format).

Note: SAS recommends referring to the **$deploy/site-config/sas-risk-cirrus-krm/resources/krm_transform.yaml** file while updating the new configuration.env file to ensure that the values used in your current deployment are used in the update.

3 The new configuration of the files means that the properties and values used in the old krm_transform.yaml file are now in the configuration.env file. Because the file is no longer needed, delete **$deploy/site-config/site-config/sas-risk-cirrus-krm/resources/krm_transform.yaml** from your deployment assets. Also, remove the reference to it from the transformers block of the base kustomization.yaml file.

2025.09 (September 2025)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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 Decisions Runtime Builder

Before Deployment Commands

Removal of Support for Kaniko

Support for Kaniko has been removed from SAS Decisions Runtime Builder, finishing the process begun in the 2025.06 release (see "BuildKit Replaces Kaniko as Default for Container Images" on page 240 for more information). If you continued to use Kaniko with the SAS Decisions Runtime Builder service after the 2025.06 update, or if you were directed to skip the deployment note at 2025.06 because you are updating through several releases at once, references to Kaniko must be removed from the base kustomization.yaml file.

- Search for `- site-config/sas-decisions-runtime-builder/kaniko` in the resources block. If it is present, remove it.
- Search for `- sas-bases/overlays/sas-decisions-runtime-builder/kaniko/kaniko-transformer.yaml` in the transformers block. If it is present, remove it.
- Search for `- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/buildkit-kaniko-transformer.yaml` in the transformers block. If it is present, replace it with `- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/buildkit-transformer.yaml`.

IMPORTANT If you are performing an update that includes the 2025.10 release, do not forget to configure BuildKit for the SAS Model Publish service using the tasks in the 2025.10 deployment note, "SAS Decisions Runtime Builder Service Merging with SAS Model Publish Service" on page 215. Especially pay attention to Step 2.

SAS Model Publish Service

Before Deployment Commands

New Configuration File Format

The configuration files for controlling the size of the persistent volume claim (PVC) used by the SAS Model Publish service have been revised. The new format will simplify future updates.

1 In the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the following line:

```
- site-config/sas-model-publish/git
```

If the line is not present, you are not affected by the change and should skip the rest of the deployment note. If the line is present, continue with Step 2.

2 To use the new configuration files, follow the steps described in the README file at **$deploy/sas-bases/examples/sas-model-publish/git/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_git_for_sas_model_publish_service.htm** (for HTML format).

Note: SAS recommends referring to the **$deploy/site-config/sas-model-publish/git/storage.yaml** file while updating the new configuration.env file to ensure that the values used in your current deployment are used in the update.

3 The new configuration of the files means that the properties and values used in the old storage.yaml file are now in the configuration.env file. Because the file is no longer needed, delete **$deploy/site-config/sas-model-publish/git/storage.yaml** from your deployment assets.

Remove Kaniko References

If you continued to use Kaniko with the SAS Model Publish service after the 2025.06 update, remove references to it from the base kustomization.yaml file. Take the following steps:

1 Search for `- site-config/sas-model-publish/kaniko` in the resources block. If it is present, remove it.

2 Search for `- sas-bases/overlays/sas-model-publish/kaniko/kaniko-transformer.yaml` in the transformers block. If it is present, remove it.

3 Delete the **$deploy/site-config/sas-model-publish/kaniko** directory from your deployment assets.

> **IMPORTANT** If you are using Red Hat OpenShift, ensure that you have bound and applied the security context constraint for sas-model-publish. For more information, see "sas-model-publish" in *SAS Viya Platform: Deployment Guide*.

2025.08 (August 2025)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Revised SAS Configurator for Open Source Transformer

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised to support a newer version of Python and is 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. 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.
5. After your revisions are complete, delete the change-configuration2.yaml file.

For more information about the change-configuration.yaml file, see "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

JAVAOBJ Added to LOCKDOWN Default Blocked List

The JAVAOBJ access method has been added to the list of access methods that LOCKDOWN blocks by default. If your deployment uses the JAVAOBJ access method, after the update your access will be blocked unless you add JAVAOBJ to the enabled list before the update.

To enable the JAVAOBJ access method, follow the instructions in the README file located at **$deploy/sas-bases/examples/sas-programming-environment/lockdown/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/lockdown_settings_for_the_sas_programming_environment.htm** (for HTML format). Ensure that you modify the enable-lockdown-access-methods.yaml file and also that you add a reference to that file in the transformers block of the base kustomization.yaml file (**$deploy/kustomization.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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

Model Studio

Before Deployment Commands

JAVAOBJ Added to LOCKDOWN Default Blocked List

To run the Open Source Code node in Model Studio, the JAVAOBJ access method must be enabled. However, the JAVAOBJ access method has been added to the list of access methods that LOCKDOWN blocks by default. If your deployment uses the JAVAOBJ access method, after the update your access will be blocked unless you add JAVAOBJ to the enabled list before the update.

To enable the JAVAOBJ access method, follow the instructions in the README file located at **$deploy/sas-bases/examples/sas-programming-environment/lockdown/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/lockdown_settings_for_the_sas_programming_environment.htm** (for HTML

format). Ensure that you modify the enable-lockdown-access-methods.yaml file and also that you add a reference to that file in the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**).

SAS Data Quality

Before Deployment Commands

New Configuration File Format

The configuration files for controlling the size of the persistent volume claim (PVC) used to store the Quality Knowledge Base have been completely revised. The new format will simplify future updates.

1 In the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**), search for the following line:

```
- site-config/data-quality/storageclass/storage-class-transformer.yaml
```

If the line is not present, you are not affected by the change and should skip the rest of the deployment note. If the line is present, continue with Step 2.

2 Delete the `- site-config/data-quality/storageclass/storage-class-transformer.yaml` line from the base kustomization.yaml.

3 To use the new configuration files, follow the steps described in the README file at **$deploy/sas-bases/examples/data-quality/storagesize/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/quality_knowledge_base_for_the_sas_viya_platform.htm** (for HTML format).

4 Delete the actual `$deploy/site-config/data-quality/storageclass/storage-class-transformer.yaml` file from the deployment assets.

SAS Decisions Runtime Builder

Before Deployment Commands

BuildKit Replaces Kaniko as Default for Container Images

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

Note: This deployment note is repeated from version 2025.06 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 update, BuildKit is used by default instead of Kaniko to publish models or decisions to container destinations. Kaniko is not removed from your deployment, but BuildKit is used, even if Kaniko is deployed.

1 Look at the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following entry:

```
- site-config/sas-decisions-runtime-builder/kaniko
```

If the line is not present, then you do not have the SAS Decisions Runtime Builder service with Kaniko deployed, and you should skip the rest of this deployment note. If the line is present, continue with Step 2.

2 In the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**), check for the following entry:

```
- site-config/sas-decisions-runtime-builder/buildkit
```

If the line is present, BuildKit has been installed and configured. You should skip to Step 4. Otherwise, continue with Step 3.

3 To install and configure BuildKit, perform the steps described in the README file located at **$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_buildkit_for_sas_decisions_runtime_builder_service.htm** (for HTML format).

4 If you want to continue to use Kaniko, perform the following tasks:

IMPORTANT Support for the SAS Container Runtime Kaniko feature flag ends with the 2025.09 release. SAS recommends configuring BuildKit

for use now. If you are updating to 2025.09, you can skip the remainder of this note. When you reach the *Deployment Notes* for 2025.09, complete the steps in "SAS Decisions Runtime Builder" on page 219 instead.

a See the relevant deployment note, "Continue to Use Kaniko for Container Images" on page 226. That note provides context for the steps that follow.

b In the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**), search for `- sas-bases/overlays/sas-decisions-runtime-builder/kaniko/kaniko-transformer.yaml`.

   If this transformer is present, you have both Kaniko and BuildKit configured in your deployment. You must replace the following reference:

   ```
   - sas-bases/overlays/sas-decisions-runtime-builder/buildkit/buildkit-transformer.yaml
   ```

   with this reference:

   ```
   - sas-bases/overlays/sas-decisions-runtime-builder/buildkit/buildkit-kaniko-transformer.yaml
   ```

   If the search does not return the reference to the kaniko-transformer.yaml, no replacement is required.

5 If you choose to stop using Kaniko now and want to remove it from your deployment, delete the following lines from your base kustomization.yaml file (**$deploy/kustomization.yaml**):

```
resources:
- site-config/sas-decisions-runtime-builder/kaniko
- site-config/sas-model-publish/kaniko

transformers:
- sas-bases/overlays/sas-decisions-runtime-builder/kaniko/kaniko-transformer.yaml
- sas-bases/overlays/sas-model-publish/kaniko/kaniko-transformer.yaml
```

After Deployment Commands

Continue to Use Kaniko for Container Images

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

Note: This deployment note is based on the previous note, "BuildKit Replaces Kaniko as Default for Container Images" on page 225. For the background about this change, see that note.

After the update, BuildKit is used by default for container images. To continue to use Kaniko, either by itself or alongside BuildKit, enable the SAS Container Runtime

Kaniko feature flag using SAS Environment Manager. For more information, see "Configure Container Publishing Destinations" in *SAS Viya Platform: Publishing Destinations*.

Note: The SAS Container Runtime Kaniko feature flag will be removed with the 2025.09 release, and Kaniko will no longer be available.

SAS SpeedyStore

Before Deployment Commands

Updated Configuration Transformer File

Note: The tasks described in this deployment note only apply if your deployment is running on AWS. If your deployment is running on a different provider, you should skip this deployment note.

The update contains a revised transformer file, sas-singlestore-cluster-config.yaml. If you used the sas-singlestore-cluster-config.yaml file in your deployment, you must replace the existing version of the file with the new version.

1. In the transformers block of your base kustomization.yaml file (**`$deploy/kustomization.yaml`**), search for the string "site-config/sas-singlestore/sas-singlestore-cluster-config.yaml". If that string is not present, then no further steps are necessary. If the string is present, continue with step 2.
2. Go to **`$deploy/site-config/sas-singlestore/`** and change the file name to `sas-singlestore-cluster-config2.yaml`.
3. Copy **`$deploy/sas-bases/examples/sas-singlestore/sas-singlestore-cluster-config.yaml`** to **`$deploy/site-config/sas-singlestore/`**. You should now have a sas-singlestore-cluster-config.yaml file and a sas-singlestore-cluster-config2.yaml file in the same directory.
4. Using the sas-singlestore-cluster-config2.yaml file as a guide, revise the sas-singlestore-cluster-config.yaml file so that it includes the same values you used in the earlier version of the file. Notice that the revisions to the sas-singlestore-cluster-config.yaml can include user-supplied values for variables as well as commenting out and uncommenting lines within the file.

   Note: For information about using the sas-singlestore-cluster-config.yaml file, see the README file located at **`$deploy/sas-bases/examples/sas-`**

`singlestore/README.md` (for Markdown format) or at `$deploy/sas-bases/docs/sas_singlestore_cluster_operator.htm` (for HTML format).

5 After your revisions are complete, delete the sas-singlestore-cluster-config2.yaml file.

2025.07 (July 2025)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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 Decisions Runtime Builder

Before Deployment Commands

BuildKit Replaces Kaniko as Default for Container Images

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

Note: This deployment note is repeated from version 2025.06 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 update, BuildKit is used by default instead of Kaniko to publish models or decisions to container destinations. Kaniko is not removed from your deployment, but BuildKit is used, even if Kaniko is deployed.

1 Look at the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following entry:

```
- site-config/sas-decisions-runtime-builder/kaniko
```

If the line is not present, then you do not have the SAS Decisions Runtime Builder service with Kaniko deployed, and you should skip the rest of this deployment note. If the line is present, continue with Step 2.

2 Look at the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following entry:

```
- site-config/sas-decisions-runtime-builder/buildkit
```

If the line is present, then BuildKit has been installed and configured. You should skip to Step 4. Otherwise, continue with Step 3.

3 To install and configure BuildKit, perform the steps described in the README file located at **$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_buildkit_for_sas_decisions_runtime_builder_service.htm** (for HTML format).

4 If you want to continue to use Kaniko, perform the following tasks:

> **IMPORTANT** Support for the SAS Container Runtime Kaniko feature flag ends with the 2025.09 release. SAS recommends configuring BuildKit for use now. If you are updating to 2025.09, you can skip the remainder of this note. When you reach the *Deployment Notes* for 2025.09, complete the steps in "SAS Decisions Runtime Builder" on page 219 instead.

a See the relevant deployment note, "Continue to Use Kaniko for Container Images" on page 232. That note provides context for the steps that follow.

b In the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**), search for `- sas-bases/overlays/sas-decisions-runtime-builder/kaniko/kaniko-transformer.yaml`.

If this transformer is present, you have both Kaniko and BuildKit configured in your deployment. You must replace the following reference:

```
- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/
buildkit-transformer.yaml
```

with this reference:

```
- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/
buildkit-kaniko-transformer.yaml
```

If the search does not return the reference to the kaniko-transformer.yaml, no replacement is required.

5 If you choose to stop using Kaniko now and want to remove it from your deployment, delete the following lines from your base kustomization.yaml file (**$deploy/kustomization.yaml**):

```
resources:
```

```
- site-config/sas-decisions-runtime-builder/kaniko
- site-config/sas-model-publish/kaniko

transformers:
- sas-bases/overlays/sas-decisions-runtime-builder/kaniko/kaniko-transformer.yaml
- sas-bases/overlays/sas-model-publish/kaniko/kaniko-transformer.yaml
```

Revised Configuration File

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

Note: If you are performing an update that includes installing and configuring BuildKit from Step 3 of the tasks associated with "BuildKit Replaces Kaniko as Default for Container Images" on page 229, skip this note.

The publish-job-template.yaml file has been revised and is replaced during update. Ensure that its changes are included in your deployment by performing the following steps.

1. Go to the **$deploy/site-config/sas-decisions-runtime-builder/buildkit** directory.
2. (Red Hat OpenShift only) In that directory, change the publish-job-template.yaml file name to publish-job-template2.yaml.
3. Copy the **$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/publish-job-template.yaml** file to **$deploy/site-config/sas-decisions-runtime-builder/buildkit**. If your deployment is not on Red Hat OpenShift, the new publish-job-template.yaml file is in place and you are finished with this deployment note. If your deployment is on Red Hat OpenShift, you now have a publish-job-template.yaml file and a publish-job-template2.yaml file in the same directory.
4. (Red Hat OpenShift only) In the publish-job-template.yaml file, revise the `{{ FSGROUP_VALUE }}` to match the value used in the publish-job-template2.yaml file.
5. (Red Hat OpenShift only) Delete the publish-job-template2.yaml file.

For more information about the publish-job-template.yaml file, see the README file located at **$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_buildkit_for_sas_decisions_runtime_builder_service.htm** (for HTML format).

Revised Time-Sensitive Configuration Files

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

Note: This deployment note is repeated from version 2025.06 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 or if you have performed the steps described in "BuildKit Replaces Kaniko as Default for Container Images" on page 240, do not perform the tasks described in this deployment note.

If your deployment includes SAS Decisions Runtime Builder running with BuildKit, your configuration contains two files that are time-sensitive. The update contains revised versions of those files to keep your deployment running. If the revised files are not placed correctly, your deployment will cease working because the configuration files being used by your deployment will expire.

1 In your base kustomization.yaml file, (**`$deploy/kustomization.yaml`**), search the resources block for the following line:

```
- site-config/sas-decisions-runtime-builder/buildkit
```

If the line is not present, your deployment is unaffected by the change and you should skip this deployment note and the related one that follows.

If the line is present, continue with Step 2.

2 Copy the following files to **`$deploy/site-config/sas-decisions-runtime-builder/buildkit`**:

- **`$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/buildkit-client-certs.yaml`**
- **`$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/buildkit-daemon-certs.yaml`**

Copying the files should overwrite files in the target location with the same name. This is the desired behavior.

3 After updating your deployment with the appropriate deployment commands, be sure to perform the task described in "Restart the Daemon Buildkit Pod" on page 233.

After Deployment Commands

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

Continue to Use Kaniko for Container Images

Note: This deployment note is based on the previous note, "BuildKit Replaces Kaniko as Default for Container Images" on page 229. For the background about this change, see that note.

After the update, BuildKit is used by default for container images. To continue to use Kaniko, either by itself or alongside BuildKit, enable the SAS Container Runtime Kaniko feature flag using SAS Environment Manager. For more information, see "Configure Container Publishing Destinations" in *SAS Viya Platform: Publishing Destinations*.

Note: The SAS Container Runtime Kaniko feature flag will be removed with the 2025.09 release, and Kaniko will no longer be available.

Restart the Daemon Buildkit Pod

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

Note: If you did not perform the tasks described in "Revised Time-Sensitive Configuration Files" on page 231, skip this deployment note.

Restart your deployment's buildkit daemon pod with the following command:

```
kubectl -n name-of-namespace rollout restart deployments/buildkitd
```

SAS Micro Analytic Service

Before Deployment Commands

Revised ASTORES Transformer File

Note: This deployment note is repeated from version 2025.06 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.

SAS Micro Analytic Service is a component used in a number of offerings in the SAS Viya platform, including SAS Event Stream Processing Studio, SAS Intelligent Decisioning, and SAS Model Manager. This update contains a revised transformer for SAS Micro Analytic Service, astores-transformer.yaml. For some deployments, the revised astores-transformer.yaml file might be in a different location than expected. Use the following steps to determine if your deployment is affected and to ensure that you have the latest astores-transformer.yaml file in the correct location for the update.

