



SAS[®] Viya[®]: Models Administration

2020.1 - 2021.1.1

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

<i>Administering Models: Overview</i>	2
Modeling Integration Points	3
<i>Access to Models</i>	5
Overview	5
Initial Access to Registered Models	8
How to Narrow Access to Models in Standard Repositories	10
How to Narrow Access to Models in a Custom Repository	15
How to Create Repositories	20
<i>Configuring Model Data Libraries</i>	22
About Configuring Model Data Libraries	22
SAS Model Manager Predefined Caslibs	22
File System Directories	23
Add ModelStore and ModelPerformanceData Caslibs to a CAS Server	23
<i>Accessing Analytic Store Model Files</i>	24
Analytic Store Directories	24
Directory Permissions	25
Migrating Analytic Store Models	25

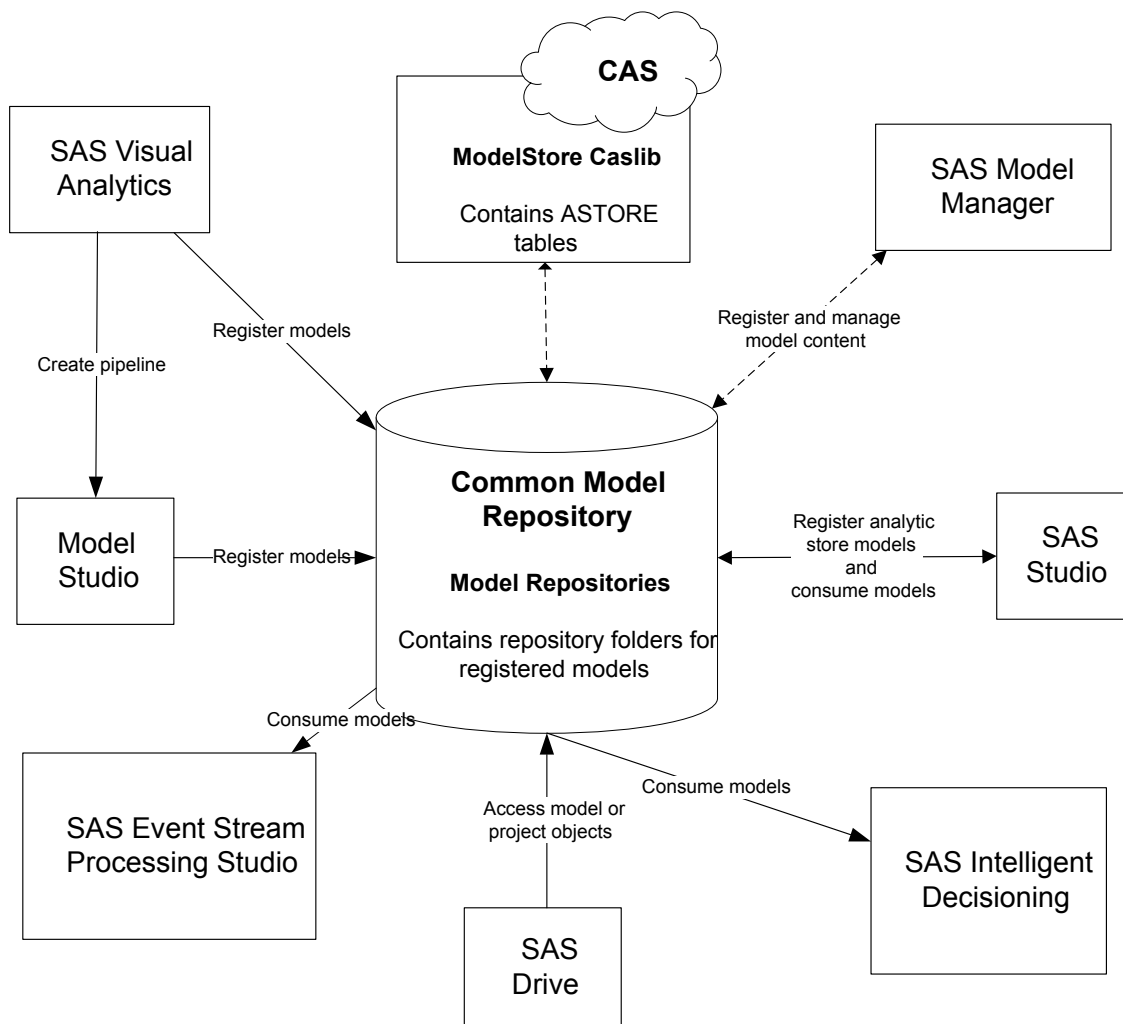
Administering Models: Overview

This document covers how to configure access to models and the modeling integration points between the common model repository and SAS Viya products. The common model repository is associated with the **Model Repositories** content root folder and can contain one or more repository folders. The Model Repository service provides access to a common model repository for SAS applications, and enables users to perform the following actions:

- import models into SAS Model Manager
- register models from Model Studio, SAS Visual Analytics, and SAS Studio into the common model repository
- add a model from the common model repository into a decision flow in SAS Intelligent Decisioning
- publish models or decisions to SAS Cloud Analytic Services (CAS), Hadoop, Teradata, and the SAS Micro Analytic Service
- refer to models that are stored in the common model repository from a project within SAS Event Stream Processing Studio.

