# Promotion: Overview

- What Is Promotion? ............................................. 2
- What Can Be Promoted? .................................... 2
- Who Can Promote? ........................................ 3

## Promotion from SAS 9: Tasks

- About These Tasks ........................................ 4
- Step 1: Create an Initial Map of System Information 4
- Step 2: Promote Internal Groups ....................... 6
- Step 3: Promote Data Definitions ...................... 6
- Step 4: Promote Content ................................ 8

## Promotion from SAS 9: Reference

- Participating Resources .................................. 9
- Details: Identities ......................................... 9
- Details: Authorizations .................................. 11
- Details: Data Definitions ............................... 14
- Details: Explorations ................................... 14
- Details: Reports .......................................... 15
- Workflow Examples ...................................... 17

## Promotion within SAS Viya: Tasks

- Introduction ................................................. 18
- Promote Resources That Are Stored in SAS Folders 19
- Promote Caslibs and Access Controls ................. 20
- Promote Resources That Are Not Stored in Folders 23

## Promotion within SAS Viya: Reference

- Participating Resources: Promotable with the Transfer Plug-in 25
- Other Participating Resources ............................ 26
- Details: Reports That Contain ASTORE Tables ...... 26
- Details: Jobs ............................................... 27
- Details: Themes .......................................... 28
- Additional Details ........................................ 28
- Process Flow Examples .................................. 30
Promotion: Overview

This document describes promotion of resources to SAS Viya.

What Is Promotion?

Promotion is the process of making resources that exist in one environment present, available, and usable in another environment.

The promotion process has two phases:

1. Export resources from the source environment.
2. Import resources to the target environment.

A central aspect of both phases is the creation, refinement, and application of mappings. Mappings govern the transformation of exported resources for use in the target environment. Transformation occurs during the import phase based on specified mappings. You can save, edit, and reuse mapping files.

IMPORTANT Starting in SAS Environment Manager 3.3, the import phase of the promotion process can be performed using the Import wizard. To determine your SAS Environment Manager release, click About under your user name in the top right. The release is listed in the pop-up window, along with other information such as site name and number.

What Can Be Promoted?

The following table summarizes the supported source environments and resources:
Table 1  Supported Resources for Promotion to SAS Viya

<table>
<thead>
<tr>
<th>Environment</th>
<th>Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAS 9 (9.4 and later)</td>
<td>Content: folders, reports, explorations</td>
</tr>
<tr>
<td></td>
<td>Supporting resources: graph templates and images</td>
</tr>
<tr>
<td></td>
<td>Data definitions: Libraries and tables for LASR, Base</td>
</tr>
<tr>
<td></td>
<td>User groups: most metadata-only groups</td>
</tr>
<tr>
<td></td>
<td>Authorizations: most explicit access control entries (ACEs) for promoted objects</td>
</tr>
<tr>
<td>SAS Viya 3.x</td>
<td>Content: folders, reports, data plans, statistical models, themes, and job requests/definitions/flows/actions</td>
</tr>
<tr>
<td></td>
<td>Supporting resources: graph templates, images, files, and comments</td>
</tr>
<tr>
<td></td>
<td>CAS authorization: all direct access controls for promoted objects</td>
</tr>
<tr>
<td></td>
<td>General authorization: all direct rules for promoted objects</td>
</tr>
</tbody>
</table>

What about SAS Visual Statistics Projects?

You cannot directly promote a SAS Visual Statistics project object in SAS 9 to SAS Viya.

To make the content of SAS Visual Statistics projects (in SAS 9 and from SAS Visual Statistics 6.4 or 7.1) available in SAS Viya, perform the following steps:

1 Upgrade to SAS Visual Analytics Explorer 7.2 or 7.3.

   Note: This upgrade converts SAS Visual Statistics project objects to SAS Visual Analytics objects.

2 Promote the SAS Visual Analytics objects to SAS Viya.

Who Can Promote?

You must have administrator privileges to promote data. For details and instructions, see SAS Viya Administration: Identity Management.
About These Tasks

Here are key points about these tasks:

- These tasks assume that you are familiar with the information in “Promotion from SAS 9: Reference”. For successful promotion, it is essential to understand process, scope, and effects.
- These tasks use graphical (point-and-click) interfaces where possible. You can use alternate tools as follows:
  - To export, you can use the SAS 9 metadata batch export tool.
  - To import, you can use the SAS Viya sas-admin transfer command-line interface.
- These tasks use JSON mapping files. For more information about JSON and YAML formats and mapping files, see “Mapping Files”.