1 Search the transformers block of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for a reference to astores-transformer.yaml.

2 Based on the result of your search, perform one of the following tasks:

- If the full reference is `- sas-bases/overlays/sas-microanalytic-score/astores/astores-transformer.yaml`, the reference is correct. Do not take any further steps.
- If the full reference is `- site-config/sas-microanalytic-score/astores/astores-transformer.yaml`, revise the line to read as follows: `- sas-bases/overlays/sas-microanalytic-score/astores/astores-transformer.yaml`. Then save and close the base kustomization.yaml file.

  Note: With this revision, the **`$deploy/site-config/sas-microanalytic-score/astores/astores-transformer.yaml`** file will no longer be used. It can be safely deleted from your deployment assets.

2025.06 (June 2025)

Additional Documentation

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Updated CAS Host Launch Configuration Files

A property in the file to set up and configure CAS Host Launch and two security content constraints (SCCs) have been revised. If your deployment uses CAS Host Launch, you must replace the original configuration file with the new one before updating your software. Additionally, if you are using Red Hat OpenShift as your cloud provider and depending on the cas-server SCC you used in your deployment, you might also have to apply and bind the revised SCC.

Note: For details on CAS Host Launch, see "Enable Host Launch" in *SAS Viya Platform: Deployment Guide*.

1. To determine if your deployment uses CAS Host Launch, look at the base kustomization file (**`$deploy/kustomization.yaml`**). In the transformers block, search for the string "cas-enable-host". If the string is present, continue with the steps in this deployment note. If the string is not present, your deployment does not have CAS Host Launch enabled, and you should skip the rest of this deployment note.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `cas-enable-host2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/cas/configure/cas-enable-host.yaml`** file to the location of your cas-enable-host2.yaml file. You should now have a cas-enable-host.yaml file and a cas-enable-host2.yaml file in the same directory
4. Compare the cas-enable-host.yaml and cas-enable-host2.yaml files. Use the cas-enable-host2.yaml file as a guide for the revisions to be made to the cas-enable-host.yaml file in order to match the configuration settings used in your existing deployment.
5. After your revisions are complete, delete the cas-enable-host2.yaml file.

   Note: If your deployment runs on Red Hat OpenShift, continue with the steps in this deployment note. Otherwise, you have completed the required steps, and you should move on to the next deployment note.

6. Determine the SCC that you used in the deployment with the following command:

```
kubectl get scc | grep "cas-server"
```

7 Your next steps depend on the output of the command in Step 6.

- If `sas-cas-server-host` is in the output of the command, apply the revised SCC and bind it to the service account with the following commands:

  ```
  kubectl apply -f sas-bases/examples/cas/configure/cas-server-scc-host-launch.yaml
  ```

  ```
  oc -n name-of-namespace adm policy add-scc-to-user sas-cas-server-host -z sas-cas-server
  ```

- If `sas-cas-server-sssd` is in the output of the command, apply the revised SCC and bind it to the service account with the following commands:

  ```
  kubectl apply -f sas-bases/examples/cas/configure/cas-server-scc-sssd.yaml
  ```

  ```
  oc -n name-of-namespace adm policy add-scc-to-user sas-cas-server-sssd -z sas-cas-server
  ```

- If neither of those SCCs is listed, skip to Step 9.

8 To ensure that the SCC used for backups is synchronized with the CAS server host SCC, perform the following commands:

```
kubectl apply -f sas-backup-pv-copy-cleanup-job-scc.yaml
```

```
oc create -f sas-backup-pv-copy-cleanup-job-scc.yaml
```

9 In the cluster, look for any namespaces that are using a release earlier than 2025.06. Perform the following commands for each namespace that is older than the 2025.06 release:

- Depending on the CAS server SCC that you use in the namespace, perform one of the following commands.

  ```
  kubectl patch scc sas-cas-server-host --type json -p '[{"op": "add", "path": "/allowedCapabilities/-", "value": "DAC_OVERRIDE"}]'
  ```

  ```
  kubectl patch scc sas-cas-server-sssd --type json -p '[{"op": "add", "path": "/allowedCapabilities/-", "value": "DAC_OVERRIDE"}]'
  ```

- Perform the following command to revise the backup SCC:

  ```
  kubectl patch scc sas-backup-pv-copy-cleanup-job --type json -p '[{"op": "add", "path": "/allowedCapabilities/-", "value": "DAC_OVERRIDE"}]'
  ```

Updated Storage Class Transformer

Note: This deployment note is repeated from version 2025.01 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 to target only sas-programming-environment podTemplates that make use of the viya volume. 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 the transformers block of 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:

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      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
```

Note: For more information about using the change-viya-volume-storage-class.yaml file, see the README file located 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).

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

After Deployment Commands

Multiple OIDC Configurations with Same Issuer Property

Note: This deployment note is repeated from version 2025.03 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.

OIDC support has been revised and no longer allow multiple configurations with duplicate issuer values. You should perform the following steps to determine if you have configurations with duplicate values. If you have duplicate values, the procedure also provides information for addressing the issue.

1 To list all the configuration properties under config/SASLogon that contain "oidc":

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse config/SASLogon/" | grep oauth
```

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

2 Evaluate the output of the command in Step 1 to determine if it contains multiple issuer properties. If there is not more than one issuer property in the output or if no issuer properties are duplicated, you are not affected by the concern and you can skip the rest of this note. If there are duplicate issuer properties in the output, continue with the next step.

3 Perform the following command:

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse configurationservice/configurations/" |
grep sas.logon.oauth.providers
```

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

4 Create a case at the Customer Service Portal in order to receive assistance from SAS Technical Support for the rest of the procedure. In the case that you create, be sure that you include the output of the command in Step 3.

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*.

This version contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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 Decisions Runtime Builder

Before Deployment Commands

BuildKit Replaces Kaniko as Default for Container Images

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

With the update, BuildKit is used by default instead of Kaniko to publish models or decisions to container destinations. Kaniko is not removed from your deployment, but BuildKit is used, even if Kaniko is deployed.

1 Look at the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following entry:

```
- site-config/sas-decisions-runtime-builder/kaniko
```

If the line is not present, then you do not have the SAS Decisions Runtime Builder service with Kaniko deployed, and you should skip the rest of this deployment note. If the line is present, continue with Step 2.

2 Look at the resources block of the base kustomization.yaml file (**$deploy/kustomization.yaml**) again for the following entry:

```
- site-config/sas-decisions-runtime-builder/buildkit
```

If the line is present, then BuildKit has been installed and configured. You should skip to Step 4. Otherwise, continue with Step 3.

3 To install and configure BuildKit, perform the steps described in the README file located at **$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_buildkit_for_sas_decisions_runtime_builder_service.htm** (for HTML format).

4 If you want to continue to use Kaniko, perform the following tasks:

> **IMPORTANT** Support for the SAS Container Runtime Kaniko feature flag ends with the 2025.09 release. SAS recommends configuring BuildKit for use now. If you are updating to 2025.09, you can skip the remainder of this note. When you reach the *Deployment Notes* for 2025.09, complete the steps in "SAS Decisions Runtime Builder" on page 219 instead.

a See the relevant deployment note, "Continue to Use Kaniko for Container Images" on page 242. That note provides context for the steps that follow.

b In the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**), search for `- sas-bases/overlays/sas-decisions-runtime-builder/kaniko/kaniko-transformer.yaml`.

If this transformer is present, you have both Kaniko and BuildKit configured in your deployment. You must replace the following reference:

```
- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/buildkit-transformer.yaml
```

with this reference:

```
- sas-bases/overlays/sas-decisions-runtime-builder/buildkit/buildkit-kaniko-transformer.yaml.
```

If the search does not return the reference to the kaniko-transformer.yaml, no replacement is required.

5 If you choose to stop using Kaniko now and want to remove it from your deployment, delete the following lines from your base kustomization.yaml file (**$deploy/kustomization.yaml**):

```
resources:
- site-config/sas-decisions-runtime-builder/kaniko
- site-config/sas-model-publish/kaniko

transformers:
- sas-bases/overlays/sas-decisions-runtime-builder/kaniko/kaniko-transformer.yaml
- sas-bases/overlays/sas-model-publish/kaniko/kaniko-transformer.yaml
```

Revised Time-Sensitive Configuration Files

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

Note: If you installed and configured BuildKit in Step 3 of the tasks associated with "BuildKit Replaces Kaniko as Default for Container Images" on page 240, you should skip this deployment note. The steps in that deployment note account for the required tasks in this deployment note.

If your deployment includes SAS Decisions Runtime Builder running with BuildKit, your configuration contains two files that are time-sensitive. The update contains revised versions of those files to keep your deployment running. If the revised files are not placed correctly, your deployment will cease working because the configuration files being used by your deployment will expire.

1 In your base kustomization.yaml file, (**$deploy/kustomization.yaml**), search the resources block for the following line:

```
- site-config/sas-decisions-runtime-builder/buildkit
```

If the line is not present, then your deployment is unaffected by the change and you should skip this deployment note and the related one that this note links to. If the line is present, continue with Step 2.

2 Copy the following files to **`$deploy/site-config/sas-decisions-runtime-builder/buildkit`**:

- **`$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/buildkit-client-certs.yaml`**
- **`$deploy/sas-bases/examples/sas-decisions-runtime-builder/buildkit/buildkit-daemon-certs.yaml`**

Copying the files should overwrite files in the target location with the same name. This is the desired behavior.

3 After updating your deployment with the appropriate deployment commands, be sure to perform the task described in "Restart the Daemon Buildkit Pod" on page 242.

After Deployment Commands

Continue to Use Kaniko for Container Images

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

Note: This deployment note is based on the previous note, "BuildKit Replaces Kaniko as Default for Container Images" on page 240. For the background about this change, see that note.

After the update, BuildKit is used by default for container images. To continue to use Kaniko, either by itself or alongside BuildKit, enable the SAS Container Runtime Kaniko feature flag using SAS Environment Manager. For more information, see "Configure Container Publishing Destinations" in *SAS Viya Platform: Publishing Destinations*.

Note: The SAS Container Runtime Kaniko feature flag will be removed with the 2025.09 release, and Kaniko will no longer be available.

Restart the Daemon Buildkit Pod

Note: If you are performing an update that includes the 2025.10 release, do not perform the steps in this deployment note.

Note: If you did not copy and move the files described in "Revised Time-Sensitive Configuration Files" on page 241, skip this deployment note.

Restart your deployment's buildkit daemon pod with the following command:
`kubectl -n name-of-namespace rollout restart deployments/buildkitd`

SAS Micro Analytic Service

Before Deployment Commands

Revised ASTORES Transformer File

SAS Micro Analytic Service is a component used in a number of offerings in the SAS Viya platform, including SAS Event Stream Processing Studio, SAS Intelligent Decisioning, and SAS Model Manager. This update contains a revised transformer for SAS Micro Analytic Service, astores-transformer.yaml. For some deployments, the revised astores-transformer.yaml file might be in a different location than expected. Use the following steps to determine if your deployment is affected and to ensure that you have the latest astores-transformer.yaml file in the correct location for the update.

1 Search the transformers block of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for a reference to astores-transformer.yaml.

2 Based on the result of your search, perform one of the following tasks:

- If the full reference is `- sas-bases/overlays/sas-microanalytic-score/astores/astores-transformer.yaml`, the reference is correct. Do not take any further steps.
- If the full reference is `- site-config/sas-microanalytic-score/astores/astores-transformer.yaml`, revise the line to read as follows: `- sas-bases/overlays/sas-microanalytic-score/astores/astores-transformer.yaml`. Then save and close the base kustomization.yaml file.

  Note: With this revision, the **`$deploy/site-config/sas-microanalytic-score/astores/astores-transformer.yaml`** file will no longer be used. It can be safely deleted from your deployment assets.

2025.05 (May 2025)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Updated Storage Class Transformer

Note: This deployment note is repeated from version 2025.01 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 to target only sas-programming-environment podTemplates that make use of the viya volume. 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 the transformers block of 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:

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      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
```

Note: For more information about using the change-viya-volume-storage-class.yaml file, see the README file located 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).

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

After Deployment Commands

Multiple OIDC Configurations with Same Issuer Property

Note: This deployment note is repeated from version 2025.03 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.

OIDC support has been revised and no longer allow multiple configurations with duplicate issuer values. You should perform the following steps to determine if you have configurations with duplicate values. If you have duplicate values, the procedure also provides information for addressing the issue.

1 To list all the configuration properties under config/SASLogon that contain "oidc":

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse config/SASLogon/" | grep oauth
```

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

2 Evaluate the output of the command in Step 1 to determine if it contains multiple issuer properties. If there is not more than one issuer property in the output or if no issuer properties are duplicated, you are not affected by the concern and you can skip the rest of this note. If there are duplicate issuer properties in the output, continue with the next step.

3 Perform the following command:

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse configurationservice/configurations/" |
grep sas.logon.oauth.providers
```

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

4 Create a case at the Customer Service Portal in order to receive assistance from SAS Technical Support for the rest of the procedure. In the case that you create, be sure that you include the output of the command in Step 3.

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 2025.05 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

2025.04 (April 2025)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Changes for Launcher Service Transformers

The transformers used to configure the user and super-user process limits for the launcher service have been revised. If you have used these transformers in your deployment, you must replace the original transformers with the new ones. You must also ensure that any revisions that you made to the original transformers have been duplicated to the new ones.

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

- launcher-user-process-limit.yaml

- launcher-super-user-process-limit.yaml

If either or both of the transformers are listed in the base kustomization.yaml file, continue with this deployment note. If neither is listed in the base kustomization.yaml file, your deployment is unaffected by this change, and you should skip the rest of this deployment note.

2 The full entry for the transformer or transformers is the location where those files can be found. Go to that location and rename the affected transformer file or files with a "2". For example, if you have **$deploy/site-config/sas-launcher/configure/launcher-user-process-limit.yaml**, rename that file launcher-user-process-limit2.yaml.

3 Copy the new version of the file or files from **$deploy/sas-bases/examples/sas-launcher/configure/** so that they are in the same directory location as the original file or files. Using the example from Step 2, you would have two files at **$deploy/site-config/sas-launcher/configure/**, launcher-user-process-limit.yaml and launcher-user-process-limit2.yaml.

4 Using the original version of the file or files as a guide, edit the new versions so that the original decisions you made are duplicated in the new files.

5 Delete the original version of the transformer files, thereby removing files that have a "2" in their name.

For more information about configuring the launcher service, see the README file located at **$deploy/sas-bases/examples/sas-launcher/configure/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configuration_settings_for_sas_launcher_service.htm** (for HTML format).

Updated Storage Class Transformer

Note: This deployment note is repeated from version 2025.01 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 to target only sas-programming-environment podTemplates that make use of the viya volume. 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 the transformers block of 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:

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      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
```

Note: For more information about using the change-viya-volume-storage-class.yaml file, see the README file located 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).

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

Update Configuration for SAS Servers Using Kerberos on OpenShift

Note: This deployment note is repeated from version 2025.03 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.

Note: If your deployment is not running on Red Hat OpenShift, skip this deployment note.

Because of differences in how Red Hat OpenShift handles Kerberos, SAS is introducing new transformers specifically for deployments on Red Hat OpenShift. The new transformers take the place of transformers in use prior to the update.

1 Search the transformers block of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following entry:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-tls.yaml
```

If you find the entry, replace it with the following line:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-tls-openshift.yaml
```

2 Search the transformers block of the base kustomization.yaml file for the following entry:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-no-tls.yaml
```

If you find the entry, replace it with the following line:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-no-tls-openshift.yaml
```

3 If neither entry is in your base kustomization.yaml file, no further action is required.

After Deployment Commands

Multiple Configuration Definitions for SAML Identity Providers

Note: This deployment note is repeated from version 2025.02 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.

SAML libraries have been revised and no longer allow multiple configurations with duplicate entityID values. You should perform the following steps to determine if you have configurations with duplicate values. If you have duplicate values, the procedure also provides information for addressing the issue.

1 To list all the configuration properties under config/SASLogon that contain "saml":

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-bootstrap-config kv read --recurse config/SASLogon/" | grep saml
```

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

2 Evaluate the output of the command in Step 1 to determine if it contains multiple idpMetadata properties. If the output of the command is a single item, you are not affected by the issue and you can skip the rest of this note. If there is more than one item in the output, continue with the next step.

3 Review the contents for each idpMetadata property. Determine whether the entityID value is unique for each or if there are identical entityID values. If you do not have identical entityID values, you are not affected by the issue and you can skip the rest of this note. If you have identical entityID values, continue with the next steps.

Note: You might see XML returned for the idpMetadata property. You can view the XMl directly in the command output to check the entityID values. You might also see a URI returned for the idpMetadata property. If a URI is returned for the idpMetadata property, you must enter that URI in a browser to view the XML and check the entityID values.

4 Perform the following command:

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse configurationservice/configurations/" |
grep sas.logon.saml.providers
```

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

5 Create a case at the Customer Service Portal in order to receive assistance from SAS Technical Support for the rest of the procedure. In the case that you create, be sure that you include the output of the command in Step 4.

Multiple OIDC Configurations with Same Issuer Property

Note: This deployment note is repeated from version 2025.03 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.

OIDC support has been revised and no longer allow multiple configurations with duplicate issuer values. You should perform the following steps to determine if you have configurations with duplicate values. If you have duplicate values, the procedure also provides information for addressing the issue.

1 To list all the configuration properties under config/SASLogon that contain "oidc":

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse config/SASLogon/" | grep oauth
```

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

2 Evaluate the output of the command in Step 1 to determine if it contains multiple issuer properties. If there is not more than one issuer property in the output or if no issuer properties are duplicated, you are not affected by the concern and you can skip the rest of this note. If there are duplicate issuer properties in the output, continue with the next step.