Modeling Integration Points

Here are the integration points for the common model repository:



CAS

CAS stores analytic store files in the ModelStore caslib. Users can also publish models from SAS Model Manager and Model Studio to a CAS publishing destination.

See [“Data” in SAS Cloud Analytic Services: Fundamentals](#).

Common Model Repository

The common model repository contains multiple repositories that are used to store models that have been registered from SAS Model Manager, Model Studio, SAS Studio, and SAS Visual Analytics. The models can then be accessed from other SAS applications, such as SAS Intelligent Decisioning, or published to external publishing destinations.

See [“Managing Model Repositories” in SAS Model Manager: User’s Guide](#).

Model Studio

A suite of SAS products that enables you to build SAS Visual Data Mining and Machine Learning models as well as SAS Visual Text Analytics models and then to register them into the common model repository. SAS Visual Data Mining and Machine Learning models can also be published from Model Studio to a configured published destination.

See [SAS Visual Data Mining and Machine Learning: User's Guide](#) and [SAS Visual Text Analytics: User's Guide](#).

SAS Drive

A hub for the SAS Viya applications that enables you to easily view, organize, and share your content from one place. The availability of the features in SAS Drive depends on the applications that have been installed and the features and permissions that have been specified by your administrator.

See [SAS Drive: Documentation](#)

SAS Event Stream Processing Studio

A SAS web application that enables you to create, edit, upload, publish, and test event stream processing models using SAS Event Stream Processing Studio Modeler. SAS Event Stream Processing Studio Modeler displays a model as a data flow diagram, which enables you to see and control how windows relate and flow to one another. Projects can reference models that are stored in the common model repository. When a project is deployed, the model is retrieved from the common model repository and written to the ESP server. SAS Micro Analytic Service modules are used to accommodate the imported content that was created in SAS Model Manager. The module is uploaded and then referenced from the Calculate window's input handler. See [SAS Event Stream Processing: Using SAS Event Stream Processing Studio](#)

SAS Intelligent Decisioning

SAS web application that enables you to combine analytical models, rule sets, and conditional logic into decisions. You can investigate various scenarios, test and refine the decision logic, and then publish the decisions for use in batch applications and online transactions. After a decision has been published, it is available for use by other applications. See [SAS Intelligent Decisioning: User's Guide](#).

SAS Model Manager

SAS web application that enables you to store and manage models in a common model repository, as well as to organize them within projects and folders. You can import models that you developed using a SAS application (such as Model Studio, SAS Visual Analytics, and SAS Studio) as well as SAS code, open-source programming languages, and PMML. You can also create a new model with the model's files in a folder or project. Models that are located within a project can be evaluated for champion model selection and monitored for performance. Models within projects can also be published to a configured publishing destination that can be defined for CAS, Hadoop, SAS Micro Analytic Service, and Teradata, as well as Amazon Web Services (AWS), Azure, and Private Docker containers.

See [SAS Model Manager: User's Guide](#).

SAS Studio

SAS development application for running SAS programs, which enables users to use macros to create, update, and delete objects within the common model repository. A user can also use the Register task to import a scoring model from SAS Studio into a SAS Model Manager project that is located within the common model repository. A scoring model is an analytic object in a CAS table. Scoring models can be created using several SAS Studio tasks such as the Forest task.

See "Register a Model" in [SAS Studio: Task Reference Guide](#).

SAS Visual Analytics

A SAS web application that enables you to explore, discover, and predict using your data. If SAS Visual Statistics is licensed at your site, then you can create, test, and compare models based on the patterns that are discovered during exploration of your data. You can export a model before or

after performing model comparison in order to use it with other SAS products in a production environment. If SAS Visual Data Mining and Machine Learning is licensed at your site, then additional models are available. SAS Visual Data Mining and Machine Learning cannot be licensed without SAS Visual Statistics. You can also register SAS Visual Data Mining and Machine Learning and SAS Visual Statistics models from SAS Visual Analytics to the common model repository.

See [SAS Visual Analytics: Working with SAS Visual Data Mining and Machine Learning](#) and [SAS Visual Analytics: Working with SAS Visual Statistics](#).

Access to Models

Overview

Registered models are stored within the repository folders that reside within the **Model Repositories** content root folder. Initially, models that are registered to the SAS Model Manager default repository and standard repositories are accessible by authenticated users. See [“Initial Access to Registered Models”](#).

Model Studio projects contain models and are stored beneath a user’s folder in **My Folder**. Initially, Model Studio projects and models are private. For more information, see the following documentation:

- [“Managing Projects” in SAS Visual Data Mining and Machine Learning: User’s Guide](#)
- [“Managing Projects” in SAS Visual Text Analytics: User’s Guide](#)

You can create the standard repositories for Model Studio and SAS Visual Analytics before they have been deployed in order to modify the access to the repositories. The default repository and standard repositories can initially be accessed by all authenticated users. You can also create custom repositories and set the permissions so that only specific users can view or update content in them.

IMPORTANT

- Only SAS Administrators and other authorized users can create, update, or delete repository folders. In addition, custom repositories by default cannot be accessed by authenticated users. A SAS Administrator must grant a user’s access to custom repositories.
- With the release of SAS Viya 2020.1.4, when a user tries to register a model from Model Studio or SAS Visual Analytics, if they do not have access to the default repository, then a private user repository is created for them. The name of the repository is their user ID. The user can then register models into their private repository or into another repository to which they have Read and Write permissions.