Note: The SAS Viya Import wizard supports only JSON mapping files.

Note: Mapping files are cumulative. Within a series of imports, each successive mapping file supplements and refines a previous mapping file. For that reason, you should use the same format (JSON or YAML) for all of your mapping files.

- These tasks reference other documents for details about specific tools. Not all of the SAS 9 promotion documentation is applicable to promotion to SAS Viya. (Much of the SAS 9 promotion documentation is applicable only to promotion within SAS 9.)
- These tasks are organized by object type. For some usage patterns, a different workflow might be more efficient. For example, you might prefer to perform all exports first, and then perform all imports. In any workflow, make sure that you import resources in the specified order.
- These tasks use example file names. You can choose your own names.
- Before you import content to SAS Viya, make sure that you have a current backup.

Step 1: Create an Initial Map of System Information

Create a map of system information unless you already have a current mapping file.

1. Export metadata identities.
   b. On the first page of the Export SAS Package Wizard, specify a location and name. (These tasks use the name identities.spk as an example.)
On the next page, select all users and user groups. Do not export other types of security objects (such as domains, roles, and access control templates (ACTs)).

When the export is completed, copy the export package file to a location that is available to your SAS Viya deployment. If importing using the Import wizard, copy the export package file to a file system that is local to the web browser.

2 Export the SAS LASR Analytic Server server definition.

a In SAS Management Console, under **SAS Folders** on the **Folders** tab, expand **System**, and select **Servers**. In the right pane, select the servers that you want to export. Right-click, and select **Export SAS Package**.

b On the first page of the Export SAS Package Wizard, specify a location and name. (These tasks use the name servers.spk as an example. This package is not imported but is crucial to the import process.)

c On the next page, make sure that the server that you want to export is listed.

d When the export is completed, copy the export package file to a location that is available to your SAS Viya deployment. If importing using the Import wizard, copy the export package file to a file system that is local to the web browser.

3 Generate a mapping file. At a host prompt, go to the directory where your **SAS Viya command-line executable files** are stored. Enter `/opt/sas/viya/home/bin/sas-admin`. From here, you can submit the following commands:

```
sas-admin profile init
sas-admin auth login
sas-admin transfer generate-content-mapping --mapping output-path
   --user-package identities.spk
   --group-package identities.spk
```

**Note:** If you are creating output files from a command, be sure to run it from a directory where you have Write permission.

- For visibility, option values are highlighted. If you exported server definitions, add the `--server-package` option to the `generate-content-mapping` command to reference that export package file.
- The output file is named `ContentMappings.json`. Make sure the output location does not already contain a file with the same name if you want to preserve that existing file.
- If you are promoting a LASR library, update the default authentication domain named `LASRAUTH` with appropriate credentials. This will enable you to successfully open the LASR caslib after the import.

4 Use a text editor to review, refine, and save the mapping file.

a Make sure the file's contents are as expected. See “Mapping Files”.

b If you want to change any target values, edit the mapping file.

c If you want to prevent promotion of ACEs, set the value for the `promoteAuthorization` option to `false` in the mapping file. See “Prevent Promotion of Access Controls”.

d If you plan to promote reports that reference stored processes, add an entry to the mapping file that specifies the base URL for the SAS Stored Process Server. See “References to Stored Processes”.
If you prefer to work with mapping files in YAML format, use a third-party utility to convert the ContentMappings.json file to that format.

Place a copy of the revised mapping file in a location that is available to your SAS Viya deployment. If importing using the Import wizard, copy the revised mapping file to a file system that is local to the web browser.

---

### Step 2: Promote Internal Groups

Promote internal (metadata-only) identity groups unless you already have.

1. **Export.**
   
   Locate the identities.spk file that was created in the preceding section. If that file is current, no additional steps are necessary. If that file is not current, repeat the export.

2. **Import.**
   
   a. On the **Content** page in SAS Environment Manager, click 📂, and import using the Import wizard.

      For example, you might refer to and name files as follows:

      | Import file          | identities.spk |
      |----------------------|----------------|
      | Existing mapping file (input) | ContentMappings.json |
      | Saved mapping file (output)    | map_System.json |

      **Note:** To save the mapping file, click 📁 in the mapping step of the Import wizard.

   b. (Optional) Use a text editor to review and refine the mapping file that the import process creates.