3 Perform the following command:

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse configurationservice/configurations/" |
grep sas.logon.oauth.providers
```

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

4 Create a case at the Customer Service Portal in order to receive assistance from SAS Technical Support for the rest of the procedure. In the case that you create, be sure that you include the output of the command in Step 3.

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 2025.04 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

Guest Access

If you have enabled guest access for users of SAS Job Execution Web Application, a manual step is required. An authorization rule has changed, and it potentially affects guest access.

Users who are logged in to the Guest account will see errors from the file service when they attempt to run jobs. The errors state, "`Initial Files Service creation of file failed`." To avoid these errors, modify an authorization rule after you have deployed the updates. Use the SAS Viya platform command-line interface to run the following commands:

Note: The authorization plug-in to the SAS Viya platform CLI is required.

```
sas-viya --profile c1 --insecure auth login

sas-viya plugins install --repo SAS authorization

./sas-viya authorization update-rule --id jobexec-files-1 --condition
"clientId() == 'sas.SASJobExecution' || clientId() ==
'sas.jobExecution' || clientId() == 'sas.compute'" || clientId() ==
'sas.files'
```

SAS Detection Architecture

Before Deployment Commands

Revised Configuration YAML File

The update contains a revision to the primary configuration file, **`/$deploy/site-config/sas-detection/detection-engine-deployment.yaml`**. In order for SAS Detection Architecture to continue to work smoothly, you must revise that file before the update.

1 In your deployment assets, search for a **/$deploy/site-config/sas-detection/detection-engine-deployment.yaml** file. If that file is not present, then you do not have SAS Detection Architecture and should skip the rest of this deployment note. If the file is present, continue with this procedure.

2 In the **/$deploy/site-config/sas-detection/detection-engine-deployment.yaml** file, find the following code block.

```
### Only the NAMESPACE must be specified
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: reader
  namespace: {{ NAMESPACE }}
rules:
  - apiGroups: [""]
    resources: ["deployments", "deployments/scale", "pods"]
    verbs: ["get", "watch", "list"]
  - apiGroups: ["apps"]
    resources: ["deployments", "deployments/scale", "pods"]
    verbs: ["get", "watch", "list"]
---
```

In your file, the variable `{{ NAMESPACE }}` has been replaced with the name of your namespace.

3 In that code block, revise the `resources` line (highlighted in the example above) by adding `"services"` and `"endpoints"`. The revised code block should look like this:

```
### Only the NAMESPACE must be specified
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: reader
  namespace: {{ NAMESPACE }}
rules:
  - apiGroups: [""]
    resources: ["deployments", "deployments/scale", "pods", "services", "endpoints"]
    verbs: ["get", "watch", "list"]
  - apiGroups: ["apps"]
    resources: ["deployments", "deployments/scale", "pods", "services", "endpoints"]
    verbs: ["get", "watch", "list"]
---
```

2025.03 (March 2025)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Updated Storage Class Transformer

Note: This deployment note is repeated from version 2025.01 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 to target only sas-programming-environment podTemplates that make use of the viya volume. 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 the transformers block of 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:

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      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
```

Note: For more information about using the change-viya-volume-storage-class.yaml file, see the README file located 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).

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

Update Configuration for SAS Servers Using Kerberos on OpenShift

Note: If your deployment is not running on Red Hat OpenShift, skip this deployment note.

Because of differences in how Red Hat OpenShift handles Kerberos, SAS is introducing new transformers specifically for deployments on Red Hat OpenShift. The new transformers take the place of transformers in use prior to the update.

1 Search the transformers block of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following entry:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-tls.yaml
```

If you find the entry, replace it with the following line:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-tls-openshift.yaml
```

2 Search the transformers block of the base kustomization.yaml file for the following entry:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-no-tls.yaml
```

If you find the entry, replace it with the following line:

```
- sas-bases/overlays/kerberos/sas-servers/sas-kerberos-deployment-no-tls-openshift.yaml
```

3 If neither entry is in your base kustomization.yaml file, no further action is required.

Revised SAS Configurator for Open Source Transformer

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised to change the list of supported pip install packages. The change-configuration.yaml transformer file is 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 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.

5 After the change-configuration.yaml file has been revised, you can delete the change-configuration2.yaml file.

For more information about the change-configuration.yaml file, see "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

After Deployment Commands

Multiple Configuration Definitions for SAML Identity Providers

Note: This deployment note is repeated from version 2025.02 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.

SAML libraries have been revised and no longer allow multiple configurations with duplicate entityID values. You should perform the following steps to determine if you have configurations with duplicate values. If you have duplicate values, the procedure also provides information for addressing the issue.

1 To list all the configuration properties under config/SASLogon that contain "saml":

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse config/SASLogon/" | grep saml
```

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

2 Evaluate the output of the command in Step 1 to determine if it contains multiple idpMetadata properties. If there are not more than one idpMetadata properties in the output, you are not affected by the issue and you can skip the rest of this note. If there are more than one idpMetadata properties in the output, continue with the next step.

3 Review the contents for each idpMetadata property. Determine whether the entityID value is unique for each or if there are identical entityID values. If you do not have identical entityID values, you are not affected by the issue and you can skip the rest of this note. If you have identical entityID values, continue with the next steps.

Note: You might see XML returned for the idpMetadata property. You can view the XMl directly in the command output to check the entityID values. You might

also see a URI returned for the idpMetadata property. If a URI is returned for the idpMetadata property, you must enter that URI in a browser to view the XML and check the entityID values.

4 Perform the following command:

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse configurationservice/configurations/" |
grep sas.logon.saml.providers
```

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

5 Create a case at the Customer Service Portal in order to receive assistance from SAS Technical Support for the rest of the procedure. In the case that you create, be sure that you include the output of the command in Step 4.

Multiple OIDC Configurations with Same Issuer Property

OIDC support has been revised and no longer allows multiple configurations with duplicate issuer values. You should perform the following steps to determine if you have configurations with duplicate values. If you have duplicate values, the procedure also provides information for addressing the issue.

1 To list all the configuration properties under config/SASLogon that contain "oidc":

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse config/SASLogon/" | grep oauth
```

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

2 Evaluate the output of the command in Step 1 to determine if it contains multiple issuer properties. If the output of the command is a single item, you are not affected by the issue and you can skip the rest of this note. If there is more than one item in the output, continue with the next step.

3 Perform the following command:

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse configurationservice/configurations/" |
grep sas.logon.oauth.providers
```

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

4 Create a case at the Customer Service Portal in order to receive assistance from SAS Technical Support for the rest of the procedure. In the case that you create, be sure that you include the output of the command in Step 3.

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 2025.03 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects.

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken:

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in “Restore Projects After Update”.

Delete espserver Before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS/ACCESS to Impala

Before Deployment Commands

Impala ODBC Driver Client Change

Prior to this release, SAS/ACCESS to Impala used the Cloudera Impala ODBC driver client. With this release, SAS/ACCESS to Impala now uses the Simba Impala ODBC driver client. The change requires users of SAS/ACCESS to Impala to evaluate their configuration and potentially make revisions to account for the new ODBC driver client.

1. Check the **$deploy/site-config/data-access/sas-access.properties** file for the `IMPALA`. If you find it, SAS recommends that it be removed from the file.
2. If you have set any variables in the Impala INI file, revise the **$deploy/site-config/data-access/sas-access.properties** to remove the `CLOUDERAIMPALAINI` property and replace it with a `SIMBAIMPALAINI` property. Therefore, if the properties file contains `CLOUDERAIMPALAINI=`*`path-to-the-Impala-INI-file`*, it should be replaced with `SIMBAIMPALAINI=`*`path-to-the-Impala-INI-file`*

If you have not set any variables in the Impala INI file, SAS recommends that the `CLOUDERAIMPALAINI` property be removed from the **`$deploy/site-config/data-access/sas-access.properties`** file.

SAS/ACCESS to Spark

After Deployment Commands

Revise JREOPTION for Updated JDBC Data Connector

The Simba Databricks JDBC Data Connector uses the Apache Arrow library to efficiently transfer result sets from Databricks to the client. The Arrow library requires the JRE option `JREOPTION --add-opens=java.base/java.nio=ALL-UNNAMED` to be passed to the SAS Java Runtime Environment at startup. This property must be set in the SAS Studio compute context, the SAS batch server context, and the CAS compute context. When this property is not set, SAS returns an error when attempting to read data from Databricks.

For the steps to revise the property in the appropriate contexts, see "Resolution" in *SAS Viya Platform: Deployment Guide*.

2025.02 (February 2025)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Updated Storage Class Transformer

Note: This deployment note is repeated from version 2025.01 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 to target only sas-programming-environment podTemplates that make use of the viya volume. 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 the transformers block of 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:

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      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
```

Note: For more information about using the change-viya-volume-storage-class.yaml file, see the README file located 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).

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

After Deployment Commands

Multiple Configuration Definitions for SAML Identity Providers

SAML libraries have been revised and no longer allow multiple configurations with duplicate entityID values. You should perform the following steps to determine if you have configurations with duplicate values. If you have duplicate values, the procedure also provides information for addressing the issue.

1 To list all the configuration properties under config/SASLogon that contain "saml":

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse config/SASLogon/" | grep saml
```

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

2 Evaluate the output of the command in Step 1 to determine if it contains multiple idpMetadata properties. If the output of the command is a single item, you are not affected by the issue and you can skip the rest of this note. If there is more than one item in the output, continue with the next step.

3 Review the contents for each idpMetadata property. Determine whether the entityID value is unique for each or if there are identical entityID values. If you do not have identical entityID values, you are not affected by the issue and you can skip the rest of this note. If you have identical entityID values, continue with the next steps.

Note: You might see XML returned for the idpMetadata property. You can view the XMl directly in the command output to check the entityID values. You might also see a URI returned for the idpMetadata property. If a URI is returned for the idpMetadata property, you must enter that URI in a browser to view the XML and check the entityID values.

4 Perform the following command:

```
kubectl -n name-of-namespace exec sas-consul-server-0 -- bash -c "export
CONSUL_HTTP_ADDR=https://localhost:8500; /opt/sas/viya/home/bin/sas-
bootstrap-config kv read --recurse configurationservice/configurations/" |
grep sas.logon.saml.providers
```

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

5 Create a case at the Customer Service Portal in order to receive assistance from SAS Technical Support for the rest of the procedure. In the case that you create, be sure that you include the output of the command in Step 4.

Configure the Multi-Language Python Library for the Enhanced SAS Compute Server

With this update, SAS has released enhancements to the SAS Compute Server (for details of the enhancements, see "Enhancements to the SAS Compute Server" in *SAS Programmer's Guide: Essentials*.) The enhancements take advantage of the configuration of Python. If you have not already configured Python and want to use it in your deployment, use the procedure described at "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*. If you have configured Python before this update, ensure that you have performed step 2d and step 2e in "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*. Both steps are new with this update.

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 2025.02 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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 Allowance for Credit Loss

Before Deployment Commands

SAS Allowance for Credit Loss Configuration Changes

The process of configuring SAS Allowance for Credit Loss has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Allowance for Credit Loss, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-acl/resources/acl_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete steps 3-9 in the "Installation" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-acl/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_allowance_for_credit_loss_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Asset and Liability Management

Before Deployment Commands

SAS Asset and Liability Management Configuration Changes

The process of configuring SAS Asset and Liability Management has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Asset and Liability Management, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-alm/resources/alm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete steps 3-7 in the "Installation" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-alm/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_asset_and_liability_management_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Dynamic Actuarial Modeling

Before Deployment Commands

SAS Dynamic Actuarial Modeling Configuration Changes

The process of configuring SAS Dynamic Actuarial Modeling has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Dynamic Actuarial Modeling, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete steps 3-6 in the "Installation" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-pcpricing/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_dynamic_actuarial_modeling_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Event Stream Processing

Before Deployment Commands

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken:

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in “Restore Projects After Update”.

Delete espserver Before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS Expected Credit Loss

Before Deployment Commands

SAS Expected Credit Loss Configuration Changes

The process of configuring SAS Expected Credit Loss has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Expected Credit Loss, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-ecl/resources/ecl_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete steps 3-9 in the "Installation" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-ecl/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_expected_credit_loss_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Insurance Capital Management

Before Deployment Commands

SAS Insurance Capital Management Configuration Changes

The process of configuring SAS Insurance Capital Management has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Insurance Capital Management, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-icm/resources/icm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration for SAS Insurance Capital Management" section of the README file located at **`$deploy/sas-bases/examples/sas-risk-cirrus-icm/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_sas_insurance_capital_management_for_deployment.htm`** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Insurance Contract Valuation Foundation

Before Deployment Commands

SAS Insurance Contract Valuation Foundation Configuration Changes

The process of configuring SAS Insurance Contract Valuation Foundation has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Insurance Contract Valuation Foundation, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-icv/resources/icv_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration Files for SAS Insurance Contract Valuation Foundation" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-icv/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_insurance_contract_valuation_foundation_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Integrated Regulatory Reporting

Before Deployment Commands

SAS Integrated Regulatory Reporting Configuration Changes

The process of configuring SAS Integrated Regulatory Reporting has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Integrated Regulatory Reporting, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-eba/resources/eba_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration Files for SAS Integrated Regulatory Reporting" section of the README file located at **`$deploy/sas-bases/examples/sas-risk-cirrus-eba/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_sas_integrated_regulatory_reporting_for_deployment.htm`** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Model Risk Management

Before Deployment Commands

SAS Model Risk Management Configuration Changes

The process of configuring SAS Model Risk Management has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Model Risk Management, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-mrm/resources/mrm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration Files Related to SAS Model Risk Management" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-mrm/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_model_risk_management_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Regulatory Capital Management

Before Deployment Commands

SAS Regulatory Capital Management Configuration Changes

The process of configuring SAS Regulatory Capital Management has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Regulatory Capital Management, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-rcm/resources/rcm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration Files Related to SAS Regulatory Capital Management" section of the README file located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rcm/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_sas_regulatory_capital_management_for_deployment.htm`** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Risk Cirrus Core

Before Deployment Commands

SAS Risk Cirrus Core Configuration Changes

The process of configuring SAS Risk Cirrus Core has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Risk Cirrus Core, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-core/resources/core_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration for Risk Cirrus Core" section of the README file located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rcc/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm`** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Risk Factor Manager

Before Deployment Commands

SAS Risk Factor Manager Configuration Changes

The process of configuring SAS Risk Factor Manager has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Risk Factor Manager, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-rfm/resources/rfm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Perform the tasks described in the "Modify the Configuration Files Related to SAS Risk Factor Manager" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-rfm/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_factor_manager_for_deployment.htm** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Risk Modeling

Before Deployment Commands

SAS Risk Modeling Configuration Changes

The process of configuring SAS Risk Modeling has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Risk Modeling, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**`$deploy/kustomization.yaml`**) for the following line:

```
- site-config/sas-risk-cirrus-rm/resources/rm_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete the steps in the "Installation" section of the README file located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rm/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_sas_risk_modeling_for_deployment.htm`** (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

SAS Risk Offerings

Before Deployment Commands

Enable Risk Cirrus Objects Features

The SAS Risk Cirrus Objects Microservice stores information related to business object definitions and business object instances such as analysis data, analysis

runs, models or model reviews. Cirrus Objects also stores and retrieves items related to business objects, such as attachments. These business objects are associated with the Risk Cirrus Platform that underlies most SAS Risk offerings.

A new feature for enabling the promotion of Cirrus Objects across SAS Viya platform environments has been introduced. For more details, see *What's New in SAS Risk Cirrus*.

Perform the following steps to determine if your deployment is affected and, if so, how to enable the new feature Risk Cirrus Objects.

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

```
- site-config/risk-cirrus-builder/resources/sas_risk_cirrus_builder_transform.yaml
```

If the line is present, then continue with the steps in this deployment note. If it is not present, your deployment is unaffected and you can skip the rest of this note.

2 Perform the steps described in the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-objects/resources/README.md** (for Markdown format) and **$deploy/sas-bases/docs/configure_environment_id_settings_for_sas_risk_cirrus_builder_microservice.htm** (for HTML format).

SAS Stress Testing

Before Deployment Commands

SAS Stress Testing Configuration Changes

The process of configuring SAS Stress Testing has been changed to make updates easier in the future. These changes require manual steps because the configuration information has been moved to different files. Additionally, Git is no longer required for SAS Stress Testing, and the changes remove content about Git from the configuration process.

1 To determine if your deployment includes the affected files, search the transformers section of the base kustomization.yaml file (**$deploy/kustomization.yaml**) for the following line:

```
- site-config/sas-risk-cirrus-st/resources/st_transform.yaml
```

If the line is present, continue with the steps in this deployment note. If the line is not present, you can skip the rest of this deployment note.

2 Complete steps 3-9 in the "Installation" section of the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-st/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/**

preparing_and_configuring_sas_stress_testing_for_deployment.htm (for HTML format).

Note: After you have completed the tasks in this deployment note, be sure to also review the generic SAS Risk deployment note, "Enable Risk Cirrus Objects Features" on page 280.