Here are the different tools that you can use to create repositories:

- SAS Model Manager web application
- models plug-in to the SAS Viya command-line interface (CLI)

- %MM_CREATE_REPOSITORY macro
- Model Repository API request

For more information, see [“How to Create Repositories” on page 20](#).

CAUTION

Projects and models that are registered into a standard repository should not be moved using content selection windows within SAS Environment Manager and other SAS web applications. Moving a project or model that was registered from Model Studio breaks the connection between SAS Model Manager and Model Studio. Models that are copied to another folder do not retain the connection to the model in Model Studio.

Table 1 Initial Default and Standard Repositories

Repository Name	Description
Public	SAS Model Manager default repository for authenticated users to create projects and import models. The initial default repository is predefined as Public .
DMRepository	SAS Visual Data Mining and Machine Learning models registered from Model Studio to the common model repository are stored within the DMRepository by default. The DMRepository is made available with SAS Visual Data Mining and Machine Learning as part of the Model Studio installation process.
VARRepository	SAS Visual Data Mining and Machine Learning and SAS Visual Statistics models registered from SAS Visual Analytics to the common model repository are stored within the VARRepository by default. The VARRepository folder is created the first time a SAS Visual Analytics report object is registered as a model to the common model repository.
VTARRepository	SAS Visual Text Analytics models registered from Model Studio to the common model repository are stored within the VTARRepository by default. The VTARRepository folder is created the first time a SAS Visual Text Analytics model is registered from Model Studio to the common model repository.

Here are the initial authorizations for the **Model Repositories** folder, the default repository folder, the standard repository folders, and a custom repository folder.

Note: The SAS Demo User is an example of a user that assumed the SAS Administrators group when signing in to SAS Environment Manager or another SAS web application.

The following figure is an example of the initial authorization for the **Model Repositories** folder.

Figure 1 Authorization for the Model Repositories Folder

Principal	Read	Update	Delete	Secure	Add	Remove	Read (convey)	Update (convey)	Delete (convey)	Secure (convey)	Add (convey)	Remove (convey)
Authenticated Users	✓ ♦	✗	✗	✗	✓ ♦	✗	✗	✗	✗	✗	✗	✗
SAS Administrators	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
SAS Demo User	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗

The following figure is an example of the initial authorization for the default repository (Public) and standard repository (DMRepository, VARRepository, and VTARRepository) folders.

Figure 2 Initial Authorization for the Default Repository and the Standard Repository Folders

Principal	Read	Update	Delete	Secure	Add	Remove	Read (convey)	Update (convey)	Delete (convey)	Secure (convey)	Add (convey)	Remove (convey)
Authenticated Users	✓ ♦	✗	✗	✗	✓ ♦	✓ ♦	✓ ♦	✓ ♦	✓ ♦	✓ ♦	✓ ♦	✓ ♦
SAS Administrators	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SAS Demo User	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

The following figure is an example of the initial authorization for a custom repository folder.

Figure 3 Initial Authorization for a Custom Repository Folder

Principal	Read	Update	Delete	Secure	Add	Remove	Read (convey)	Update (convey)	Delete (convey)	Secure (convey)	Add (convey)	Remove (convey)
Authenticated Users	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
SAS Administrators	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
SAS Demo User	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗

Initial Access to Registered Models

Note: Initially, all authenticated users have Read and Write access to all registered models in the default repository and standard repositories. Restrict access as appropriate for your usage patterns and security goals. See [“How to Narrow Access to Models in Standard Repositories”](#).

Common Model Repository Predefined Rules

The predefined rules grant Authenticated Users Read and Add access to the default repository and standard model repositories, model projects, and models as follows:

CAUTION

The Common Model Repository predefined rules should not be modified. Authorization changes should be implemented on a repository folder or folders within a repository folder.

Here are the initial authorization rule permissions for the Model Repository service (/modelRepository) endpoints for the new SAS Viya deployment.

Figure 4 Common Model Repository Rules (/modelRepository)

/modelRepository			
Object URI	Principal	Setting	Permissions
/modelRepository/	Authenticated Users	Grant	Read
/modelRepository/models	Authenticated Users	Grant	Read, Create
/modelRepository/projects	Authenticated Users	Grant	Read, Create
/modelRepository/repositories/*	SASAdministrators	Grant	Update, Delete
/modelRepository/repositories	sasapp	Grant	Create
/modelRepository/models/	Authenticated Users	Grant	Read
/modelRepository/projects/userProperties	Authenticated Users	Grant	Read
/modelRepository	Authenticated Users	Grant	Read
/modelRepository/repositories	Authenticated Users	Grant	Read
/modelRepository/projects/	Authenticated Users	Grant	Read
/modelRepository/models/transfer	Authenticated Users	Grant	Create
/modelRepository/models/userProperties	Authenticated Users	Grant	Read
/modelRepository/repositories	SASAdministrators	Grant	Create
/modelRepository/jobs	Authenticated Users	Grant	Create
/modelRepository/keywords	Authenticated Users	Grant	Read

Predefined Rules for the Model Repositories Folder

Predefined rules for the **Model Repositories** folder grant access to Authenticated Users as follows:

Target: /Model Repositories/ (folder)

Object URI: /folders/folders/folder-ID/**

Permission Read, Add
s:

Target: /Model Repositories/ (folder)

Object URI: /folders/folders/folder-ID/members

Permission Read
s:

Predefined rules for the **Model Repositories** folder grant access to the sas.modelRepository user as follows:

Target: /Model Repositories (folder)

Container URI/folders/folders/folder-ID/**

Permission Read, Remove, Secure, Update, Add, Delete
s: **Note:** The user that creates the folder can access it, by default.