3. In the target SAS Viya environment, verify that results are as expected. See “Details: Identities”.
   
   Make post-promotion adjustments as needed.

---

### Step 3: Promote Data Definitions

Promote definitions for libraries and tables.

1. **Export.**
   
   a. In SAS Management Console on the **Folders** tab, right-click **SAS Folders**. Select Export SAS Package.
b On the first page of the Export SAS Package Wizard, provide information as follows:

i Specify a location and name. (These tasks use the name libraries.spk as an example.)

ii If you want to include table definitions for the libraries that you export, select the Include dependent objects when retrieving initial collection of objects check box. (You can instead include table definitions as dependent objects when you export content.)

CAUTION
If you select the Include dependent objects when retrieving initial collection of objects check box, and you have tables and libraries that were in folders in SAS 9 but are not in folders in SAS Viya, extraneous empty folders could get created in the target environment.

iii Click Filter. In the Filter window’s Types section, clear all of the check boxes, and select only the Library type. Click OK.

c On the next page, select the libraries to export.

Note: If you are promoting a LASR library, make sure that you have an authentication domain named LASRAUTH with appropriate credentials. This will enable you to successfully open the LASR caslib after the import.

d When the export is completed, copy the export package file to a location that is available to your SAS Viya deployment.

2 Import.

a On the Content page in SAS Environment Manager, click Import, and import using the Import wizard.

For example, you might refer to and name files as follows:

<table>
<thead>
<tr>
<th>Import file</th>
<th>libraries.spk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Existing mapping file (input)</td>
<td>map_System.json</td>
</tr>
<tr>
<td>Saved mapping file (output)</td>
<td>map_SystemLibraries.json</td>
</tr>
</tbody>
</table>

Note: To save the mapping file, click in the mapping step of the Import wizard.

b (Optional) Use a text editor to review and refine the mapping file that the import process creates.

3 In the target SAS Viya environment, verify that results are as expected. See “Details: Data Definitions” and “Details: Authorizations”. Make post-promotion adjustments as needed.
Step 4: Promote Content

Promote SAS 9 content.

Note: If you promote content that is associated with data stored in libraries, you should provide equivalent data in CAS libraries in your SAS Viya environment before promoting that content. Include mappings that associate the promoted content with the equivalent data.

1 Export.
   a In SAS Management Console on the Folders tab, right-click SAS Folders. Select Export SAS Package.
   b On the first page of the Export SAS Package Wizard, provide information as follows:
      i Specify a location and name. (These tasks use the name content.spk as an example.)
      ii Select the Include dependent objects when retrieving initial collection of objects check box.
      iii If you want to filter what is promoted, click Filter. Make selections in the Filter window. Click OK.
   c On the next page, select the objects to export.
   d When the export is completed, copy the export package file to a location that is available to your SAS Viya deployment.

2 Import.
   Note: You cannot choose the folder in which imported content is placed.
   a On the Content page in SAS Environment Manager, click \( \mathcal{V} \), and import using the Import wizard. For more information, see “Import SAS 9 Resources”.
   For example, you might refer to and name files as follows:

<table>
<thead>
<tr>
<th>Import file</th>
<th>content.spk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Existing mapping file (input)</td>
<td>map_SystemLibraries.json</td>
</tr>
<tr>
<td>Saved mapping file (output)</td>
<td>map_SystemLibrariesContent.json</td>
</tr>
</tbody>
</table>

   b (Optional) Use a text editor to review and refine the mapping file that the import process creates.

3 If you promoted reports that reference stored processes, make sure that all required configuration is complete. See “References to Stored Processes”.

4 If you promoted reports that reference custom report themes, see “References to Custom Themes”.

In the target SAS Viya environment, verify that results are as expected. See “Details: Reports”, “Details: Explorations”, “Details: Authorizations”. Make post-promotion adjustments as needed.

See Also
The SAS 9 export tools are documented in “Promotion Tools Overview” in SAS Intelligence Platform: System Administration Guide. Not all SAS 9 export information is relevant for promotion to a SAS Viya target environment.

Promotion from SAS 9: Reference

Participating Resources
The following resources participate in promotion from SAS 9:

- identities
- authorizations
- data definitions
- content
  - folders
  - reports
    - references to stored processes
    - references to custom report themes
  - explorations
  - other supporting resources
    - images
    - graph templates

**IMPORTANT** For any resource that is promoted, the creator of the resource in the source environment might be changed to the person who is performing the promotion in the target environment. This is especially important if authorization settings are promoted because they might become invalid if the resource creator changes.