2025.01 (January 2025)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Updated Storage Class Transformer

The change-viya-volume-storage-class.yaml file has been modified to target only sas-programming-environment podTemplates that make use of the viya volume. 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 the transformers block of 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:

```
patch: |-
  - op: add
    path: /template/spec/volumes/-
    value:
      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
```

Note: For more information about using the change-viya-volume-storage-class.yaml file, see the README file located 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).

5 After your revisions are complete, delete the change-viya-volume-storage-class2.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 2025.01 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken:

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in “Restore Projects After Update”.

Delete espserver Before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

2024.12 (December 2024)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Required Ingress Controller Configuration

If you are using ingress-nginx 1.12.*x* and later for your Kubernetes cluster, a change to the default configuration from NGINX will affect your deployment. For more information, see Required Ingress Controller Configuration.

Red Hat OpenShift and Root Routes

Note: If your deployment is not using Red Hat OpenShift, skip this note.

Note: If your deployment is maintained with the SAS Viya Platform deployment operator or the sas-orchestration command, skip this deployment note. This note only applies to deployments maintained using Kubernetes commands.

Red Hat OpenShift uses routes to allow you to host your application at a public URL. It can either be secure or unsecured, depending on the network security configuration of your application. Prior to this version, the root routes redirected to **/SASDrive**. However, with this update, **/SASDrive** has been retired and the root routes redirect to **/SASLanding**. The change happens automatically with the update if the existing root routes are deleted. Therefore, deployments on Red Hat OpenShift must perform the following command in order to use the new root routes:

```
kubectl delete route sas-drive-app-root sas-drive-app-root-multitenant -n name-of-namespace
```

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 2024.12 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

Guest Access

If you have enabled guest access for users of SAS Job Execution Web Application, a manual step is required after an update to 2024.12. An authorization rule has changed, and it potentially affects guest access.

Users who are logged in to the Guest account will see errors from the file service when they attempt to run jobs. The errors state, "`Initial Files Service creation of file failed`." To avoid these errors, modify an authorization rule after you have deployed the updates. Use the SAS Viya platform command-line interface to run the following commands:

Note: The authorization plug-in to the SAS Viya platform CLI is required.

```
sas-viya --profile c1 --insecure auth login

sas-viya plugins install --repo SAS authorization

./sas-viya authorization update-rule --id jobexec-files-1 --condition
"clientId() == 'sas.SASJobExecution' || clientId() ==
'sas.jobExecution' || clientId() == 'sas.compute'"
```

SAS Event Stream Processing

Before Deployment Commands

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken:

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in “Restore Projects After Update”.

Delete espserver Before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS Model Repository Service

Before Deployment Commands

Starting in 2024.12, service accounts are created for some SAS services by default. Previously, an overlay was used to create these service accounts for the Model Repository service, enabling it to run with additional permissions in a Red Hat OpenShift environment. Two transformers in this overlay are no longer needed. To avoid failures when you perform an update on OpenShift, references to two YAML files must be removed.

Note: The Model Repository service is part of the SAS Viya Platform deployment for SAS Model Manager, SAS Intelligent Decisioning, and Model Studio.

Before you perform an update for the Model Repository service in a Red Hat OpenShift environment, perform these steps:

1. Use your preferred text editor to modify the base kustomization.yaml file (**`$deploy/kustomization.yaml`**).
2. From the resources block, remove the reference to **`sas-bases/overlays/sas-model-repository/service-account/sa.yaml`**.

3 From the transformers block, remove the reference to **sas-bases/overlays/sas-model-repository/service-account/sa-transformer.yaml**.

4 Save your changes to the file.

2024.11 (November 2024)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Ensure RabbitMQ Feature Flags Are Enabled

IMPORTANT The SAS Viya platform uses RabbitMQ as its messaging and streaming broker. The SAS Viya platform requires that RabbitMQ be updated periodically along with the rest of your SAS Viya platform software. To ensure that RabbitMQ is updated successfully and thus provide the

framework for a successful SAS Viya platform update, run the following command:

```
kubectl -n name-of-namespace exec -it sas-rabbitmq-server-0 -- bash -c
"source ~/.bashrc ; rabbitmqctl enable_feature_flag all"
```

Required Ingress Controller Configuration

If you are using ingress-nginx 1.12.*x* and later for your Kubernetes cluster, a change to the default configuration from NGINX will affect your deployment. For more information, see Required Ingress Controller Configuration.

Change to CAS Backing Store Enablement

The method by which the CAS backing store is enabled has been changed. Previously, the backing store was enabled by modifying the custom resource for the CAS operator. Now, you must enable the backing store in four different scenarios using patch transformer .yaml files. To determine whether you have enabled the backing store for your current deployment, run the following command:

```
kubectl -n name-of-namespace get cas default -
o=jsonpath={.spec.controllerTemplate.spec.containers[0].env} | grep
TK_BACKING_STORE_DIR > /dev/null;echo $?
```

If the output is `0`, then the CAS backing store is enabled in your current deployment, and you must use the new transformer files to enable it in your updated deployment. If the output is `1`, then the backing store has not been enabled and you can either skip the rest of this deployment note or enable the backing store for the first time.

There are four transformers to choose from:

- If you are using CAS auto-resourcing, **$deploy/sas-bases/examples/cas/configure/cas-enable-default-backing-store.yaml** causes the CAS operator to select an appropriate size for the backing store.
- **$deploy/sas-bases/examples/cas/configure/cas-enable-backing-store.yaml** selects a specific size for the backing store.
- **$deploy/sas-bases/examples/cas/configure/cas-enable-backing-store-with-priority-groups.yaml** selects a specific size for five separate backing stores, one for each priority group.
- **$deploy/sas-bases/examples/cas/configure/cas-enable-backing-store-with-priority-group-one.yaml** selects specific sizes for two backing stores, one for users in priority group one and a second for all other users. This is appropriate only when CAS resource management in enabled.

To use the new transformer files:

1. Copy the patch transformer that you want to use from **$deploy/sas-bases/examples/cas/configure/*file-name*** to **$deploy/site-config/cas/configure/*file-name***.
2. Follow the instructions in the comments of the copied patch transformer to replace variables with the appropriate values.

3 In the base kustomization.yaml file (**$deploy/kustomization.yaml**), add a line to the transformers block that points to the patch transformer you want to use. Here is an example using the transformer that uses a specific size for the backing store:

```
transformers:
...
- site-config/cas/configure/cas-enable-backing-store.yaml
```

Replace `cas-enable-backing-store.yaml` with the name of the transformer you hve chosen to use..

For more information about the new required .yaml files, see the "Enable a Backing Store for CAS" 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).

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 2024.11 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

Guest Access

If you have enabled guest access for users of SAS Job Execution Web Application, a manual step is required after an update to 2024.11. An authorization rule has changed, and it potentially affects guest access.

Users who are logged in to the Guest account will see errors from the file service when they attempt to run jobs. The errors state, "`Initial Files Service creation of file failed`." To avoid these errors, modify an authorization rule after you have deployed the updates. Use the SAS Viya platform command-line interface to run the following commands:

Note: The authorization plug-in to the SAS Viya platform CLI is required.

```
sas-viya --profile c1 --insecure auth login

sas-viya plugins install --repo SAS authorization

./sas-viya authorization update-rule --id jobexec-files-1 --condition
"clientId() == 'sas.SASJobExecution' || clientId() ==
'sas.jobExecution' || clientId() == 'sas.compute'"
```

SAS Event Stream Processing

Before Deployment Commands

PVC Changes

SAS Event Stream Processing now creates a PersistentVolumeClaim (PVC) with a default storage capacity of 5 GB. Here are the steps to implement the new process and remove the files that supported the previous process:

1 In the base kustomization.yaml file (**`$deploy/kustomization.yaml`**), remove the following lines:

- In the resources block, remove `- site-config/sas-event-stream-processing-studio-app/pvc/resources.yaml`
- In the transformer block, remove the following:
  - `- sas-bases/overlays/sas-event-stream-processing-studio-app/data/data-transformer.yaml`

- `- sas-bases/overlays/sas-event-stream-manager-app/data/data-transformer.yaml`
- `- sas-bases/overlays/sas-event-stream-processing-studio-app/pvc/pvc-transformer.yaml`
- `- sas-bases/overlays/sas-event-stream-manager-app/pvc/pvc-transformer.yaml`

2 Copy the files in the **`$deploy/sas-bases/examples/sas-event-stream-processing-studio-app/storage`** directory to the **`$deploy/site-config/sas-event-stream-processing-studio-app/storage`** directory. Create the destination directory if it does not exist.

3 Review the storage-capacity value in the current **`$deploy/site-config/sas-event-stream-processing-studio-app/pvc/resources.yaml`** file. Update the **`$deploy/site-config/sas-event-stream-processing-studio-app/storage/esp-storage-size-transformer.yaml`** file with that same value.

4 Revise the base kustomization.yaml file by adding `- site-config/sas-event-stream-processing-studio-app/storage/esp-storage-size-transformer.yaml` to the transformers block. The resulting kustomization.yaml should look like this:

```
transformers:
...
- site-config/sas-event-stream-processing-studio-app/storage/esp-storage-size-transformer.yaml
...
```

5 Also in the base kustomization.yaml file, in the patches block, add `sas-event-stream-processing-studio-app` to the `annotationSelector` line of the `site-config/storageclass.yaml` patch. 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,sas-cas-operator,sas-pyconfig,
sas-risk-cirrus-search,sas-risk-modeling-core,sas-event-stream-processing-studio-app)
```

Note: For more details about the SAS Event Stream Processing PVC, see the README file located at **`$deploy/sas-bases/examples/sas-event-stream-processing-studio-app/storage/README.md`** (for Markdown format) and at **`$deploy/sas-bases/docs/configuring_file_storage_for_sas_event_stream_processing_studio_projects.htm`** (for HTML format).

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in “Restore Projects After Update”.

Delete espserver Before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

2024.10 (October 2024)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Required Ingress Controller Configuration

If you are using ingress-nginx 1.12.*x* and later for your Kubernetes cluster, a change to the default configuration from NGINX will affect your deployment. For more information, see Required Ingress Controller Configuration.

Revised SAS Configurator for Open Source Transformer

The change-configuration.yaml transformer file for SAS Configurator for Open Source has been revised to support a newer version of Python and is 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. 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. Note that the `default_py.python_signature` and `default_py.python_tarball` properties have been modified to support a newer version of Python. SAS recommends that you keep the values for these properties.

For more information about the change-configuration.yaml file, see "Configure Python Integration Using SAS Configurator for Open Source" in *SAS Viya Platform: Integration with External Languages*.

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 2024.10 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

Guest Access

If you have enabled guest access for users of SAS Job Execution Web Application, a manual step is required after an update to 2024.10. An authorization rule has changed, and it potentially affects guest access.

Users who are logged in to the Guest account will see errors from the file service when they attempt to run jobs. The errors state, "`Initial Files Service creation of file failed`." To avoid these errors, modify an authorization rule

after you have deployed the updates. Use the SAS Viya platform command-line interface to run the following commands:

Note: The authorization plug-in to the SAS Viya platform CLI is required.

```
sas-viya --profile c1 --insecure auth login

sas-viya plugins install --repo SAS authorization

./sas-viya authorization update-rule --id jobexec-files-1 --condition
"clientId() == 'sas.SASJobExecution' || clientId() ==
'sas.jobExecution' || clientId() == 'sas.compute'"
```

SAS Event Stream Processing

Before Deployment Commands

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
```

```
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in "Restore Projects After Update".

Delete espserver Before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS Risk Cirrus Core

Before Deployment Commands

Updated Configuration File

SAS Risk Cirrus Core is a common layer that is used by all Cirrus solutions. The main configuration file for SAS Risk Cirrus Core, core_transform.yaml, has been revised with two new properties. While the new properties are optional, users of

SAS Risk Solutions should be aware of their presence and, if desired, take steps to add the new properties to their deployment.

1 To determine if your deployment includes SAS Risk Cirrus Core, go to the base kustomization.yaml file (**$deploy/kustomization.yaml**). In the transformers block, search for "core_transform.yaml". If that phrase is present, then you have SAS Risk Cirrus Core in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.

2 Review the SAS Risk Cirrus Core README file for information about the two new variables, `SAS_RISK_CIRRUS_CODE_LIB_APPEND_NAMESPACE_TO_BRANCH_FLG` and `SAS_RISK_CIRRUS_CODE_LIB_SKIP_PUSH_IF_BRANCH_EXISTS`. Based on the information in the README, decide if you want to add either or both of the properties to your deployment. If you decide to add the properties, continue with these steps. If you do not want to add either of the new properties, skip the rest of this note.

   The SAS Risk Cirrus Core README file is located at **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

3 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 `core_transform2.yaml`. For example, if the file is located at **$deploy/site-config/sas-risk-cirrus-rcc/resources/core_transform.yaml**, change the file name to **$deploy/site-config/sas-risk-cirrus-rcc/resources/core_transform2.yaml**.

4 Copy the **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/core_transform.yaml** file to the directory you used in step 2. You should now have a core_transform.yaml file and a core_transform2.yaml file in the same directory.

5 Follow the steps in the SAS Risk Cirrus Core README file to add the values for the new variables, `SAS_RISK_CIRRUS_CODE_LIB_APPEND_NAMESPACE_TO_BRANCH_FLG` and `SAS_RISK_CIRRUS_CODE_LIB_SKIP_PUSH_IF_BRANCH_EXISTS`, to the core_transform.yaml file.

6 Use the core_transform2.yaml file as a guide for filling in the rest of the values in the core_transform.yaml file.

7 When the core_transform.yaml file is complete, delete the core_transform2.yaml file.

2024.09 (September 2024)

Additional Documentation

Before Deployment Commands

Deployment Notes and Critical What's New Topics

When updating to a new version of the software, be sure to review both the Deployment Notes and the What's New topics. The Deployment Notes alert you to changes to the deployment process. The What's New topics can include critical changes, such as changes to system requirements and pod names. For both the Deployment Notes and the What's New topics, be sure to review each note after the version that is currently deployed, up to and including the version to which you are updating.

What's New topics with critical changes are identified by "Critical Change" in the title. The What's New topics are located at *What's New in SAS Viya Platform Operations*.

General Deployment

Before Deployment Commands

Upgrade to PostgreSQL 16

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

IMPORTANT Before you attempt the procedure described in this deployment note, complete the procedures described in every other deployment note within your current update path that is listed in a "Before Deployment Commands" section. This deployment note should be the last

procedure you perform from a deployment note in a "Before Deploymnt Commands" section. After you have completed the procedure in this deployment note, perform the deployment/update commands and then proceed with any deployment notes listed in an "After Deployment Commands" section.

SAS is upgrading the version of PostgreSQL used in the internal instances of PostgreSQL (Crunchy Postgres from Crunchy Data) from version 12 to version 16. This deployment note contains the necessary steps to perform the PostgreSQL upgrade on a running SAS Viya platform deployment.

1 Take a full pgBackRest backup. For more information, see "Take an Ad Hoc pgBackRest Backup " in *SAS Viya Platform: Infrastructure Servers*.

2 Consider changing the PostgreSQLdata storage to block storage. Block storage might improve PostgreSQL performance and, more importantly, it can help you avoid a potential PostgreSQL upgrade issue on NFS storage. For details, see "Change the Storage Class of the Data Pod" in *SAS Viya Platform: Infrastructure Servers*.

3 Check the current checksum setting for the database.

   To upgrade to Crunchy Postgres 16, the checksum must be `on`. If it is `off`, the upgrade process detects the current checksum setting and automatically enables it. Checksums can cause a slight Write overhead, but the overhead is considered worthwhile for the enhanced data integrity.

   a Determine the name of the cluster or clusters.

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

   The output of this command is one or two cluster names.

   b Use each of the cluster names in the following commands. If you have two cluster names as the output, perform the commands twice, once for each cluster name:

   ```
   POD1=$(kubectl -n name-of-namespace get pods --selector="postgres-
   operator.crunchydata.com/cluster=cluster-name,postgres-
   operator.crunchydata.com/role=master" -o name)
   ```

   ```
   kubectl exec -it $POD1 -c database -n name-of-namespace -- psql -c 'show
   data_checksums;'
   ```

   The output of the commands contains the term `on` or `off`.

   - If the current checksum setting is `on`, the upgrade should take 5-10 minutes for each PostgreSQL cluster.
   - If the current checksum setting is `off`, the upgrade may take some extra time in order to enable checksum across the database. You can check the database size with the following command:

     ```
     kubectl exec -it $POD1 -c database -n name-of-namespace -- psql -c
     '\l+'
     ```

     For each 100 Gi in the database, add approximately one hour to the time required for the upgrade.

Note: If you are using the SAS Viya Platform Deployment Operator or the sas-orchestration command to update your software, you should skip Step 4 through Step 8. Only perform those steps if you are using the Kubernetes command deployment method.

4 **(Kubernetes command deployment method only)** Create the site.yaml file.

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

5 **(Kubernetes command deployment method only)** As a Kubernetes cluster-level administrator, go to the upgrade script and change its permissions:

```
cd $deploy/sas-bases/examples/crunchydata/postgres-upgrade
```

```
chmod +x postgres-upgrade.sh
```

6 **(Kubernetes command deployment method only)** Using bash, run the upgrade script with the indicated parameters:

Note: The upgrade script requires the `yq` command (version 4) to be installed. For information about how to install the `yq` command, see https://github.com/mikefarah/yq.