Predefined rules for the **Model Repositories** folder grant access to the sas.workflow user as follows:

.....
Note: This predefined rule exists only if you have a license for SAS Model Manager.
.....

Target: /Model Repositories (folder)

Container URI/folders/folders/folder-ID/**

Permission Read, Remove, Update, Add
s: **Note:** The user that creates the folder can access it, by default.

Generated Rules for Each Repository

Two generated rules on each immediate child of the **Model Repositories** folder grant access to Authenticated Users. For example, generated rules on the **DMRepository** folder grant access to Authenticated Users as follows:

Target: /Model Repositories/DMRepository/** (folder)

Object URI: /folders/folders/folder-ID/**

Permission Read, Create, Remove, Add
s:

Target: /Model Repositories/DMRepository (folder)

Container URI/folders/folders/folder-ID

Permission Read, Create, Remove, Secure, Update, Add, Delete
s:

See Also

- [“Managing Permissions” in SAS Model Manager: Administrator’s Guide](#)
- [“General Authorization: Concepts” in SAS Viya: General Authorization](#)
- [“Rules Page” in SAS Environment Manager: User’s Guide](#)
- [SAS Viya: General Authorization Window](#)

How to Narrow Access to Models in Standard Repositories

Overview

Perform the following tasks to narrow access to models in standard repositories:

IMPORTANT Do not change the default repository permissions for authenticated users. Create a custom group to limit user access to the default repository.

- 1 [Create custom groups.](#)

- 2 Check whether the licensed products exist in your SAS Viya deployment.
- 3 Create the standard repositories.
- 4 Secure the Model Repositories folder and its immediate children.
- 5 Reduce the availability of model applications and features.

Create Custom Groups

Create two new [custom groups](#) as follows:

Name	ID	Membership Description
Modelers	_modelers	Anyone who uses model functionality.
Model Repository Administrators	_modelRepoAdmins	Anyone who should be able to add, update, and delete repositories.

Note: The names and IDs are suggested values. You can specify different values.

Note: If you have planned to use separated access, where different groups access different models, create additional groups as members of the Modelers group.

Check Whether the Licensed Products Exist in Your SAS Viya Deployment

- 1 From the applications menu () , select **Manage Environment**.
- 2 In SAS Environment Manager, click  to view the **Licensed Products** page.
- 3 Enter **SAS Model Manager** in the **Product** filter field.
- 4 Enter the following licensed products names in the **Product** filter field.

Licensed Product	Standard Repository
SAS Data Mining and Machine Learning	DMRepository
SAS Visual Analytics Explorer	VARepository
Text Analytics UI Setinit	VTARepository

TIP If the licensed product does not exist, then you do not need to create the associated standard repository or to secure access to the repository.

Create the Standard Repositories

Not all deployments include all repositories. The default repository “Public” is included for all installations. The standard repositories are not available until you have a license for the associated product and a SAS Administrator creates them or an authorized user registers a model from Model Studio or SAS Visual Analytics.

On the **Content** page in SAS Environment Manager, verify that the following repository folders do not already exist directly beneath the **Model Repositories** root folder. If they already exist, you do not need to create the standard repositories. You can continue to the next step and [secure the Model Repositories folder and its immediate children](#).

- VARepository
- DMRepository
- VTARepository

If you have a license for SAS Model Manager, and the standard repositories do not already exist, a SAS Administrator can create them using the SAS Model Manager web application, models CLI plug-in, or the %MM_CREATE_REPOSITORY macro. When creating the standard repositories before they have been deployed, the names must match exactly. See [“Create a New Repository Using SAS Model Manager”](#).

If you do not have a license for SAS Model Manager, then the standard repositories can be created by using the models CLI plug-in or registering models from Model Studio and SAS Visual Analytics. See [“Create Standard Repositories by Registering Models”](#).

Secure the Model Repositories Folder and Its Immediate Children

- 1 In SAS Environment Manager, click  to view the **Content** page.
- 2 Click the arrow to the right of the **SAS Content** folder to locate the **Model Repositories** folder. Right-click the folder and select **Edit authorization**.

Note: Only users with SECURE permission to the selected folder can edit its authorization.

Adjust access as follows:

- a Click  and select **Add identities**. The Add Identities window appears.

Note: If guest access is not configured, the Add Identities window appears when you click .

- b Select the **Modelers** and **Model Repository Administrators** custom groups as identities. Click **OK**.

Note: Both groups inherit Read and Add object permissions.

- c For the Model Repository Administrators group, click  to set a direct grant for object permissions of Remove, as well as grant container (convey) permissions of Read, Update, Delete, Secure, Add, and Remove.
- d For the Modelers group, click  to set a direct grant for object permissions of Read and Add.
- e Click **Preview**.