Details: Identities

**IMPORTANT** Promotion assumes that source and target environments share an LDAP identity store.
Scope and Approach

The SAS Viya approach to identity management differs from the SAS 9 approach. For that reason, promotion excludes a lot of SAS 9 identity metadata. Here are details:

<table>
<thead>
<tr>
<th>Exclusion</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metadata for identities that also exist in LDAP.</td>
<td>SAS Viya does not import and synchronize LDAP identity information. Instead, SAS Viya reads LDAP identity information in place. During promotion, SAS 9 metadata identities that also exist in LDAP are excluded because SAS Viya does not keep its own copy of that information.</td>
</tr>
<tr>
<td>Metadata for roles and capabilities.</td>
<td>SAS Viya does not have roles (except for CAS roles, which are outside the scope of promotion).</td>
</tr>
<tr>
<td>Metadata for outbound authentication.</td>
<td>SAS 9 authentication domains and stored credentials are outside the scope of promotion. However, you can enter authentication domains and stored credentials using SAS Environment Manager.</td>
</tr>
<tr>
<td>Metadata for internal (metadata-only) users.</td>
<td>SAS Viya does not have internal users. Each user must exist in LDAP. Any users who do not have an external identity value in SAS 9 metadata are assumed to be internal (metadata-only users).</td>
</tr>
</tbody>
</table>

To help preserve memberships and convert authorizations, promotion extracts and manipulates SAS 9 identity information as follows:

- Most SAS 9 internal (metadata-only) groups are converted to SAS Viya custom groups. In most cases, membership information is preserved for members that also exist in LDAP. Here are exceptions:
  - Two of the predefined SAS 9 internal groups (SAS General Servers and SAS System Services) are discarded. In SAS Viya, corresponding predefined internal service accounts exist as needed.
  - A few other predefined SAS 9 identities have specialized mappings. See the following section.

  Note: If a custom group already exists in the target environment, it is not re-created. Instead, membership is compared and any missing members are added. Promotion is cumulative; it does not delete custom groups or memberships from the target environment.

- If your SAS 9 and SAS Viya environments do not use the same field as a unique identifier, resulting variations in identity names (or IDs) are resolved so that they match the LDAP field that SAS Viya uses as a unique identifier.

  Note: To map SAS 9 users to LDAP, promotion uses user IDs from logins in the DefaultAuth authentication domain.
Specialized Mappings

Promotion uses the following specialized mappings:

**Table 3  Specialized Mappings (for Predefined Identities)**

<table>
<thead>
<tr>
<th>SAS 9 Metadata Identity</th>
<th>SAS Viya Principal</th>
<th>How the Mapping Is Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUBLIC group</td>
<td>Everyone¹</td>
<td>Authorizations</td>
</tr>
<tr>
<td>SASUSERS group</td>
<td>Authenticated Users</td>
<td>Authorizations</td>
</tr>
<tr>
<td>SAS Administrators group</td>
<td>SASAdministrators</td>
<td>Authorizations, Members</td>
</tr>
<tr>
<td>Visual Data Builder Administrators group</td>
<td>Data Builders</td>
<td>Members</td>
</tr>
<tr>
<td>Visual Analytics Data Administrators group</td>
<td>ApplicationAdministrators</td>
<td>Members</td>
</tr>
</tbody>
</table>

¹ In CAS authorization, there is no Everyone principal, so PUBLIC is mapped to Authenticated Users.

Here are additional details:

- The specialized mappings are implicit. They cannot be customized. They are not reflected in any mapping file.
- Identities that are not listed in the preceding table do not have specialized mappings.

Details: Authorizations

**IMPORTANT**  The SAS Viya approach to authorization is very different from the SAS 9 approach. For that reason, promotion cannot fully replicate your SAS 9 authorizations in SAS Viya. See “Comparison to SAS 9 Authorization” in SAS Viya Administration: Orientation to Authorization. Simple access controls might be promotable to SAS Viya, but you should carefully verify their behavior after promotion. You might need to modify your authorization model to match the access control features in SAS Viya.

Scope and Approach

For most exported objects, any explicit ACEs are extracted from the SAS 9 metadata. Promotion converts most of the extracted ACEs to SAS Viya authorizations.