```
./postgres-upgrade.sh \
location-of-admin-kubeconfig \
name-of-namespace \
$deploy/site.yaml \
debug-log \
2>&1 | tee ~/postgres-upgrade-$(date +'%m%d-%H%M%S').log
```

Running the script sends some log files to the pgupgrade-log directory under home.

7 **(Kubernetes command deployment method only)** After the PostgreSQL upgrade has completed, the vacuumdb task is performed and the PostgreSQL extensions are upgraded. The PostgreSQL cluster can be used normally while these actions are performed, including continuing to update your SAS Viya platform deployment.

8 **(Kubernetes command deployment method only)** Change the directory to the $deploy directory. Delete the site.yaml file that was created in Step 4.

9 After you perform the deployment commands to update your software, perform the tasks described at "PostgreSQL 16 Post-Upgrade Processing" on page 306.

CAUTION

The PostgreSQL upgrade process is not idempotent, so the manual script or the lifecycle operation should not be run more than once. If any issues occur during their execution, contact SAS Technical Support to remediate the incomplete upgrade instead of retrying the upgrade. Retrying or attempting to fix the issue might make the PostgreSQL cluster unrecoverable.

Required Ingress Controller Configuration

If you are using ingress-nginx 1.12.*x* and later for your Kubernetes cluster, a change to the default configuration from NGINX will affect your deployment. For more information, see Required Ingress Controller Configuration.

File Used to Identify fsGroup Has Changed

The fsGroup field defines a special supplemental group that assigns a GID for all containers in the pod. Volumes that support ownership management are modified to be owned and writable by the GID specified in fsGroup. For ease of maintenance, the fsGroup identification has been moved to a different file.

1 To determine if you have modified the fsGroup value in your current deployment, look for the following line in the transformers block of your base kustomization.yaml file (**`$deploy/kustomization.yaml`**):

```
- site-config/security/container-security/update-fsgroup.yaml
```

If the line is present, then complete the steps in this deployment note. If the line is not present, there is no work to be done and you should skip the rest of this deployment note.

2 Perform the steps described in the "Update the fsGroup Field" section of the README file at **`$deploy/sasbases/examples/security/container-security/README.md`** (for Markdown format) or **`$deploy/sas-bases/docs/modify_container_security_settings.htm`** (for HTML format).

3 In the transformer block of the base kustomization.yaml file, remove `- site-config/security/container-security/update-fsgroup.yaml`.

After Deployment Commands

PostgreSQL 16 Post-Upgrade Processing

Note: These steps for configuring PostgreSQL after it has been upgraded are applicable to all deployment methods. If you used the Kubernetes commands deployment method, and if you did not perform the steps described in "Upgrade to PostgreSQL 16" on page 303, do not perform the steps in this deployment note.

CAUTION
The PostgreSQL upgrade process is not idempotent, so the manual script or the lifecycle operation should not be run more than once. If any issues occurred during their execution, contact SAS Technical Support to remediate the incomplete upgrade instead of retrying the upgrade. Retrying or attempting to fix the issue might make the PostgreSQL cluster unrecoverable.

For more information about post-upgrade processing, see the following documentation from Crunchy Data: Step 6: Complete the Post-Upgrade Tasks.

Note: If applicable, repeat the following steps for both the platform PostgreSQL cluster and the CDS PostgreSQL cluster.

1 To determine the names of your PosgreSQL clusters, run the following command:

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

The output of this command is a cluster name that includes "platform" (such as sas-crunchy-platform-postgres) and, if you have a CDS PostgreSQL cluster, a cluster name that includes "cds" (such as sas-crunchy-cds-postgres).

In the remaining steps in this deployment note, the variable *cluster-name* should be replaced by the actual cluster name. If you have two clusters, run the command with the *cluster-name* variable twice, once for each cluster.

2 (Optional) Delete the old data directory.

a Exec into the primary pod of each cluster.

```
POD1=$(kubectl -n name-of-namespace get pods --selector="postgres-
operator.crunchydata.com/cluster=cluster-name,postgres-
operator.crunchydata.com/role=master" -o name)
```

```
kubectl exec -it $POD1 -c database -n name-of-namespace -- bash
```

After the second command, you will have a bash prompt, which you can use to perform additional steps.

b Check the PostgreSQL version:

```
psql -c "select version()"
```

If the version is not 16.*x*, the PostgreSQL upgrade might have failed. Stop and contact SAS Technical Support.

c Determine if pg12 and pg12_wal directories are present in the leader pod:

```
ls -la /pgdata
```

If the output of the command does not include information about pg12 and pg12_wal directories, you might have had a failover. You should instead check replica nodes for the pg12 directories. Use the `patronictl list` command to show the list of replicas.

d Execute the following commands to rename or to remove the old data directory. Do so at your discretion:

```
mv -v /pgdata/pg12 /pgdata/pg12.old
```

```
mv -v /pgdata/pg12_wal /pgdata/pg12_wal.old
```

or

```
rm -rf /pgdata/pg12
```

```
rm -rf /pgdata/pg12_wal
```

e Close the shell session by entering the following command:

```
exit
```

3 (Optional) Clean up objects that are no longer needed:

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

```
kubectl -n name-of-namespace annotate postgrescluster cluster-name postgres-operator.crunchydata.com/allow-upgrade-
```

Note: The "-" at the end of the second command is not a typographical error and should be included.

IMPORTANT If you have upgraded PostgreSQL in a namespace that is in a cluster that contains other namespaces still at PostgreSQL 12, update those namespaces with the latest patches available for the version of the SAS Viya platform that is deployed in each namespace. For example, if the cluster contains a namespace at version 2024.07, find the latest patch for 2024.07 and update the namespace with that patch. This task might require coordination among namespace administrators.

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 2024.09 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

Guest Access

If you have enabled guest access for users of SAS Job Execution Web Application, a manual step is required after an update to 2024.09. An authorization rule has changed, and it potentially affects guest access.

Users who are logged in to the Guest account will see errors from the file service when they attempt to run jobs. The errors state, "`Initial Files Service creation of file failed`." To avoid these errors, modify an authorization rule after you have deployed the updates. Use the SAS Viya platform command-line interface to run the following commands:

Note: The authorization plug-in to the SAS Viya platform CLI is required.

```
sas-viya --profile c1 --insecure auth login

sas-viya plugins install --repo SAS authorization

./sas-viya authorization update-rule --id jobexec-files-1 --condition
"clientId() == 'sas.SASJobExecution' || clientId() ==
'sas.jobExecution' || clientId() == 'sas.compute'"
```

SAS Event Stream Processing

Before Deployment Commands

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in "Restore Projects After Update".

Delete espserver Before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

2024.08 (August 2024)

General Deployment

Before Deployment Commands

Required Ingress Controller Configuration

If you are using ingress-nginx 1.12.*x* and later for your Kubernetes cluster, a change to the default configuration from NGINX will affect your deployment. For more information, see Required Ingress Controller Configuration.

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 2024.08 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration' -n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l 'esp.iot.sas.com/server-type!=client-configuration' --show-managed-fields=false -o json | jq 'del(.items[].access)|del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-configuration"])|del(.items[].metadata.creationTimestamp)|del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
```

```
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in "Restore Projects After Update".

Delete espserver Before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

2024.07 (July 2024)

General Deployment

Before Deployment Commands

SAS Workload Orchestrator Configuration File Location Change

The transformer file that automatically disables SAS Workload Orchestrator has been moved. Therefore, the reference to its location in the base kustomization.yaml file ($deploy/kustomization.yaml) must be revised to account for the new location.

1 To determine whether your deployment is affected by this change, go to the base kustomization.yaml file. Search for the following line in the transformers block of that file:

```
- sas-bases/examples/sas-workload-orchestrator/enable-disable/sas-workload-orchestrator-disable-patch-transformer.yaml
```

If the line is present, continue with the steps in this note. If the line is not present, then you are unaffected by this change and should skip the rest of this note.

2 In the transformers block of the base kustomization.yaml file, revise the line that you searched for by replacing "examples" with "overlays":

```
transformers
...
- sas-bases/overlays/sas-workload-orchestrator/enable-disable/sas-workload-orchestrator-disable-patch-transformer.yaml
```

3 Because the version of the transformer in the examples subdirectory is no longer being used by your deployment, you should feel free to remove it.

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 2024.07 contains an update to CAS. However, until the existing CAS servers are restarted, your deployment continues 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
```

OpenSearch

Before Deployment Commands

OpenSearch Adds Configurable Parameters for Retaining Audit Log Indices

Note: This topic applies to internal instances of OpenSearch only.

OpenSearch is changing its audit log indices retention policy so that the indices are no longer retained indefinitely. Instead, the audit log indices are retained only for seven days by default. If you want to use a different retention period, OpenSearch has added a new transformer file, audit-log-retention-transformer.yaml, to configure the retention of audit log indices. For details about using this new file, see the README file at **$deploy/sas-bases/examples/configure-elasticsearch/internal/security-audit-logs/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/opensearch_security_audit_logs.htm** (for HTML format).

When the software is updated for the first time after the introduction of this new policy, any audit log indices that are older than the defined retention period are deleted.

SAS Event Stream Processing

Before Deployment Commands

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to update your software, be sure to perform the tasks described in "Restore Projects after Update".

Delete espserver before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects after Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS RFC Solution Configuration

Before Deployment Commands

Revised kafka_configuration_patch.yaml File

The kafka_configuration_patch.yaml file for SAS RFC Solution Configuration has been modified with new variables. Use the following steps to get the new version of the kafka_configuration_patch.yaml file in order to add the new variables while maintaining the rest of the content in that file.

1. To determine whether your deployment includes SAS RFC Solution Configuration, go to the base kustomization.yaml file (**$deploy/kustomization.yaml**). In the patches block, search for "kafka_configuration_patch.yaml". If that phrase is present, then you have SAS RFC Solution Configuration in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **$deploy/site-config/sas-rfc-solution-config/kafka-configuration-patch.yaml** and change the file name to `kafka_configuration_patch2.yaml`.
3. Copy the **$deploy/sas-bases/examples/sas-rfc-solution-config/configure-kafka/kafka-configuration-patch.yaml** file to **$deploy/site-config/sas-rfc-solution-config/kafka-configuration-patch.yaml**. You should now have a kafka_configuration_patch.yaml file and a kafka_configuraon_patch2.yaml file in the same directory.
4. Follow the steps in the SAS RFC Solution Configuration README file to add the values for the new variables. The README file is located at **$deploy/sas-bases/examples/sas-rfc-solution-config/configure-kafka/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_rfc_solution_configuration_patch_to_communicate_with_apache_kafka.htm** (for HTML format).
5. Use the kafka-configuration-patch2.yaml file as a guide for filling in the rest of the values in the kafka-configuration-patch.yaml file.
6. When the kafka-configuration-patch.yaml file is complete, delete the kafka-configuration-patch2.yaml file.

2024.06 (June 2024)

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 2024.06 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

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to updaate your software, be sure to perform the tasks described in “Restore Projects After Update”.

Delete espserver before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS Regulatory Capital Management

Before Deployment Commands

Revised rcm_transform.yaml File

The rcm_transform.yaml file for SAS Regulatory Capital Management has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the rcm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Regulatory Capital Management, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "rcm_transform.yaml". If that phrase is present, then you have SAS Regulatory Capital Management in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-rcm/resources/rcm_transform.yaml`** and change the file name to `rcm_transform2.yaml`.

3 Copy the **$deploy/sas-bases/examples/sas-risk-cirrus-rcm/resources/rcm_transform.yaml** file to **$deploy/site-config/sas-risk-cirrus-rcm/resources/rcm_transform.yaml**. You should now have an rcm_transform.yaml file and an rcm_transform2.yaml file in the same directory.

4 Follow the steps in the SAS Regulatory Capital Management README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the rcm_transform.yaml file. The README file is located at **$deploy/sas-bases/examples/sas-risk-cirrus-rcm/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_connection_settings_for_sas_regulatory_capital_management.htm** (for HTML format).

5 Use the rcm_transform2.yaml file as a guide for filling in the rest of the values in the rcm_transform.yaml file.

6 When the rcm_transform.yaml file is complete, delete the rcm_transform2.yaml file.

7 If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

2024.05 (May 2024)