After the **Model Repositories** folder has been secured, here are its authorizations.

Principal	Read	Update	Delete	Secure	Add	Remove	Read (convey)	Update (convey)	Delete (convey)	Secure (convey)	Add (convey)	Remove (convey)
 Authenticated Users	 ♦				 ♦							
 Model Repository Administrators						 ♦	 ♦	 ♦	 ♦	 ♦	 ♦	 ♦
 Modelers	 ♦				 ♦							
 SAS Administrators												
 SAS Demo User												

- f Click **Save**.

3 Right-click a repository (any immediate child of the **Model Repositories** folder) and select **Edit authorization**.

Note: Only users with SECURE permission to the selected folder can edit its authorization.

- a Click  and select **Add identities**. The Add Identities window appears.
- b Select the **Modelers** custom group as an identity. Click **OK**.
- c Click  in each cell to remove the current grants that are set for all object permissions for Authenticated Users. Change the **Direct setting** value to **None**.

Note: Do not change the **Direct setting** value to **Prohibit**.

- d For the Modelers group (or if you are securing a custom repository for a more specific group), click  in each cell to add direct object grants of Read, Add, and Remove and container (convey) grants of Read, Update, Delete, Secure, Add, and Remove.
- e Click **Preview**.

Here is an example of the authorizations for the **DMRepository** folder.

DMRepository													
☐ Show horizontal column headings 👤 ↺ ?													
Principal	Read	Update	Delete	Secure	Add	Remove	Read (convey)	Update (convey)	Delete (convey)	Secure (convey)	Add (convey)	Remove (convey)	
👤 Authenticated Users	🚫	🚫	🚫	🚫	🚫	🚫	🚫	🚫	🚫	🚫	🚫	🚫	
👤 Model Repository Administrators	✅	✅	✅	✅	✅	✅	✅	✅	✅	✅	✅	✅	
👤 Modelers	✅ ♦	🚫	🚫	🚫	✅ ♦	✅ ♦	✅ ♦	✅ ♦	✅ ♦	✅ ♦	✅ ♦	✅ ♦	
👤 SAS Administrators	✅	✅	✅	✅	✅	✅	🚫	🚫	🚫	🚫	🚫	🚫	
👤 SAS Demo User	✅	✅	✅	✅	✅	✅	🚫	🚫	🚫	🚫	🚫	🚫	

🔧 Click **Save**.

Repeat this step for each repository (each immediate child of the **Model Repositories** folder). Here are key points:

- The DMRepository model repository is created during the installation process. The VARRepository and VTARRepository model repositories are created automatically the first time a user registers a model. It is recommended that an administrator create those repositories and adjust access before the deployment is made available to users. For details, see [“Create the Standard Repositories”](#).
- Whenever a custom repository is added, access to that repository must be adjusted. (For that reason, these instructions allow only Model Repository Administrators to add repositories.) For more information, see [“How to Narrow Access to Models in a Custom Repository”](#) on page 15.

Reduce the Availability of Model Applications and Features

The purpose of this task is to enhance usability rather than to increase security.

- 1 In SAS Environment Manager, click  to view the **Rules** page.
- 2 For each rule in the following table, replace the original principal (principalType=Authenticated Users) with the custom modelers group (for example, principalType=group, name=Modelers, and group ID=_modelers) that you created earlier. For more information, see [“Create Custom Groups”](#) on page 11.

Application or Feature	Target URI (or URIs)
Ability to use model objects in SAS Visual Analytics	/SASVisualAnalytics_capabilities/buildAnalyticalModel
Ability to access SAS Model Manager ¹	/SASModelManager/ and /SASModelManager

Application or Feature	Target URI (or URIs)
Ability to access Model Studio ²	/ModelStudio/**

- 1 These are two alternate forms of the same URI. In the current release, because SAS Model Manager uses both forms, both rules are necessary.
- 2 In the current release, you cannot limit direct access to Model Studio. The adjusted rule limits access to Model Studio only from the applications menu.

How to Narrow Access to Models in a Custom Repository

Overview

Perform the following tasks to narrow access to models in a custom repository:

IMPORTANT Do not change the default repository permissions for authenticated users. Create a custom group to limit user access to the default repository.

- 1 Create custom groups.
- 2 Check whether the licensed product exists in your SAS Viya deployment.
- 3 Add model repository rules for a custom repository.
- 4 Create a custom repository.
- 5 Secure the Model Repositories folder and its immediate children.
- 6 Reduce the availability of model applications and features.

Create Custom Groups

Create two new [custom groups](#) as follows:

Name	ID	Description
Modelers	_modelers	Anyone who uses model functionality.
Model Repository Administrators	_modelRepoAdmins	Anyone who should be able to add, update, and delete repositories.

Note: The names and IDs are suggested values. You can specify different values.

Note: If you have planned to use separated access, where different groups access different models, create additional groups as members of the Modelers group.

Check Whether the Licensed Products Exist in Your SAS Viya Deployment

- 1 From the applications menu () , select **Manage Environment**.
- 2 In SAS Environment Manager, click  to view the **Licensed Products** page.
- 3 Enter **SAS Model Manager** in the **Product** filter field.