Here are key points:

- For promoted libraries and tables, all explicit ACEs are converted to CAS access controls. Row-level filters (referred to in SAS 9 as permission conditions) are preserved.
For other promoted objects, only explicit ACE grants are converted to general authorization rules. ACE denials are discarded.

Here are additional details:

- If you promote an object that already exists in the target environment, authorizations are managed as follows:
  - For libraries and folders, no ACEs are promoted. No new CAS access controls are created in the target environment for libraries. No new general authorization rules are created in the target environment for folders.
  - For other objects, ACEs are promoted and merged with any pre-existing authorizations in the target environment.

- In the target environment, authorizations for new caslibs and new top-level folders that are created by promotion include the same authorizations that are automatically generated if those objects are created directly.

- Any access controls that existed in the SAS 9 metadata to protect identity definitions are discarded. Such access controls affect access to the identity definitions only. They do not affect what each identity can see or do.

Permission Mappings

The promotion tools map SAS 9 permissions to SAS Viya permissions as follows:

**Table 4  LASR Libraries**

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ReadMetadata</td>
<td>ReadInfo</td>
</tr>
<tr>
<td>WriteMetadata</td>
<td>AlterCaslib, CreateTable, AlterTable, DropTable, DeleteSource, ManageAccess</td>
</tr>
<tr>
<td>Administer</td>
<td>Promote</td>
</tr>
<tr>
<td>Read</td>
<td>Select, LimitedPromote</td>
</tr>
<tr>
<td>Write</td>
<td>CreateTable, AlterTable, DropTable, DeleteSource, Insert, Update, Delete</td>
</tr>
</tbody>
</table>

**Table 5  Base Libraries**

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ReadMetadata</td>
<td>ReadInfo, Select</td>
</tr>
<tr>
<td>WriteMetadata</td>
<td>AlterCaslib, CreateTable, AlterTable, DropTable, DeleteSource, ManageAccess, Promote, Insert, Update, Delete</td>
</tr>
</tbody>
</table>
### Table 6  Tables (LASR and Base)

<table>
<thead>
<tr>
<th>SAS 9: Tables</th>
<th>SAS Viya: CAS Authorization</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReadMetadata</td>
<td>ReadInfo</td>
</tr>
<tr>
<td>WriteMetadata</td>
<td>CreateTable, AlterTable, DropTable, DeleteSource, ManageAccess</td>
</tr>
<tr>
<td>Read</td>
<td>Select, LimitedPromote</td>
</tr>
<tr>
<td>Write</td>
<td>CreateTable, AlterTable, DropTable, DeleteSource, Insert, Update, Delete</td>
</tr>
</tbody>
</table>

### Table 7  Folders

<table>
<thead>
<tr>
<th>SAS 9: Folders</th>
<th>SAS Viya: General Authorization</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReadMetadata</td>
<td>Read</td>
</tr>
<tr>
<td>WriteMetadata</td>
<td>Update, Delete, Secure</td>
</tr>
<tr>
<td>WriteMemberMetadata</td>
<td>Add, Remove</td>
</tr>
</tbody>
</table>

### Table 8  Other Content

<table>
<thead>
<tr>
<th>SAS 9</th>
<th>SAS Viya: General Authorization</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReadMetadata</td>
<td>Read</td>
</tr>
<tr>
<td>WriteMetadata</td>
<td>Update, Delete, Secure</td>
</tr>
</tbody>
</table>

### Prevent Promotion of Access Controls

To prevent promotion of access controls, use a mapping file that specifies a value of `false` for the `PROMOTEAUTHORIZATION` option.

Here is a YAML example:

```yaml
options:
  promoteAuthorization: false
```

Here is a JSON example:

```json
"options":{
  "promoteAuthorization":false
}
```
Details: Data Definitions

**IMPORTANT** Physical data is not promoted (moved). Promotion assumes that source and target environments access data in a common location or use a shared file system. Metadata is not needed because after a caslib is defined in SAS Viya, it includes the table and metadata from its source. Table promotion is mostly used to promote authorization settings on the table.

You can promote the following types of SAS 9 data definitions:

- LASR libraries, which become LASR caslibs in SAS Viya
- LASR tables
- Base libraries, which become path caslibs in SAS Viya
- Base tables

Promotion of other types of SAS 9 data definitions is not supported.

Here are additional details:

- If a caslib that results from the promotion of a SAS 9 library already exists in the SAS Viya target environment, the caslib will not be promoted.
- If the SAS 9 environment is secure, then you should import the SAS 9 certificate chain to the trustedcerts.pem file on the CAS controller before promoting. For more information, see "Manage Truststores" in Encryption in SAS Viya: Data in Motion.