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 2024.05 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
```

Crunchy Data seccompProfile Patch

Note: This deployment note is repeated from version 2023.09 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.

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' | xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec": {"template": {"spec":{"containers":[{"name": "database", "securityContext": {"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

SAS Allowance for Credit Loss

Before Deployment Commands

Revised acl_transform.yaml File

The acl_transform.yaml file for SAS Allowance for Credit Loss has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the acl_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Allowance for Credit Loss, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "acl_transform.yaml". If that phrase is present, then you have SAS Allowance for Credit Loss in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-acl/resources/acl_transform.yaml`** and change the file name to `acl_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-acl/resources/acl_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-acl/resources/acl_transform.yaml`**. You should now have an acl_transform.yaml file and an acl_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Allowance for Credit Loss README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the acl_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-acl/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_allowance_for_credit_loss.htm`** (for HTML format).
5. Use the acl_transform2.yaml file as a guide for filling in the rest of the values in the acl_transform.yaml file.
6. When the acl_transform.yaml file is complete, delete the acl_transform2.yaml file.

7 If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Anti-Money Laundering

After Deployment Commands

New Functionality for SAS Anti-Money Laundering

The latest update for SAS Anti-Money Laundering contains new functionality in several areas. After you have completed the update, refer to the *SAS Anti-Money Laundering: Administrator's Guide* for descriptions of the new areas of functionality and how to configure them:

- Chapter 2, Provision and Configure.
- Chapter 4, Databases.
- Chapter 5, Roles and Capabilities for Users and Groups.

SAS Asset and Liability Management

Before Deployment Commands

Revised alm_transform.yaml File

The alm_transform.yaml file for SAS Asset and Liability Management has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the alm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1 To determine if your deployment includes SAS Asset and Liability Management, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "alm_transform.yaml". If that phrase is present, then you have SAS Asset and Liability Management in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.

2 Go to **$deploy/site-config/sas-risk-cirrus-alm/resources/alm_transform.yaml** and change the file name to `alm_transform2.yaml`.

3 Copy the **$deploy/sas-bases/examples/sas-risk-cirrus-alm/resources/alm_transform.yaml** file to **$deploy/site-config/sas-risk-cirrus-alm/resources/alm_transform.yaml**. You should now have an alm_transform.yaml file and an alm_transform2.yaml file in the same directory.

4 Follow the steps in the SAS Asset and Liability Management README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the alm_transform.yaml file. The README file is located at **$deploy/sas-bases/examples/sas-risk-cirrus-alm/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_connection_settings_for_sas_asset_and_liability_management.htm** (for HTML format).

5 Use the alm_transform2.yaml file as a guide for filling in the rest of the values in the alm_transform.yaml file.

6 When the alm_transform.yaml file is complete, delete the alm_transform2.yaml file.

7 If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Dynamic Actuarial Modeling

Before Deployment Commands

Revised pcpricing_transform.yaml File

The pcpricing_transform.yaml file for SAS Dynamic Actuarial Modeling has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the pcpricing_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1 To determine if your deployment includes SAS Dynamic Actuarial Modeling, go to the base kustomization.yaml file (**$deploy/kustomization.yaml**). In the transformers block, search for "pcpricing_transform.yaml". If that phrase is present, then you have SAS Dynamic Actuarial Modeling in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.

2 Go to **$deploy/site-config/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml** and change the file name to `pcpricing_transform2.yaml`.

3 Copy the **$deploy/sas-bases/examples/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml** file to **$deploy/site-config/sas-risk-cirrus-pcpricing/resources/pcpricing_transform.yaml**. You should now have a pcpricing_transform.yaml file and a pcpricing_transform2.yaml file in the same directory.

4 Follow the steps in the SAS Dynamic Actuarial Modeling README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the pcpricing_transform.yaml file. The README file is located at **$deploy/sas-bases/examples/sas-risk-cirrus-pcpricing/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_sas_dynamic_actuarial_modeling_for_deployment.htm** (for HTML format).

5 Use the pcpricing_transform2.yaml file as a guide for filling in the rest of the values in the pcpricing_transform.yaml file.

6 When the pcpricing_transform.yaml file is complete, delete the pcpricing_transform2.yaml file.

7 If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Event Stream Processing

Before Deployment Commands

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to updaate your software, be sure to perform the tasks described in "Restore Projects After Update".

Delete espserver before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources", skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS Expected Credit Loss

Before Deployment Commands

Revised ecl_transform.yaml File

The ecl_transform.yaml file for SAS Expected Credit Loss has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the ecl_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Expected Credit Loss, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "ecl_transform.yaml". If that phrase is present, then you have SAS Expected Credit Loss in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-ecl/resources/ecl_transform.yaml`** and change the file name to `ecl_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-ecl/resources/ecl_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-ecl/resources/ecl_transform.yaml`**. You should now have an ecl_transform.yaml file and an ecl_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Expected Credit Loss README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the ecl_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-ecl/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_expected_credit_loss.htm`** (for HTML format).
5. Use the ecl_transform2.yaml file as a guide for filling in the rest of the values in the ecl_transform.yaml file.
6. When the ecl_transform.yaml file is complete, delete the ecl_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Insurance Capital Management

Before Deployment Commands

Revised icm_transform.yaml File

The icm_transform.yaml file for SAS Insurance Capital Management has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the icm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Insurance Capital Management, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "icm_transform.yaml". If that phrase is present, then you have SAS Insurance Capital Management in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-icm/resources/icm_transform.yaml`** and change the file name to `icm_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-icm/resources/icm_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-icm/resources/icm_transform.yaml`**. You should now have an icm_transform.yaml file and an icm_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Insurance Capital Management README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the icm_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-icm/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_insurance_capital_management.htm`** (for HTML format).
5. Use the icm_transform2.yaml file as a guide for filling in the rest of the values in the icm_transform.yaml file.
6. When the icm_transform.yaml file is complete, delete the icm_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Model Risk Management

Before Deployment Commands

Revised mrm_transform.yaml File

The mrm_transform.yaml file for SAS Model Risk Management has been modified with a new variable for increased functionality with Git repositories, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`. Use the following steps to get the new version of the mrm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Model Risk Management, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "mrm_transform.yaml". If that phrase is present, then you have SAS Model Risk Management in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-mrm/resources/mrm_transform.yaml`** and change the file name to `mrm_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-mrm/resources/mrm_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-mrm/resources/mrm_transform.yaml`**. You should now have a mrm_transform.yaml file and a mrm_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Model Risk Management README file to add the value for the new variable, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`, to the mrm_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-mrm/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_model_risk_management.htm`** (for HTML format).
5. Use the mrm_transform2.yaml file as a guide for filling in the rest of the values in the mrm_transform.yaml file.
6. When the mrm_transform.yaml file is complete, delete the mrm_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Risk Cirrus Core

Before Deployment Commands

Revised core_transform.yaml File

The core_transform.yaml file for SAS Risk Cirrus Core has been modified with a new variable for increased functionality with Git repositories, `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the core_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Risk Cirrus Core, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "core_transform.yaml". If that phrase is present, then you have SAS Risk Cirrus Core in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. The reference in the base kustomization.yaml file is the location of the transformer file. Go to that location and change the file name to `core_transform2.yaml`. For example, if the file is located at **`$deploy/site-config/sas-risk-cirrus-rcc/resources/core_transform.yaml`**, change the file name to **`$deploy/site-config/sas-risk-cirrus-rcc/resources/core_transform2.yaml`**.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/core_transform.yaml`** file to the directory you used in step 2. You should now have a core_transform.yaml file and a core_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Risk Cirrus Core README file to add the values for the new variable, `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the core_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm`** (for HTML format).
5. Use the core_transform2.yaml file as a guide for filling in the rest of the values in the core_transform.yaml file.
6. When the core_transform.yaml file is complete, delete the core_transform2.yaml file.

SAS Risk Factor Manager

Before Deployment Commands

Revised rfm_transform.yaml File

The rfm_transform.yaml file for SAS Risk Factor Manager has been modified with a new variable for increased functionality with Git repositories, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`. Use the following steps to get the new version of the rfm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Risk Factor Manager, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "rfm_transform.yaml". If that phrase is present, then you have SAS Risk Factor Manager in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-rfm/resources/rfm_transform.yaml`** and change the file name to `rfm_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-rfm/resources/rfm_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-rfm/resources/rfm_transform.yaml`**. You should now have an rfm_transform.yaml file and an rfm_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Risk Factor Manager README file to add the value for the new variable, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`, to the rfm_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rfm/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_risk_factor_manager.htm`** (for HTML format).
5. Use the rfm_transform2.yaml file as a guide for filling in the rest of the values in the rfm_transform.yaml file.
6. When the rfm_transform.yaml file is complete, delete the rfm_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Risk Modeling

Before Deployment Commands

Revised rm_transform.yaml File

The rm_transform.yaml file for SAS Risk Modeling has been modified with a new variable for increased functionality with Git repositories, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`. Use the following steps to get the new version of the rm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Risk Modeling, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "rm_transform.yaml". If that phrase is present, then you have SAS Risk Modeling in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-rm/resources/rm_transform.yaml`** and change the file name to `rm_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-rm/resources/rm_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-rm/resources/rm_transform.yaml`**. You should now have an rm_transform.yaml file and an rm_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Risk Modeling README file to add the value for the new variable, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE`, to the rm_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-rm/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_risk_modeling.htm`** (for HTML format).
5. Use the rm_transform2.yaml file as a guide for filling in the rest of the values in the rm_transform.yaml file.
6. When the rm_transform.yaml file is complete, delete the rm_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

SAS Stress Testing

Before Deployment Commands

Revised st_transform.yaml File

The st_transform.yaml file for SAS Stress Testing has been modified with new variables for increased functionality with Git repositories. The new fields are `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`. Use the following steps to get the new version of the rm_transform.yaml file in order to add this new information while maintaining the rest of the content in that file.

1. To determine if your deployment includes SAS Stress Testing, go to the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block, search for "st_transform.yaml". If that phrase is present, then you have SAS Stress Testing in your deployment and should continue with these steps. If the phrase is not present, skip the rest of this note.
2. Go to **`$deploy/site-config/sas-risk-cirrus-st/resources/st_transform.yaml`** and change the file name to `st_transform2.yaml`.
3. Copy the **`$deploy/sas-bases/examples/sas-risk-cirrus-st/resources/st_transform.yaml`** file to **`$deploy/site-config/sas-risk-cirrus-st/resources/st_transform.yaml`**. You should now have an st_transform.yaml file and an st_transform2.yaml file in the same directory.
4. Follow the steps in the SAS Stress Testing README file to add the values for the new variables, `SAS_RISK_CIRRUS_SOLUTION_GIT_REPO_TYPE` and `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_TYPE`, to the st_transform.yaml file. The README file is located at **`$deploy/sas-bases/examples/sas-risk-cirrus-st/resources/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_connection_settings_for_sas_stress_testing.htm`** (for HTML format).
5. Use the st_transform2.yaml file as a guide for filling in the rest of the values in the st_transform.yaml file.
6. When the st_transform.yaml file is complete, delete the st_transform2.yaml file.
7. If you have not already, complete the steps in "Revised core_transform.yaml File" on page 332.

2024.04 (April 2024)

General Deployment

Before Deployment Commands

Job Definition Rule Change Affects Group-Managed Service Account Functionality

Note: This deployment note is repeated from version 2023.12 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.

A change in the rules for job definitions prevent job requests with an associated job definition from being run by users other than the job creator or a SAS Administrator. Therefore group-managed service account functionality requires that job definitions are created in such a way that they can be shared with the group-managed service account, such as being created in a folder and then shared. Alternatively, you can create a rule that grants permission to a specific job definition, to all job definitions, or to a collection of job definitions. Perform one of the following set of steps depending on the restrictions you want to place on your service account.

Have a service account able to run any job

1 As a SAS Administrator, go to SAS Environment Manager → Rules.

2 Create a new rule with the following settings:

- Object URI: `/jobDefinitions/definitions/*`
- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

Have a service account able to run a specific job

1 Go to SAS Environment Manager → Jobs and Flows

2 Select the **Scheduling** tab.

3 Right-click the appropriate job and choose **Properties**.

4 Under the **Job Definition** tab, determine if the ID field is populated. If so, copy the ID. If the field is not populated, then the job definition is part of the job request object and no further steps are necessary.

5 As a SAS Administrator, go to SAS Environment Manager → Rules.

6 Create a rule with the following settings:

- Object URI: `/jobDefinitions/definitions/`*`ID-from-step-4`*
- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

7 Repeat Step 3 through Step 6 for each job you want to run under the service account.

Have the service account able to run jobs with definitions stored in SAS Content

Perform one of the following tasks:

1 Use SAS Drive to "share" the job with the service account.

2 Ensure the service account has Read (convey) permission on the folder where the job definition is stored.

SAS Configurator for Open Source Requires SCC for OpenShift

If you are updating a deployment on Red Hat OpenShift, SAS Configurator for Open Source now requires that you apply and bind a Security Context Constraint (SCC). For details, see "sas-pyconfig" in *SAS Viya Platform: Deployment Guide*.

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 2024.04 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
```

Crunchy Data seccompProfile Patch

Note: This deployment note is repeated from version 2023.09 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.

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' | xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec": {"template": {"spec":{"containers":[{"name": "database", "securityContext": {"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

SAS Event Stream Processing

Before Deployment Commands

Save espserver Resources

Before you perform an update, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to updaate your software, be sure to perform the tasks described in "Restore Projects After Update" on page 340.

Delete espserver before the Update

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

Before you update, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources" on page 339, skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

2024.03 (March 2024)

General Deployment

Before Deployment Commands

Job Definition Rule Change Affects Group-Managed Service Account Functionality

Note: This deployment note is repeated from version 2023.12 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.

Changes in the rules for job definitions prevent job requests with an associated job definition from being run by users other than the job creator or a SAS Administrator. Therefore group-managed service account functionality requires that job definitions are created in such a way that they can be shared with the group-managed service account, such as being created in a folder and then shared. Alternatively, you can create a rule that grants permission to a specific job definition, to all job definitions, or to a collection of job definitions. Perform one of the following set of steps depending on the restrictions you want to place on your service account.

Have a service account able to run any job

1 As a SAS Administrator, go to SAS Environment Manager → Rules.

2 Create a new rule with the following settings:

- Object URI: `/jobDefinitions/definitions/*`
- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

Have a service account able to run a specific job

1 Go to SAS Environment Manager → Jobs and Flows

2 Select the **Scheduling** tab.

3 Right-click the appropriate job and choose **Properties**.

4 Under the **Job Definition** tab, determine if the ID field is populated. If so, copy the ID. If the field is not populated, then the job definition is part of the job request object and no further steps are necessary.

5 As a SAS Administrator, go to SAS Environment Manager → Rules.

6 Create a rule with the following settings:

- Object URI: `/jobDefinitions/definitions/`*`ID-from-step-4`*
- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

7 Repeat Step 3 through Step 6 for each job you want to run under the service account.

Have the service account able to run jobs with definitions stored in SAS Content

Perform one of the following tasks:

1 Use SAS Drive to "share" the job with the service account.

2 Ensure the service account has Read (convey) permission on the folder where the job definition is stored.

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 2024.03 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
```

Crunchy Data seccompProfile Patch

Note: This deployment note is repeated from version 2023.09 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.

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' | xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec": {"template": {"spec":{"containers":[{"name": "database", "securityContext": {"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

OpenSearch

Before Deployment Commands

OpenSearch Adds Configurable Parameters for Retaining Audit Log Indices

Note: This topic applies to internal instances of OpenSearch only.

OpenSearch is changing its audit log indices retention policy so that the indices are no longer retained indefinitely. Instead, the audit log indices are retained only for seven days by default. If you want to use a different retention period, OpenSearch has added a new transformer file, audit-log-retention-transformer.yaml, to configure the retention of audit log indices. For details about using this new file, see the README file at **`$deploy/sas-bases/examples/configure-elasticsearch/internal/security-audit-logs/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/opensearch_security_audit_logs.htm`** (for HTML format).

When the software is updated for the first time after the introduction of this new policy, any audit log indices that are older than the defined retention period are deleted.

SAS Allowance for Credit Loss

Before Deployment Commands

Support for Apache Airflow

SAS Allowance for Credit Loss now supports using Apache Airflow for Process Orchestration with an instance of PostgreSQL. For details about configuring Apache Airflow, see the README located at **`$deploy/sas-bases/examples/sas-airflow/metadata/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm`** (for HTML format).

SAS Climate Risk Stress Testing

Before Deployment Commands

Support for Apache Airflow

SAS Climate Risk Stress Testing now supports using Apache Airflow for Process Orchestration with an instance of PostgreSQL. For details about configuring Apache Airflow, see the README located at **`$deploy/sas-bases/examples/sas-airflow/metadata/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm`** (for HTML format).

SAS Credit Risk Stress Testing

Before Deployment Commands

Support for Apache Airflow

SAS Credit Risk Stress Testing now supports using Apache Airflow for Process Orchestration with an instance of PostgreSQL. For details about configuring Apache Airflow, see the README located at **`$deploy/sas-bases/examples/sas-airflow/metadata/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm`** (for HTML format).

SAS Event Stream Processing

Before Deployment Commands

Required Transformer for SAS Event Stream Manager

A transformer for SAS Event Stream Manager that enables direct file access to the persistent volume file system is required. To apply the transformer, modify the transformers block of the base kustomization.yaml ($deploy/kustomization.yaml) by adding the following highlighted line:

```
transformers
- sas-bases/overlays/sas-event-stream-manager-app/data/data-transformer.yaml
...
```

For more information, see the README file located at **$deploy/sas-bases/overlays/sas-event-stream-manager-app/data/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configuring_file_storage_for_sas_event_stream_manager_projects.htm** (for HTML format).

Required Transformer for SAS Event Stream Processing Studio

A transformer for SAS Event Stream Processing Studio that enables direct file access to the persistent volume file system is required. To apply the transformer, modify the transformers block of the base kustomization.yaml file ($deploy/kustomization.yaml) by adding the following highlighted line:

```
transformers
- sas-bases/overlays/sas-event-stream-processing-studio-app/data/data-transformer.yaml
...
```

For more information, see the README file located at **$deploy/sas-bases/overlays/sas-event-stream-processing-studio-app/data/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configuring_file_storage_for_sas_event_stream_processing_studio_projects.htm** (for HTML format).

Save espserver Resources

Before you update to version 2024.03, save any running SAS Event Stream Processing projects to a local backup file.

1 Check for any existing SAS Event Stream Processing projects:

```
kubectl get espserver -l 'esp.iot.sas.com/server-type!=client-configuration'
-n name-of-namespace
```

If the command returns the following output, no further steps in this note need to be taken.

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

If the command returns other output, continue with Step 2.

2 Set an environment variable for your namespace:

```
export K8S_NAMESPACE_SOURCE='name-of-namespace'
```

3 Set an environment variable for the path to the backup file for your ESPServer custom resource:

```
export ESP_SERVERS_BACKUP='path-to-your-espservers-backup.json-file'
```

4 Create a backup file for your ESPServer custom resource:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" get espservers.iot.sas.com -l
'esp.iot.sas.com/server-type!=client-configuration' --show-managed-
fields=false -o json | jq 'del(.items[].access)|
del(.items[].metadata.annotations["kubectl.kubernetes.io/last-applied-
configuration"])|del(.items[].metadata.creationTimestamp)|
del(.items[].metadata.generation)|del(.items[].metadata.ownerReferences)|
del(.items[].metadata.resourceVersion)|del(.items[].metadata.uid)' | kubectl
-n "${K8S_NAMESPACE_SOURCE}" create -f - --dry-run=client -o json | jq --
slurp 'del(.[].metadata.namespace)|.[]' >"${ESP_SERVERS_BACKUP}"
```

5 After you perform the deployment commands to updaate your software, be sure to perform the tasks described in “Restore Projects After Update” on page 348.

Delete espserver before the Update

When you update to version 2024.03, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2024.03, 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.
```

After Deployment Commands

Restore Projects After Update

Note: If you did not back up any SAS Event Stream Processing resources by following the steps in "Save espserver Resources" on page 346, skip this deployment note.

Restore the ESPServer custom resource from the backup file to your Kubernetes namespace:

```
kubectl -n "${K8S_NAMESPACE_SOURCE}" apply -f "${ESP_SERVERS_BACKUP}"
```

SAS Expected Credit Loss

Before Deployment Commands

Support for Apache Airflow

SAS Expected Credit Loss now supports using Apache Airflow for Process Orchestration with an instance of PostgreSQL. For details about configuring Apache Airflow, see the README located at **$deploy/sas-bases/examples/sas-airflow/metadata/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm** (for HTML format).

SAS Insurance Capital Management

Before Deployment Commands

Support for Apache Airflow

SAS Insurance Capital Management now supports using Apache Airflow for Process Orchestration for both internal and external instances of PostgreSQL.

If your current deployment uses Process Orchestration for an external instance of PostgreSQL, you must perform the following steps:

1 Run the following commands to delete resources that are being replaced with updated versions:

```
kubectl delete deployment sas-airflow-scheduler -n name-of-namespace
kubectl delete deployment sas-airflow-triggerer -n name-of-namespace
kubectl delete statefulset sas-airflow-worker -n name-of-namespace
kubectl delete job sas-airflow-create-nonadmin-user -n name-of-namespace
kubectl delete job sas-airflow-run-airflow-migrations -n name-of-namespace
```

2 Add the following line to top of the components block base kustomization.yaml file (**$deploy/kustomization.yaml**):

```
components:
- sas-bases/components/sas-airflow/external-airflow
```

Changing Process Orchestration from an external instance of PostgreSQL to an internal instance is not supported.

For details about Process Orchestration and Apache Airflow, see the README located at **$deploy/sas-bases/examples/sas-airflow/metadata/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm** (for HTML format).

SAS Insurance Contract Valuation Foundation

Before Deployment Commands

Support for Apache Airflow

SAS Insurance Contract Valuation Foundation now supports using Apache Airflow for Process Orchestration for both internal and external instances of PostgreSQL.

If your current deployment uses Process Orchestration for an external instance of PostgreSQL, you must perform the following steps:

1 Run the following commands to delete resources that are being replaced with updated versions:

```
kubectl delete deployment sas-airflow-scheduler -n name-of-namespace
kubectl delete deployment sas-airflow-triggerer -n name-of-namespace
kubectl delete statefulset sas-airflow-worker -n name-of-namespace
kubectl delete job sas-airflow-create-nonadmin-user -n name-of-namespace
kubectl delete job sas-airflow-run-airflow-migrations -n name-of-namespace
```

2 Add the following line to top of the components block base kustomization.yaml file (**$deploy/kustomization.yaml**):