Add Model Repository Rules for a Custom Repository

In order to create, delete, and update custom repositories within a custom repository, additional rules must be added.

- 1 In SAS Environment Manager, click  to view the **Rules** page.
- 2 Click .
- 3 By default, only SAS Administrators can create, update, or delete repository folders. You can add additional Model Repository rules for repository object access to enable other users such as the Model Repository Administrators to perform these actions.
 - Here is the rule to add for granting users the ability to read and create a repository:

Field Name	Value
Object URI	/modelRepository/repositories
Principal type	Group
Principal	Model Repository Administrators
Rule type	Grant
Permissions	Read, Create
Reason	Provides the ability to create a model repository.

- Here is the rule to add for granting users the ability read, delete, and update repositories:

Field Name	Value
Object URI	/modelRepository/repositories/*
Principal type	Group
Principal	Model Repository Administrators
Rule type	Grant
Permissions	Read, Delete, Update
Reason	Provides the ability to delete, read, and update repositories.

For more information, see [“Add a Rule” in SAS Environment Manager: User’s Guide](#).

Create a Custom Repository

A SAS Administrator or another authorized user can create a custom repository using the SAS Model Manager web application, the models CLI plug-in, or the %MM_CREATE_REPOSITORY macro. See [“Create a New Repository Using SAS Model Manager”](#).

Note: You can authorize other users or groups the ability to create and update custom repositories using rules. See [“Add Model Repository Rules for a Custom Repository” on page 16](#).

Secure the Model Repositories Folder and Its Immediate Children

- 1 From the applications menu () , select **Manage Environment**.
- 2 In SAS Environment Manager, click  to view the **Content** page.
- 3 Click the arrow to the right of the **SAS Content** folder to locate the **Model Repositories** folder. Right-click the folder and select **Edit authorization**.

Note: Only users with SECURE permission to the selected folder can edit its authorization.

Adjust access as follows:

- a Click  and select **Add identities**. The Add Identities window appears.
- b Select the **Modelers** and **Model Repository Administrators** custom groups as identities. Click **OK**.

Note: Both groups inherit Read and Add object permissions.

- c For the Model Repository Administrators group, click  to set a direct grant for object permissions of Remove, as well as grant container (convey) permissions of Read, Update, Delete, Secure, Add, and Remove.
- d For the Modelers group, click  to set a direct grant for object permissions of Read and Add.

Note: If you have updated your installation from a previous release, the Authenticated Users principal group might have container (convey) permissions for the **Model Repositories** folder. Remove container access for Authenticated Users by changing the direct setting to **(none)** for each convey permission. Authenticated users should have only Read and Add object permissions.

- e Click **Preview**.

After the **Model Repositories** folder has been secured, here are its authorizations:

Principal	Read	Update	Delete	Secure	Add	Remove	Read (convey)	Update (convey)	Delete (convey)	Secure (convey)	Add (convey)	Remove (convey)
 Authenticated Users												
 Model Repository Administrators												
 Modelers												
 SAS Administrators												
 SAS Demo User												

- f Click **Save**.

- 4 Right-click a repository (any immediate child of the **Model Repositories** folder) and select **Edit authorization**.

Note: Only users with SECURE permission to the selected folder can edit its authorization.

- a Click  and select **Add identities**. The Add Identities window appears.
- b Select the **Modelers** custom group as an identity. Click **OK**.
- c For the Modelers group (or if you are securing a custom repository for a more specific group), click  in each cell, and grant object permissions of Read, Add, and Remove, as well as container (convey) grants of Read, Update, Delete, Add, and Remove.
- d Click **Preview**.

Here is an example of the authorizations for the custom repository folder:

Edit Authorization													
Custom_Repository													
☐ Show horizontal column headings 👤 ↺ ?													
Principal	Read	Update	Delete	Secure	Add	Remove	Read (convey)	Update (convey)	Delete (convey)	Secure (convey)	Add (convey)	Remove (convey)	
👤 Authenticated Users	❌	❌	❌	❌	❌	❌	❌	❌	❌	❌	❌	❌	
👤 Model Repository Administrators	✅	✅	✅	✅	✅	✅	✅	✅	✅	✅	✅	✅	
👤 Modelers	✅ ♦	❌	❌	❌	✅ ♦	✅ ♦	✅ ♦	✅ ♦	✅ ♦	❌	✅ ♦	✅ ♦	
👤 SAS Administrators	✅	✅	✅	✅	✅	✅	❌	❌	❌	❌	❌	❌	
👤 SAS Demo User	✅	✅	✅	✅	✅	✅	❌	❌	❌	❌	❌	❌	

- Click **Save**.

Repeat this step for each repository (each immediate child of the **Model Repositories** folder).

Whenever a custom repository is added, access to that repository must be adjusted. (For that reason, these instructions allow only Model Repository Administrators to add repositories.) For more information, see [“How to Narrow Access to Models in a Custom Repository”](#) on page 15.

Reduce the Availability of Model Applications and Features

The purpose of this task is to enhance usability rather than to increase security.

- 1 In SAS Environment Manager, click  to view the **Rules** page.
- 2 In each rule in the following table, replace the original principal (principalType=Authenticated Users) with the custom modelers group (for example, principalType=group, name=Modelers, and group ID=_modelers) that you created earlier. For more information, see [“Create Custom Groups”](#) on page 11.