Details: Explorations

- Promotion converts explorations to reports.
- For each report that is converted from an exploration, the string (Exploration) (or the localized equivalent of that string) is appended to its name. For example, in English locations, promotion of an exploration that is named MyModel creates a report that is named MyModel(Exploration).

**TIP** In the transfer plug-in, the LOCALE global option determines the locale. The default value for that option is en (English).

- Reports that were originally explorations use the default report theme. This can result in color differences in some objects.
Details: Reports

General

- In the mapping file, review the target table name and make any necessary adjustments. The default mapping is as follows:
  - In most cases, the source table name is used as the target table name.
  - If the source table name cannot be used for a particular table, the source table’s metadata object name is used as the target table name.
- In the mapping file, make sure that the source that is named Default specifies target values for at least the caslib and CAS server. This is essential if you use any auto-generated target values without modification.
- Be aware that three-color ramps are converted to two-color ramps. Consequently, you might notice a difference in the color gradients in heat maps. Also, the color gradient in some SAS sample reports look different.

References to Custom Themes

If you promote a report that uses a custom report theme, and that theme does not exist in your target SAS Viya environment, the following result occurs:

- In the target environment, a copy of the default report theme is created. The copied theme’s name (and ID) is in the following format: "Custom_Report_original-theme-name".
- When the promoted report is displayed, it uses the default report theme.

If you want to avoid this result, complete the following steps before you promote the report:

2. Assign each original theme’s name to the corresponding new theme in SAS Viya using the THEMES UPDATE command in the reports plug-in to the sas-admin CLI.

References to Stored Processes

**IMPORTANT** With appropriate configuration, reports in SAS Viya can call stored processes in SAS 9. Or, you can promote stored processes.

1. Before you import reports that reference stored processes, you must manually insert a mapping file entry for the option STOREDPROCESSBASEURL.
   a. Determine the appropriate value for STOREDPROCESSBASEURL.
   i. Log on to SAS Management Console from the SAS 9 environment.
Examine the values on the **Internal Connection** tab and the **External Connection** tab.

- If the values on the **External Connection** tab are different from the **Internal Connection** tab, and the Use internal connection information check box is not selected, then use the values on the **External Connection** tab for the following step.

- If the values on the **External Connection** tab are the same as the **Internal Connection** tab, or the Use internal connection information check box is selected, then use the values on the **Internal Connection** tab for the following step.

Concatenate the following values to compose the URL:

- **Communication Protocol** value
- `://`
- **Host Name** value
- `:`
- **Port Number** value
- **Service** value
- `/do`

Click **Cancel**

In the appropriate mapping file, add the STOREDPROCESSBASEURL option, and specify the URL from the preceding step.

Here is a YAML example:

```yaml
options:
```

Here is a JSON example:

```json
"options":{
}
```

2 Before you use a report that references a stored process, whitelist the SAS Viya deployment.

**Note:** If you do not whitelist the SAS Viya deployment, SAS Viya cannot use the stored processes because the SAS Viya deployment cannot access the SAS 9 system where the stored processes run.

a On the SAS 9 machine where the SAS Stored Process Server runs, in the value for the `sas.web.csrf.referers.knownHosts` property, specify the URL of the SAS Viya deployment.

For instructions, see [Modifying the Whitelist for URLs and HTTP Request Methods](#).

b Restart the SAS Web Server on the SAS 9 middle tier. If you are performing the next step for inline frames, you can wait until after completing that step to restart the SAS Web Server.

3 If you are running SAS 9.4M5 or later, and you want to allow stored processes to be included in an inline frame (IFRAME), you must adjust the security options for the SAS Stored Process web application, as follows:

a From the SAS 9.4 middle-tier machine, make a copy of the `SAS_configuration-directory/Lev'n/Web/WebServer/conf/sas.conf` file to ensure that you have a backup. Then, edit the file as follows.
Comment out this line:

Header set X-Frame-Options SAMEORIGIN

c For the Chrome browser or Firefox browser, add this line:

Header set Content-Security-Policy "frame-ancestors <communication-protocol>://<SAS Viya host name>"

For example, the following line applies to an HTTPS environment with the specified host:

Header set Content-Security-Policy "frame-ancestors https://host.example.com"

d For Internet Explorer, add this line:

Header set X-Frame-Options "ALLOW-FROM <communication-protocol>://<SAS Viya host name>"

For example, the following line applies to an HTTPS environment with the specified host:

Header set X-Frame-Options "ALLOW-FROM https://host.example.com"

e Restart the SAS Web Server on the SAS 9.4 middle tier.