```
components:
- sas-bases/components/sas-airflow/external-airflow
```

Changing Process Orchestration from an external instance of PostgreSQL to an internal instance is not supported.

For details about Process Orchestration and Apache Airflow, see the README located at **$deploy/sas-bases/examples/sas-airflow/metadata/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm** (for HTML format).

SAS Regulatory Capital Management

Before Deployment Commands

Support for Apache Airflow

SAS Regulatory Capital Management now supports using Apache Airflow for Process Orchestration for both internal and external instances of PostgreSQL.

If your current deployment uses Process Orchestration for an external instance of PostgreSQL, you must perform the following steps:

1 Run the following commands to delete resources that are being replaced with updated versions:

```
kubectl delete deployment sas-airflow-scheduler -n name-of-namespace
kubectl delete deployment sas-airflow-triggerer -n name-of-namespace
kubectl delete statefulset sas-airflow-worker -n name-of-namespace
kubectl delete job sas-airflow-create-nonadmin-user -n name-of-namespace
kubectl delete job sas-airflow-run-airflow-migrations -n name-of-namespace
```

2 Add the following line to top of the components block base kustomization.yaml file (**$deploy/kustomization.yaml**):

```
components:
- sas-bases/components/sas-airflow/external-airflow
```

Changing Process Orchestration from an external instance of PostgreSQL to an internal instance is not supported.

For details about Process Orchestration and Apache Airflow, see the README located at **$deploy/sas-bases/examples/sas-airflow/metadata/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/**

configure_apache_airflow_for_process_orchestration.htm (for HTML format).

SAS Stress Testing

Before Deployment Commands

Support for Apache Airflow

SAS Stress Testing now supports using Apache Airflow for Process Orchestration with an instance of PostgreSQL. For details about configuring Apache Airflow, see the README located at **$deploy/sas-bases/examples/sas-airflow/metadata/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/configure_apache_airflow_for_process_orchestration.htm** (for HTML format).

2024.02 (February 2024)

General Deployment

Before Deployment Commands

Job Definition Rule Change Affects Group-Managed Service Account Functionality

Note: This deployment note is repeated from version 2023.12 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.

Changes in the rules for job definitions prevent job requests with an associated job definition from being run by users other than the job creator or a SAS Administrator. Therefore group-managed service account functionality requires that job definitions are created in such a way that they can be shared with the group-managed service account, such as being created in a folder and then shared.

Alternatively, you can create a rule that grants permission to a specific job definition, to all job definitions, or to a collection of job definitions. Perform one of the following set of steps depending on the restrictions you want to place on your service account.

Have a service account able to run any job

1 As a SAS Administrator, go to SAS Environment Manager → Rules.

2 Create a new rule with the following settings:

- Object URI: `/jobDefinitions/definitions/*`
- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

Have a service account able to run a specific job

1 Go to SAS Environment Manager → Jobs and Flows

2 Select the **Scheduling** tab.

3 Right-click the appropriate job and choose **Properties**.

4 Under the **Job Definition** tab, determine if the ID field is populated. If so, copy the ID. If the field is not populated, then the job definition is part of the job request object and no further steps are necessary.

5 As a SAS Administrator, go to SAS Environment Manager → Rules.

6 Create a rule with the following settings:

- Object URI: `/jobDefinitions/definitions/`*`ID-from-step-4`*
- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

7 Repeat Step 3 through Step 6 for each job you want to run under the service account.

Have the service account able to run jobs with definitions stored in SAS Content

Perform one of the following tasks:

1 Use SAS Drive to "share" the job with the service account.

2 Ensure the service account has Read (convey) permission on the folder where the job definition is stored.

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 2024.02 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
```

Crunchy Data seccompProfile Patch

Note: This deployment note is repeated from version 2023.09 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.

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' |
xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec":
{"template": {"spec":{"containers":[{"name": "database", "securityContext":
{"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

Analytics Resources

Before Deployment Commands

Scale Down Pods Before Update

Note: If you are updating your software with the SAS Viya Platform Deployment Operator or the sas-orchestration tool, do not perform the steps in this note. Proceed with this note only if you are updating with the Kubernetes commands deployment method.

A number of offerings use the sas-analytics-resources service. That service requires that pods be scaled down to zero before the update.

1 Determine if your deployment is affected by this change.

   a If your deployment contains sas-analytics-resources, it is affected by this change. The change could also affect any sas-analytics-services pods in your deployment. Perform the following commands to determine if either type of pod is in your deployment:

   ```
   kubectl get pods -n name-of-namespace | grep sas-analytics-resources
   ```

   ```
   kubectl get pods -n name-of-namespace | grep sas-analytics-services
   ```

If the sas-analytics-resources command has no output, you do not need to perform the tasks described in this note. Otherwise, continue with Step 1b.

b To determine the version of your sas-analytics-resources pods, perform the following command:

```
kubectl get pods --no-headers --output=jsonpath='{range.items[*]}
{.metadata.labels.app} {":"} {.metadata.annotations.sas\.com/version}
{"\n"} {end} ' --selector 'app in (sas-analytics-resources)' -n name-of-
namespace
```

The output of this command takes this format:

```
sas-analytics-resources:x.y.z
```

The version number for the sas-analytics-resources pods is represented by `x.y.z`, which is three integers separated by decimal points. If the integer `x` is 22 or greater, you do not need to complete the tasks in this note. If `x` is less than 22, continue with Step 2.

2 For each type of pod returned in Step 1a, determine if Horizontal Pod Autoscaling (HPA) is being used.

a Perform the following commands, depending on which type of pod was found to be in your deployment in Step 1a on page 354:

```
kubectl get hpa --field-selector metadata.name=sas-analytics-resources -n
name-of-namespace
```

```
kubectl get hpa --field-selector metadata.name=sas-analytics-services -n
name-of-namespace
```

b The output looks like this:

```
NAME                     REFERENCE                           TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
sas-analytics-resources  Deployment/sas-analytics-resources  2%/80%    2         2         2          25h
```

Note the values for `MINPODS`, `MAXPODS`, and `REPLICAS` for each type of pod in your deployment.

- If the values for `MINPODS`, `MAXPODS`, or `REPLICAS` are greater than 1, HPA has been turned on for that type of pod and must be turned off. Use the following command:

  ```
  kubectl patch hpa pod-type --patch '{"spec": {"minReplicas": 1,
  "maxReplicas": 1}}' -n name-of-namespace
  ```

  Replace *pod-type* with `sas-analytics-resources` or `sas-analytics-services`, depending on which command produced the output. If both types of pod produce this output, you must run the command twice, once for each type.

- If the values for `MINPODS`, `MAXPODS`, and `REPLICAS` are each one, then HPA has not been turned on.

3 Scale the sas-analytics-resources pods down.

```
kubectl -n name-of-namespace scale deployment/sas-analytics-resources --
replicas=0
```

If you have sas-analytics-services pods, scale them down also.

```
kubectl -n name-of-namespace scale deployment/sas-analytics-services --
replicas=0
```

The sas-analytics-services service will be replaced by the sas-analytics-gateway service during the update. After the update, you will scale up the sas-analytics-resources service when you perform the task described in "Scale Up Pods After Update".

After Deployment Commands

Scale Up Pods After Update

Note: If you did not perform any of the steps in "Scale Down Pods Before Update" on page 354, do not perform the procedure in this note.

After the update is completed, you must return the sas-analytics pods to their original state.

1 For each type of pod that had Horizontal Pod Autoscaling (HPA) turned on, perform the following command to turn HPA back on:

```
kubectl patch hpa pod-type --patch '{"spec": {"minReplicas": previous-value-
for-MINPODS, "maxReplicas": previous-value-for-MAXPODS}}' -n name-of-
namespace
```

For *pod-type*, use either `sas-analytics-resources` or `sas-analytics-gateway`, in place of sas-analytics-services pods. For *previous-value-for-MINPODS*, use the value for `MINPODS` that you noted in Step 2b of "Scale Down Pods Before Update". For *previous-value-for-MAXPODS*, use the value for `MAXPODS` that you noted in that same step.

2 Using the value you noted in Step 2b of "Scale Down Pods Before Update", scale the sas-analytics-resources service up to the number of replicas you had before the update.

```
kubectl -n name-of-namespace scale deployment/sas-analytics-resources --
replicas=number-of-replicas-before-update
```

If your deployment had sas-analytics-services pods before the update, scale the sas-analytics-gateway service up to the number of sas-analytics-service replicas you had before the update.

```
kubectl -n name-of-namespace scale deployment/sas-analytics-gateway --
replicas=number-of-sas-analytics-services-replicas-before-update
```

3 To confirm that the restarted pods are running, perform the following commands:

```
kubectl get pods -n name-of-namespace | grep sas-analytics-resources
```

```
kubectl get pods -n name-of-namespace | grep sas-analytics-gateway
```

You will get non-zero output for each command if that type of pod is running.

SAS Anti-Money Laundering

After Deployment Commands

Remove Row-Level Bypass Privileges

SAS Anti-Money Laundering removes row-level bypass privileges during the update. After the update is complete, you should consider modifying your permissions to match the updated configuration. For detaills, see Updating from a Previous Cadence on the SAS Viya Platform

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

IMPORTANT This Important note applies to either of the following scenarios:

- your update consists of moving from the Long-Term Support cadence to the Stable cadence and includes the Stable 2024.03 version
- you are updating across more than one version and the update includes the Stable 2024.03 version

If either of those scenarios describes your update, you must perform the tasks described in "Save espserver Resources" on page 346 before performing the tasks in this deployment note.

When you update to version 2024.02, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2024.02, 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.
```

SAS/ACCESS Interface to MongoDB

Before Deployment Commands

Changes to Client Requirements

Because OpenSSL 1.0.2 libraries no longer exist in the images that SAS ships, the SAS Viya platform no longer supports executing versions of MongoDB that require OpenSSL 1.0.2. Therefore, ensure that your version of MongoDB supports a version of OpenSSL that is later than version 1.0.2.

SAS/ACCESS Interface to MySQL

Before Deployment Commands

Changes to Client Requirements

Because OpenSSL 1.0.2 libraries no longer exist in the images that SAS ships, the SAS Viya platform no longer supports executing versions of MySQL that require OpenSSL 1.0.2. Therefore, ensure that your version of MySQL supports a version of OpenSSL that is later than version 1.0.2.

2024.01 (January 2024)

General Deployment

Before Deployment Commands

Job Definition Rule Change Affects Group-Managed Service Account Functionality

Note: This deployment note is repeated from version 2023.12 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.

Changes in the rules for job definitions prevent job requests with an associated job definition from being run by users other than the job creator or a SAS Administrator. Therefore group-managed service account functionality requires that job definitions are created in such a way that they can be shared with the group-managed service account, such as being created in a folder and then shared. Alternatively, you can create a rule that grants permission to a specific job definition, to all job definitions, or to a collection of job definitions. Perform one of the following set of steps depending on the restrictions you want to place on your service account.

Have a service account able to run any job

1 As a SAS Administrator, go to SAS Environment Manager → Rules.

2 Create a new rule with the following settings:

- Object URI: `/jobDefinitions/definitions/*`
- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

Have a service account able to run a specific job

1 Go to SAS Environment Manager → Jobs and Flows

2 Select the **Scheduling** tab.

3 Right-click the appropriate job and choose **Properties**.

4 Under the **Job Definition** tab, determine if the ID field is populated. If so, copy the ID. If the field is not populated, then the job definition is part of the job request object and no further steps are necessary.

5 As a SAS Administrator, go to SAS Environment Manager → Rules.

6 Create a rule with the following settings:

- Object URI: `/jobDefinitions/definitions/`*`ID-from-step-4`*
- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

7 Repeat Step 3 through Step 6 for each job you want to run under the service account.

Have the service account able to run jobs with definitions stored in SAS Content

Perform one of the following tasks:

1 Use SAS Drive to "share" the job with the service account.

2 Ensure the service account has Read (convey) permission on the folder where the job definition is stored.

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 2024.01 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
```

Crunchy Data seccompProfile Patch

Note: This deployment note is repeated from version 2023.09 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.

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' | xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec": {"template": {"spec":{"containers":[{"name": "database", "securityContext": {"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

IMPORTANT This Important note applies to either of the following scenarios:

- your update consists of moving from the Long-Term Support cadence to the Stable cadence and includes the Stable 2024.03 version
- you are updating across more than one version and the update includes the Stable 2024.03 version

If either of those scenarios describes your update, you must perform the tasks described in "Save espserver Resources" on page 346 before performing the tasks in this deployment note.

When you update to version 2024.01, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2024.01, 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.
```

2023.12 (December 2023)

General Deployment

Before Deployment Commands

Job Definition Rule Change Affects Group-Managed Service Account Functionality

Changes in the rules for job definitions prevent job requests with an associated job definition from being run by users other than the job creator or a SAS Administrator. Therefore group-managed service account functionality requires that job definitions are created in such a way that they can be shared with the group-managed service account, such as being created in a folder and then shared. Alternatively, you can create a rule that grants permission to a specific job definition, to all job definitions, or to a collection of job definitions. Perform one of the following set of steps depending on the restrictions you want to place on your service account.

Have a service account able to run any job

1. As a SAS Administrator, go to SAS Environment Manager → Rules.
2. Create a new rule with the following settings:
   - Object URI: `/jobDefinitions/definitions/*`
   - Rule Type: `Grant`
   - Permission: `Read`
   - Principal Type: `User`
   - Principal: *`name-of-service-account`*

Have a service account able to run a specific job

1. Go to SAS Environment Manager → Jobs and Flows
2. Select the **Scheduling** tab.
3. Right-click the appropriate job and choose **Properties**.
4. Under the **Job Definition** tab, determine if the ID field is populated. If so, copy the ID. If the field is not populated, then the job definition is part of the job request object and no further steps are necessary.

5 As a SAS Administrator, go to SAS Environment Manager → Rules.

6 Create a rule with the following settings:

- Object URI: `/jobDefinitions/definitions/ID-from-step-4`
- Rule Type: `Grant`
- Permission: `Read`
- Principal Type: `User`
- Principal: *`name-of-service-account`*

7 Repeat Step 3 through Step 6 for each job you want to run under the service account.

Have the service account able to run jobs with definitions stored in SAS Content

Perform one of the following tasks:

1 Use SAS Drive to "share" the job with the service account.

2 Ensure the service account has Read (convey) permission on the folder where the job definition is stored.

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 2023.12 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
```

Crunchy Data seccompProfile Patch

Note: This deployment note is repeated from version 2023.09 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.

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' | xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec": {"template": {"spec":{"containers":[{"name": "database", "securityContext": {"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

> **IMPORTANT** This Important note applies to either of the following scenarios:
>
> - your update consists of moving from the Long-Term Support cadence to the Stable cadence and includes the Stable 2024.03 version
> - you are updating across more than one version and the update includes the Stable 2024.03 version
>
> If either of those scenarios describes your update, you must perform the tasks described in "Save espserver Resources" on page 346 before performing the tasks in this deployment note.

When you update to version 2023.12, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.12, 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 Viya with SingleStore

Before Deployment Commands

Change to SingleStore Cluster Definition

With version 2023.12 support for Kustomize moved to v5.1.1. This change requires revision to the base kustomization.yaml file (**$deploy/kustomization.yaml**) .

1 If you are unsure whether your deployment includes SAS Viya with SingleStore, search the base kustomization.yaml file for the string `singlestore`. If your search finds the string, continue with the steps in this deployment note. If your search does not find the string, then your deployment does not include SAS Viya with SingleStore and you should skip the rest of this note.

2 Remove the following line from your base kustomization.yaml file:

```
...
generators:
- site-config/sas-singlestore/sas-singlestore-secret.yaml
...
```

3 Also in the base kustomization.yaml file, add the following highlighted line to the components block:

```
...
components:
- sas-bases/components/sas-singlestore
- site-config/sas-singlestore
...
```

Ensure that the `- sas-bases/components/sas-singlestore` entry comes before any TLS components. Also ensure that the `- site-config/sas-singlestore` entry is after the `- sas-bases/components/sas-singlestore` entry. The `- site-config/sas-singlestore` entry merges the license/secret into the deployment when the update commands are performed.

The revised deployment step is located in Step 6 of the README file located at **$deploy/sas-bases/examples/sas-singlestore/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/sas_singlestore_cluster_operator.htm** (for HTML format).