Application or Feature	Target URI (or URIs)
Ability to access SAS Model Manager ¹	/SASModelManager/ and /SASModelManager

¹ These are two alternate forms of the same URI. In the current release, because SAS Model Manager uses both forms, both rules are necessary.

How to Create Repositories

Overview

Here are the different methods that you can use to create a new custom repository or standard repository:

IMPORTANT Only licensed users of SAS Model Manager can use the web application or macro to create a new repository.

- SAS Model Manager web application
- models CLI plug-in
- %MM_CREATE_REPOSITORY macro
- Model Repository API

Note: The **DMRepository** is made available with SAS Visual Data Mining and Machine Learning as part of the Model Studio installation process. If you do not create the standard repositories for SAS Visual Text Analytics (Model Studio) and SAS Visual Analytics as part of the post-installation and configuration tasks, the repositories are created the first time a user registers a model. Custom repositories are not currently supported for Model Studio and SAS Visual Analytics.

For background information about repository access, see [“Initial Access to Registered Models” on page 8](#)

Create a New Repository Using SAS Model Manager

To create a new custom or standard repository using the SAS Model Manager web application:

- 1 From the applications menu () , select **Manage Models**.
- 2 Click your name in the application bar and select **Settings** ⇒ **SAS Model Manager** ⇒ **Repositories**.
- 3 Click .
- 4 Enter a name for the repository.

IMPORTANT If you are creating a standard repository for use with Model Studio or SAS Visual Analytics, the name must match exactly what is in the [Table 1 on page 6](#). You can name a custom repository anything you want, as long as it does not already exist.

- 5 (Optional) Enter a description of the repository.

Note: After you save the new repository, the description cannot be edited.

6 Click **Save**.

7 Click **Close**.

See Also

“%MM_CREATE_REPOSITORY Macro” in *SAS Model Manager: Macro Reference*

Create Repositories Using the Models CLI Plug-In

You can use the models CLI plug-in to create custom and standard repositories.

Here is an example of using the CLI to retrieve a list of the repositories and creating a new custom repository.

Note: The commands should be entered as a single line.

```
sas-viya models repository list
```

ID	Name	Version	Description
fcc84739-4d0f-4e62-a195-1e8fc72bca5b	Public	2	This is the default model repository.
6160decf-6ce8-4309-abc4-4baf0efab401	DMRepository	2	<nil>

```
sas-viya models repository create --name "Custom_Repository" --description "My custom repository."
```

ID	Name	Version	Description
584c2994-254c-4787-9277-9d2d5bf5fddc	Custom_Repository	2	My custom repository.

For more information, see [SAS Viya: Models Command-Line Interface](#).

Create Standard Repositories by Registering Models

If you do not have a license to SAS Model Manager, you can use the models CLI plug-in to create the standard repositories by registering models from Model Studio and SAS Visual Analytics.

Note: The **DMRepository** is made available with SAS Visual Data Mining and Machine Learning as part of the Model Studio installation process.

For information about registering models from SAS Visual Analytics and SAS Text Analytics models from Model Studio, see the following documentation:

- “Registering a Model” in *SAS Visual Analytics: Working with SAS Visual Statistics*

Configuring Model Data Libraries

About Configuring Model Data Libraries

During the deployment of the Model Repository service, the ModelPerformanceData and ModelStore caslibs are created on each CAS server (for example, cas-shared-default and cas-shared-mpp). The source type for the caslibs is a file system path. All users are granted Read and Write permissions to the source file system directory paths.

The ModelPerformanceData caslib stores the performance results data tables that are created when you run a performance job for a model. This caslib can be used only if you have the SAS Model Manager web application and the Model Management service.

The ModelStore caslib contains analytic store files for models that are created using SAS Visual Analytics, Model Studio, or SAS Studio. The ModelStore caslib must exist on each CAS server that is in use. If your model references a train table in a library that exists on a CAS server, and that server does not have a ModelStore caslib, you cannot register the model in the SAS Model Manager common model repository. An error message is displayed.

Note: If you migrate analytic store models from a SAS Viya 3.5 environment to a SAS Viya 2020.1 or later environment, you must copy the ModelStore caslib content from the source system to the target system. For more information, see [“Migrating Analytic Store Models” on page 25](#).

SAS Model Manager data is managed using SAS Environment Manager and SAS Administration CLI. You can create data libraries (caslibs) and import tables to be used by SAS Model Manager and other SAS applications. For more information, see [“Making Data Available to CAS” in SAS Data Explorer: User’s Guide](#).

SAS Model Manager Predefined Caslibs

The following caslibs are automatically created during the deployment of SAS Model Manager. Each caslib has a default assignment and specifications. For a list of the SAS Viya predefined caslibs, see [“Predefined Caslibs” in SAS Viya: Data](#).

Note: The CAS data directory path is configured during installation. The default value for the CASDATADIR environment variable is `/cas/data/caslibs/`.