Here are additional details:

- To add a SAS 9 stored process to a SAS Viya report, specify Web Content as the report object in SAS Visual Analytics.
- Pre-existing SAS 9 reports that contain stored processes do not need to be re-created in SAS Viya.

Note: If the SAS Viya machine uses HTTPS and the SAS 9 machine uses HTTP, the stored process might fail. You can load the content anyway using manual steps. This behavior differs between the various browsers.

Workflow Examples

Promotion from SAS 9 exports selected resources from a source SAS 9 environment and imports resources to a target SAS Viya environment.

The following figure shows two workflows for promotion from SAS 9. Remember these important details about the figure:

- The shaded boxes list output from the adjacent activity or command.
- The lowercase verbs are commands in the transfer plug-in.
- Both workflows use the CLI to create an initial mapping file for identity and optional SAS LASR Analytic Server information. The initial mapping file is eventually merged with another mapping file as follows:
  - In the first workflow, you import using the SAS Viya Import wizard. The wizard prompts you for the initial mapping file. The wizard then creates another mapping file during the import process, and merges the two mapping files for you.
  - In the second workflow, you import using the CLI. In the UPLOAD command, you use the --MAPPING option to indicate that you want an additional mapping file to be created. After you run the UPLOAD command, you manually merge the additional mapping file (in this example, uploadMap) with the initial mapping file (ContentMappings). The merged mapping file (in this example, mergedMap) is used as input to the IMPORT command.
In other words, you combine the files as follows:

ContentMappings file + uploadMap file = mergedMap file

- The figure is not exhaustive. Other workflows, and multiple iterations within each step of each workflow, are possible. The workflow can be repeated for additional resources.

Figure 1  Promotion from SAS 9

Promotion within SAS Viya: Tasks

Introduction

Different methods are used to promote SAS content between SAS Viya environments, depending on whether items reside in SAS folders with a corresponding URI or do not reside in SAS folders.
SAS content that resides in a SAS folder with a corresponding URI can be promoted using either SAS Environment Manager or the transfer plug-in to the sas-admin CLI. The transfer plug-in manages both the export and import of SAS content.

Caslibs and their access controls, which do not reside in a SAS folder, cannot be promoted in the usual manner with the transfer plug-in to the sas-admin CLI. Instead, they can be re-created in the target environment using the cas plug-in to the sas-admin CLI.

Resources like custom groups and rules, which do not reside in a SAS folder, can be promoted using the --REQUEST option of the transfer plug-in to the sas-admin CLI. When promoting these types of content, you must manually specify the URI.

Promote Resources That Are Stored in SAS Folders

To promote resources within SAS Viya that reside in SAS folders and have corresponding URIs, you can use either SAS Environment Manager or the transfer plug-in to the sas-admin CLI.

TIP  Note the location of any Model Studio object that you want to promote so that you can specify it during the import.

This method exports the content to a package file, and then imports the package file to the target environment, as follows:

1  Review the information in “Promotion within SAS Viya: Reference”. For successful promotion, you must understand process, scope, and effects.

2  Export content from the source environment to a package file as follows:
   - To export content using SAS Environment Manager, see “Export Content” in SAS Viya Administration: Folders.
   - To export content using the sas-admin CLI:
     - Retrieve the URI of the item that you want to export as follows:
       - Open SAS Environment Manager in the source environment. Select the Content page. In the navigation pane, navigate to each item that you want to export. For each item to be exported, look at the Basic Properties in the right pane, and make a note of the item’s URI and type.
     - Make sure that your sas-admin CLI is set up. If you have a profile and have signed in on the source machine, specify the URL for the source machine as the endpoint in your profile.
     - If you have not created a profile and signed in on the source machine, do the following:
       - Create a profile that references your source SAS machine as the endpoint.
       - Sign in with an account that is a member of the SAS Administrators group in your SAS environment.
     - If you are not already there, navigate to the CLI executable files.
     - The files are on your SAS Viya machine (details). If you downloaded the CLI, the files are in your download location.
     - Run the sas-admin transfer export command to export the items to a package file.
If you are exporting only one item, you can specify its URI on the command line by using the `--RESOURCE-URI` option. To export multiple items, specify the items in a JSON file, and use the `--REQUEST` option to specify the JSON file.