2023.11 (November 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 2023.11 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
```

Delete Crunchy Data Operator Upgrade Deployment

Note: If your deployment does not use an internal instance of PostgreSQL, skip this note.

Note: This deployment note is repeated from version 2023.10 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.

Version 2023.10 contains an upgrade to the PostgreSQL operator from Crunchy Data. While the upgrade happens without user intervention, in some cases the obsolete pgo-upgrade deployment fails to remove itself from the SAS Viya platform deployment. To determine if the pgo-upgrade deployment is present after the SAS Viya platform update:

```
kubectl -n name-of-namespace get deployment sas-crunchy5-postgres-operator-upgrade
```

If the output of the command indicates that the pgo-upgrade deployment is present, delete it:

```
kubectl -n name-of-namespace delete deployment sas-crunchy5-postgres-operator-upgrade
```

Crunchy Data seccompProfile Patch

Note: This deployment note is repeated from version 2023.09 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.

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' | xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec": {"template": {"spec":{"containers":[{"name": "database", "securityContext": {"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

SAS Conversation Designer

Before Deployment Commands

Remove Transformer from kustomization yaml File

Note: This deployment note is repeated from version 2023.10 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.

SAS Conversation Designer has reached end of life. If you formerly had SAS Conversation Designer included in your SAS Viya platform deployment, it is no longer available to you after this update. Additionally, you must remove a reference to a SAS Conversation Designer transformer from your base kustomization.yaml file (**`$deploy/kustomization.yaml`**).

1. In the transformers block of the base kustomization.yaml file, search for a reference to `sas-bases/overlays/sas-natural-language-understanding/resources/disable-parser-transformer.yaml`. If the line is not present, no further actions are required.
2. Remove the `sas-bases/overlays/sas-natural-language-understanding/resources/disable-parser-transformer.yaml` line.
3. Save and close the base kustomization.yaml file.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

IMPORTANT This Important note applies to either of the following scenarios:

- your update consists of moving from the Long-Term Support cadence to the Stable cadence and includes the Stable 2024.03 version
- you are updating across more than one version and the update includes the Stable 2024.03 version

If either of those scenarios describes your update, you must perform the tasks described in "Save espserver Resources" on page 346 before performing the tasks in this deployment note.

When you update to version 2023.11, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.11, 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 Intelligent Planning

Before Deployment Commands

Configure qpStartupOptions for SAS Environment Manager

Note: This deployment note is repeated from version 2023.09 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.

SAS Environment Manager includes changes that require that the correct qpStartupOptions are in place before the update is performed.

1. From the applications menu (☰), click **Manage Environment**.
2. Click 🔧 to navigate to the Configuration workspace.
3. From the **View** list on the left, click **Definitions**.
4. In the **Filter** text box in the right pane, click **sas.planning.cas**.
5. Click the edit icon (✎).
6. In the Edit sas.planning.cas Configuration window, navigate to **qpStartupOptions** and verify that the **-XX:+UseParallelGC** option is present. If it is not present, add it.
7. Save your changes.

2023.10 (October 2023)

General Deployment

Before Deployment Commands

Updated Security Resource File

The openssl-generated-ingress-certificate.yaml file is used to generate the server identity certificate for the ingress controller and store it in a secret. Its use is optional in deployments of SAS Viya platform. However, its contents were revised in version 2023.10, so if you did use the file, it must be replaced with the new version.

1. In the base kustomization.yaml file (**`$deploy/kustomization.yaml`**), look in the resources block for a reference to the openssl-generated-ingress-certificate.yaml file. If there is no reference, you did not use the file and no further steps are necessary. If there is a reference, continue with step 2.
2. The reference in the base kustomization.yaml file is the location of the yaml file. Go to that location and change the file name to `openssl-generated-ingress-certificate2.yaml`. For example, if the file is located at **`site-config/security/openssl-generated-ingress-certificate.yaml`**, change the file name to **`site-config/security/openssl-generated-ingress-certificate2.yaml`**.
3. Copy **`$deploy/sas-bases/examples/security/openssl-generated-ingress-certificate.yaml`** to the directory in step 2. You should now have a openssl-generated-ingress-certificate.yaml file and a openssl-generated-ingress-certificate2.yaml file in the same directory.
4. The file includes variables that can be modified for your deployment. Using the openssl-generated-ingress-certificate2.yaml file as a guide, revise the openssl-generated-ingress-certificate.yaml file so that it includes any values you used in the earlier version of the file.
5. After your revisions are complete, delete the openssl-generated-ingress-certificate2.yaml file.
6. Delete the sas-create-openssl-ingress-certificate Job:

   ```
   kubectl delete job sas-create-openssl-ingress-certificate -n name-of-namespace
   ```

New File Required for OpenShift Container Registry

Note: If you are using a mirror registry other than OpenShift Container Registry, skip this note.

If you are deploying on Red Hat OpenShift using the OpenShift Container Registry, you should add a reference to a new file in the base kustomization.yaml file (**`$deploy/kustomization.yaml`**). In the transformers block in the base kustomization.yaml file, add a reference to the **`sas-bases/overlays/sas-prepull/change-relpath.yaml`** file. Here is an example:

```
transformers:
...
- sas-bases/overlays/sas-prepull/change-relpath.yaml
- sas-bases/overlays/required/transformers.yaml
```

SAS Workload 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 currently include SAS Workflow Orchestrator, after the update, the new deployment will include SAS Workload Orchestrator, and it will be enabled.

Note: For certain offerings SAS Workload Orchestrator has been included since release 2023.08. These offerings are:

- SAS Visual Statistics
- SAS Visual Machine Learning
- SAS Data Science
- SAS Data Science Decisioning
- SAS Text Analytics
- SAS Visual Forecasting
- SAS Intelligent Decisioning
- SAS Data Science Programming

For these offerings there is no change in release 2023.10.

IMPORTANT The SAS Workload Orchestrator requires that at least one node in the cluster contain the label `workload.sas.com/class=compute`. Not setting the label has critical repercussions. For example:

- Compute sessions and jobs cannot be launched.
- SAS Studio does not function.
- Batch and scheduled jobs do not run.

If your cluster does not support labels or taints, you must disable SAS Workload Orchestrator before you perform the commands that update your software.

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).

If you choose to keep SAS Workload Orchestrator enabled, its daemons require information about resources on the nodes that can be used to run jobs. In order to obtain accurate resource information, you must add a ClusterRole and a ClusterRoleBinding to the SAS Workload Orchestrator service account. For more information about the required ClusterRoles, see the README file located at **`$deploy/sas-bases/overlays/sas-workload-orchestrator/README.md`** (for Markdown format) or at **`$deploy/sas-bases/docs/cluster_privileges_for_sas_workload_orchestrator_service.htm`** (for HTML format).

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 2023.10 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
```

Delete Crunchy Data Operator Upgrade Deployment

Note: If your deployment does not use an internal instance of PostgreSQL, skip this note.

Version 2023.10 contains an upgrade to the PostgreSQL operator from Crunchy Data. While the upgrade happens without user intervention, in some cases the obsolete pgo-upgrade deployment fails to remove itself from the SAS Viya platform deployment. To determine if the pgo-upgrade deployment is present after the SAS Viya platform update:

```
kubectl -n name-of-namespace get deployment sas-crunchy5-postgres-operator-upgrade
```

If the output of the command indicates that the pgo-upgrade deployment is present, delete it:

```
kubectl -n name-of-namespace delete deployment sas-crunchy5-postgres-operator-upgrade
```

Crunchy Data seccompProfile Patch

Note: This deployment note is repeated from version 2023.09 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.

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' |
xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec":
{"template": {"spec":{"containers":[{"name": "database", "securityContext":
{"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

SAS Conversation Designer

Before Deployment Commands

Remove Transformer from kustomization yaml File

SAS Conversation Designer has reached end of life. If you formerly had SAS Conversation Designer included in your SAS Viya platform deployment, it is no longer available to you after this update. Additionally, you must remove a reference to a SAS Conversation Designer transformer from your base kustomization.yaml file (**`$deploy/kustomization.yaml`**).

1. In the transformers block of the base kustomization.yaml file, search for a reference to `sas-bases/overlays/sas-natural-language-understanding/resources/disable-parser-transformer.yaml`. If the line is not present, no further actions are required.
2. Remove the `sas-bases/overlays/sas-natural-language-understanding/resources/disable-parser-transformer.yaml` line.
3. Save and close the base kustomization.yaml file.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2023.10, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.10, 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.
```

SAS Intelligent Planning

Before Deployment Commands

Configure qpStartupOptions for SAS Environment Manager

Note: This deployment note is repeated from version 2023.09 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.

SAS Environment Manager includes changes that require that the correct qpStartupOptions are in place before the update is performed.

1. From the applications menu (☰), click **Manage Environment**.
2. Click 🔧 to navigate to the Configuration workspace.
3. From the **View** list on the left, click **Definitions**.
4. In the **Filter** text box in the right pane, click **sas.planning.cas**.

5 Click the edit icon (✎).

6 In the Edit sas.planning.cas Configuration window, navigate to **qpStartupOptions** and verify that the **-XX:+UseParallelGC** option is present. If it is not present, add it.

7 Save your changes.

SAS Risk Modeling

Before Deployment Commands

SAS Risk Modeling Requires a Git Repository

Note: This deployment note is repeated from version 2023.07 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.

Starting with version 2023.07, SAS Risk Modeling requires a Git repository. The requirements for the repository are described at "Requirements for SAS® Risk Modeling" in *System Requirements for the SAS Viya Platform*.

Additionally, parameters in the **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/core_transform.yaml** file should be revised to account for the addition of the Git repository. `SAS_RISK_CIRRUS_CODE_LIB_GIT_SETUP_NEEDED` should be set to `Y`. The following parameters should be set to the appropriate values for your Git repository:

- `SAS_RISK_CIRRUS_CODE_LIB_REPO_NAME`
- `SAS_RISK_CIRRUS_CODE_LIB_REPO_USER`
- `SAS_RISK_CIRRUS_CODE_LIB_REPO_EMAIL`
- `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_FULL_URL`

For the details about these parameters, see the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

Configuration for Workflow Service Tasks

Note: This deployment note is repeated from version 2023.09 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.

Starting with version 2023.09, SAS Risk Modeling requires that certain values in the SAS Risk Cirrus Core configuration file be set so that the offering can take advantage of the workflow service tasks. In the **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/core_transform.yaml** file, the following parameters must be set to the appropriate values:

- SAS_RISK_CIRRUS_SET_WORKFLOW_SERVICE_ACCOUNT_FLG
- SAS_RISK_CIRRUS_WORKFLOW_DEFAULT_SERVICE_ACCOUNT

For the details about these parameters, see the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

2023.09 (September 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 2023.09 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
```

Crunchy Data seccompProfile Patch

Note: If your deployment is running on Red Hat OpenShift, skip this note.

Note: If you are not using an internal instance of PostgreSQL in your deployment, skip this note.

The secure computing mode (seccomp) has been removed from Kubernetes at version 1.27. Deployments that upgrade to Kubernetes 1.27 or later and that use an internal instance of PostgreSQL should apply a patch in order to use the securityContext.seccompProfile with the internal instance of PostgreSQL. The securityContext.seccompProfile maintains functionality similar to what was available in earlier versions of Kubernetes.

Patch the Statefulsets associated with the internal PostgreSQL instances:

```
kubectl -n name-of-namespace get sts | grep postgres-00 | awk '{print $1}' | xargs -I {} kubectl -n name-of-namespace patch statefulset $1 -p '{"spec": {"template": {"spec":{"containers":[{"name": "database", "securityContext": {"seccompProfile": { "type": "RuntimeDefault"}}}]}}}}' {}
```

After the command is performed, Kubernetes may issue a warning that indicates that the annotation is deprecated and not in use. This warning may be safely ignored because you have applied the appropriate patch with the steps above.

For information about maintaining the patch through the life of your deployment, see "Crunchy Data seccomp Profile" in *SAS Viya Platform: Infrastructure Servers*.

SAS Event Stream Processing

Before Deployment Commands

Delete espserver before the Update

When you update to version 2023.09, the ESPServer custom resource must be deleted in order to pick up changes during the update.

Before you update to 2023.09, 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.
```

SAS Intelligent Planning

Before Deployment Commands

Configure qpStartupOptions for SAS Environment Manager

SAS Environment Manager includes changes that require that the correct qpStartupOptions are in place before the update is performed.

1. From the applications menu (☰), click **Manage Environment**.
2. Click 🔧 to navigate to the Configuration workspace.
3. From the **View** list on the left, click **Definitions**.
4. In the **Filter** text box in the right pane, click **sas.planning.cas**.
5. Click the edit icon (✎).
6. In the Edit sas.planning.cas Configuration window, navigate to **qpStartupOptions** and verify that the **-XX:+UseParallelGC** option is present. If it is not present, add it.

7 Save your changes.

SAS Risk Modeling

Before Deployment Commands

SAS Risk Modeling Requires a Git Repository

Note: This deployment note is repeated from version 2023.07 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.

Starting with version 2023.07, SAS Risk Modeling requires a Git repository. The requirements for the repository are described at "Requirements for SAS® Risk Modeling" in *System Requirements for the SAS Viya Platform*.

Additionally, parameters in the **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/core_transform.yaml** file should be revised to account for the addition of the Git repository. `SAS_RISK_CIRRUS_CODE_LIB_GIT_SETUP_NEEDED` should be set to `Y`. The following parameters should be set to the appropriate values for your Git repository:

- `SAS_RISK_CIRRUS_CODE_LIB_REPO_NAME`
- `SAS_RISK_CIRRUS_CODE_LIB_REPO_USER`
- `SAS_RISK_CIRRUS_CODE_LIB_REPO_EMAIL`
- `SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_FULL_URL`

For the details about these parameters, see the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

Configuration for Workflow Service Tasks

Starting with version 2023.09, SAS Risk Modeling requires that certain values in the SAS Risk Cirrus Core configuration file be set so that the offering can take advantage of the workflow service tasks. In the **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/core_transform.yaml** file, the following parameters must be set to the appropriate values:

- `SAS_RISK_CIRRUS_SET_WORKFLOW_SERVICE_ACCOUNT_FLG`
- `SAS_RISK_CIRRUS_WORKFLOW_DEFAULT_SERVICE_ACCOUNT`

For the details about these parameters, see the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-rcc/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

2023.08 (August 2023)

General Deployment

Before Deployment Commands

SAS Workload Orchestrator Included in Deployment by Default

SAS Workload Orchestrator is now deployed and enabled, by default, as part of many SAS Viya platform deployments. For those offerings for which it is not included by default, it is available upon request. A license is no longer required.

The following offerings include SAS Workload Orchestrator by default:

- SAS Visual Statistics
- SAS Visual Machine Learning
- SAS Data Science
- SAS Data Science Decisioning
- SAS Text Analytics
- SAS Visual Forecasting
- SAS Intelligent Decisioning
- SAS Data Science Programming

If you are updating a deployment that does not currently include SAS Workflow Orchestrator, after the update, the new deployment will include SAS Workflow Orchestrator, and it will be enabled.

IMPORTANT The SAS Workload Orchestrator requires that at least one node in the cluster contain the label `workload.sas.com/class=compute`. Not setting the label has critical repercussions. For example:

- Compute sessions and jobs cannot be launched.
- SAS Studio does not function.

- Batch and scheduled jobs do not run.

If your cluster does not support labels or taints, you must disable SAS Workload Orchestrator before you perform the commands that update your software.

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).

If you choose to keep SAS Workload Orchestrator enabled, its daemons require information about resources on the nodes that can be used to run jobs. In order to obtain accurate resource information, you must add a ClusterRole and a ClusterRoleBinding to the SAS Workload Orchestrator service account. For more information about the required ClusterRoles, see the README file located at **$deploy/sas-bases/overlays/sas-workload-orchestrator/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/cluster_privileges_for_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.
```

SAS Risk Modeling

Before Deployment Commands

SAS Risk Modeling Requires a Git Repository

Note: This deployment note is repeated from version 2023.07 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.

Starting with version 2023.07, SAS Risk Modeling requires a Git repository. The requirements for the repository are described at "Requirements for SAS® Risk Modeling" in *System Requirements for the SAS Viya Platform*.

Additionally, parameters in the **$deploy/sas-bases/examples/sas-risk-cirrus-core/resources/core_transform.yaml** file should be revised to account for the addition of the Git repository. `SAS_RISK_CIRRUS_CODE_LIB_GIT_SETUP_NEEDED` should be set to `Y`. The following parameters should be set to the appropriate values for your Git repository:

- SAS_RISK_CIRRUS_CODE_LIB_REPO_NAME
- SAS_RISK_CIRRUS_CODE_LIB_REPO_USER
- SAS_RISK_CIRRUS_CODE_LIB_REPO_EMAIL
- SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_FULL_URL

For the details about these parameters, see the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-core/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

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.
```

SAS Risk Modeling

Before Deployment Commands

SAS Risk Modeling Requires a Git Repository

Starting with version 2023.07, SAS Risk Modeling requires a Git repository. The requirements for the repository are described at "Requirements for SAS® Risk Modeling" in *System Requirements for the SAS Viya Platform*.

Additionally, parameters in the **`$deploy/sas-bases/examples/sas-risk-cirrus-core/resources/core_transform.yaml`** file should be revised to account for the addition of the Git repository. `SAS_RISK_CIRRUS_CODE_LIB_GIT_SETUP_NEEDED` should be set to `Y`. The following parameters should be set to the appropriate values for your Git repository:

- `SAS_RISK_CIRRUS_CODE_LIB_REPO_NAME`
- `SAS_RISK_CIRRUS_CODE_LIB_REPO_USER`
- `SAS_RISK_CIRRUS_CODE_LIB_REPO_EMAIL`

- SAS_RISK_CIRRUS_CODE_LIB_GIT_REPO_FULL_URL

For the details about these parameters, see the README file located at **$deploy/sas-bases/examples/sas-risk-cirrus-core/resources/README.md** (for Markdown format) or at **$deploy/sas-bases/docs/preparing_and_configuring_risk_cirrus_core_for_deployment.htm** (for HTML format).

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*.

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. SAS Event Stream Processing Studio must be configured to use the PV. For more information, see "Managing Persistent Volumes".

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 "Requirements for SAS Infrastructure Data Server" 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. SAS Event Stream Processing Studio must be configured to use the PV. For more information, see "Managing Persistent Volumes".

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 "Requirements for SAS Infrastructure Data Server" 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".

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 "Requirements for SAS Infrastructure Data Server" 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 Tasks*.

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 432.

   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 Properties* and revise the configuration to best suit your needs. More details can be found in Identity Management: Overview.

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 439. 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 422. 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 439.

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 436. 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 445 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 443. 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: General Management of 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 450), 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 "Configure the Certificate Generator" 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
```

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

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, 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.

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 Data Governance: 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 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
```

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 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.