Table 2 Predefined Caslibs

Caslib	Default Assignment
ModelPerformanceData	<code>/cas/data/caslibs/modelMonitorLibrary/</code> Stores application-generated performance results data that is used for reporting. The directory assignment is on the CAS controller node.
ModelStore	<code>/cas/data/caslibs/modelStore/</code> The file system directory is associated with the ModelStore CAS library. The directory assignment is on the CAS controller node. This library stores tables (SASHDAT), and each table contains one analytic store. The table is created when an analytic store model is registered to the common model repository. The analytic store file is created in the associated directory.

File System Directories

When defining a caslib where the source type is a file system directory path, the appropriate permissions must be granted. For more information, see [“CAS Authorization: Host Access” in SAS Viya: CAS Authorization](#).

By default, CAS sessions run using the **sas** account. The CASHostAccountRequired custom group is a SAS Viya reserved group name, but it is not created during the deployment of SAS Viya. If you add identities or groups to the custom group with the ID CASHostAccountRequired, members of this group automatically run their CAS sessions under their own host account. By default, all authenticated users are granted Read and Write permissions to the ModelStore and the ModelPerformanceData caslib file system directory paths in order to register analytic store models and to generate performance results.

For more information, see [“The CASHostAccountRequired Custom Group” in SAS Viya: Identity Management](#).

Add ModelStore and ModelPerformanceData Caslibs to a CAS Server

Note: If you add a new CAS server after the deployment of the Model Repository service on SAS Viya, verify that the ModelStore and ModelPerformanceData caslibs exist. If they do not exist, you must add the caslibs to the new CAS server.

- 1 Sign in to SAS Environment Manager as an administrator.

Note: If you are already logged in to SAS Model Manager, you can access SAS Environment Manager by clicking  and selecting **Manage Environment**.

- 2 Click  in the navigation bar. The **SAS Data Explorer** page appears.
- 3 Click the **Data Sources** tab.
- 4 Click  on the **Data Sources** tab. The Connections Settings window appears.
- 5 Enter the name of the library (for example, ModelStore).
- 6 Select the CAS server (for example, **cas-shared-myserver**) where you want the new caslib to reside.
- 7 Select **File system** from the **Type** drop-down list.
- 8 Select the **Persist this connection beyond the current session** check box to add a global caslib for this connection.
- 9 Select **PATH** as the data source type.
- 10 Enter a value for **Path** on the **Settings** tab.

Note: Use a similar path to the one specified for the **ModelStore** caslib on the **cas-shared-default** server.

- 11 Click **Save**.
- 12 Repeat steps 4 through 11 for the ModelPerformanceData caslib.
- 13 Restart the Model Repository service. For more information, see [SAS Viya Operations: Updating Software](#).

Accessing Analytic Store Model Files

Analytic Store Directories

The ASTORES persistent storage volume (PV) and the overlays are configured during the deployment of SAS Viya. The overlays mount the directory paths that point to the ASTORES persistent volume claim (PVC). The mounted directory paths enable the SAS Micro Analytic Service and Compute service to access the analytic store (ASTORE) files when running Compute service jobs and Launcher service jobs. For more information, see [“Configuring Analytic Store and Python Model Directories” in SAS Micro Analytic Service: Programming and Administration Guide](#).

The `/models/astores/viya` mounted directory path must be accessible in order to perform the following tasks:

- publish analytic store models (or decisions that use analytic store models) to a SAS Micro Analytic Service publishing destination

- analyze a model within SAS Micro Analytic Service
- run a test for a decision that contains an analytic store model using SAS Intelligent Decisioning
- access a project champion model in the common model repository from SAS Event Stream Processing Studio

Directory Permissions

During the deployment of SAS Viya, overlays are used to mount the `/opt/sas/viya/config/data/modelsvr/astore` and the `/models/astores/viya` directory paths. Both mounted directory paths reference the ASTORES PVC. By default, all users have Read and Write permissions to the ASTORES PV. The `sas` group has full permission to these directories.

For more information, see [“Storage Requirements” in *System Requirements for SAS Viya*](#) and [SAS Viya: Deployment Guide](#).

Migrating Analytic Store Models

When you migrate model content between SAS Viya environments for 2020.1 or later using SAS Environment Manager or the `sas-viya` CLI transfer plug-in, if a model contains references to analytic store files in the ModelStore caslib, then the analytic store files are transferred with the model.

If you migrate analytic store models from a SAS Viya 3.5 environment to a SAS Viya 2020.1 or later environment, you can use the `cas` plug-in to the SAS Viya CLI to copy ModelStore caslib content from the source system to the target system. The ModelStore caslib content is stored in the file system.

The analytic store files for SAS Viya 3.5 are located in the `/opt/sas/viya/config/data/cas/default/modelStore` directory by default on the machine where the CAS server is deployed. These files are located in the `/cas/data/caslibs/modelStore` directory by default within the CAS container for SAS Viya 2020.1 and later.

For more information, see [“Migrate Caslibs and Access Controls” in *SAS Viya: Content Migration from SAS Viya 4*](#).

Note: You can also export an analytic store model using the SAS Model Manager web application and the analytic store file is included in the ZIP file. If you use the SAS Model Manager web application to re-import the model ZIP file that contains an analytic store model to the same system or another system, the analytic store file is copied to the ModelStore caslib. However, when the model is re-imported a new model UUID is created.

For information about migrating models that are located in the common model repository, see [“Migrating Content” in *SAS Model Manager: Administrator’s Guide*](#).