For example, the following command exports a single report from a Test environment (that is, the SAS Viya environment that is specified in a profile called Test):

```
sas-admin transfer --profile Test export
--resource-uri /reports/reports/faa7f5f2-0822-4ca0-9f92-23bda3e02738
--name "Export Report"
```

The ID of the created package file is displayed at the end of the output. Make a note of (or copy) the package ID. You will need it in the next step. Here is an example:

```
Package created: faa7f5f2-0822-4ca0-9f92-23bda3e02738
```

Run the SAS-ADMIN TRANSFER DOWNLOAD command to download the exported package file to a JSON file on your local machine.

For example, the following command downloads the package file from the test environment in the prior step and stores it in a package file called MyPackage.json on your local machine. If you do not specify a path, the JSON file is created in the directory where the command was executed.

```
sas-admin transfer --profile Test download
--id faa7f5f2-0822-4ca0-9f92-23bda3e02738 --file MyPackage.json
```

3. Save the package files in a location that is accessible to the SAS Viya deployment.

4. Import using the wizard or the command-line interface.

Note: Model Studio project owners must sign in to the target system before any of their projects can be promoted. See “Import Content” for other information that you might need to know before you import.

5. In the target environment, verify results, and make any necessary post-promotion adjustments.

See Also
The transfer plug-in

---

Promote Caslibs and Access Controls

To promote caslibs and access controls within SAS Viya, you can use the cas plug-in to the sas-admin CLI to create a JSON file for caslibs and access controls from the source environment. Then, use the cas plug-in to the sas-admin CLI again to create caslibs and access controls from that same JSON file in the target environment. See the following:

1. Review the information in “Promotion within SAS Viya: Reference”. For successful promotion, you must understand process, scope, and effects.

2. Create a JSON file for caslibs and access controls from the source environment as follows:
   - Make sure that your sas-admin CLI is set up. If you have a profile and have signed in on the source machine, specify the URL for the source machine as the endpoint in your profile.
   - If you have not created a profile and signed in on the source machine, do the following:
Create a profile that references the URL for the source SAS Viya environment as the endpoint.

Sign in with an account that is a member of the SAS Administrators group in your SAS environment.

If you are not already there, navigate to the CLI executable files.

TIP: The files are on your SAS Viya machine (details). If you downloaded the CLI, the files are in your download location.

Run the SAS-ADMIN CAS CASLIBS LIST command to view information about the existing caslibs. Identify the caslib that you want to export.

For example, the following command displays caslib information in the command line. The --limit option can be adjusted to show more caslibs.

```
sas-admin --output text cas caslibs list --server serverA --limit 20
```

Run the SAS-ADMIN CAS CASLIBS SHOW-INFO command to display detailed information about the caslib that you want to export.

For example, the following command displays information about the Sales caslib:

```
sas-admin --output text cas caslibs show-info --server serverA --name Sales
```

Redirect the information about the caslib to a file in JSON format. This file is a backup of the caslib information from the source environment that can be copied to the target environment.

For example, the following command redirects information about the Sales caslib to a JSON file called salescaslib.json:

```
sas-admin --output json cas caslibs show-info --name Sales --server serverA > /tmp/salescaslib.json
```

For example, the contents of the JSON file might look like this:

```
# more /tmp/salescaslib.json
{
  "attributes": {
    "active": false,
    "personal": false,
    "subDirs": false,
  },
  "description": ",
  "name": "Sales",
  "path": "/tmp/salescaslib/",
  "scope": "global",
  "server": "serverA",
  "type": "PATH"
}
```

Note: Notice the PATH value in the JSON file. You will use the PATH value when creating the caslib in the target environment. In this example, you are creating a path-based caslib in the target environment.

To preserve access controls (caslib authorizations), run the SAS-ADMIN CAS CASLIBS LIST-CONTROLS command.
For example, the following command redirects the caslib authorizations from the Sales caslib to a JSON file called salescasauths.json:

```
sas-admin --output json cas caslibs list-controls --server serverA --name Sales > /tmp/salescasauths.json
```

**Note:** Authorizations are not included when the caslib information is retrieved from the source environment. When the caslib is created in the target environment, it gets the same default permissions that it would have received if it had been created with SAS Environment Manager. Other authorizations are lost. Therefore, you must save the authorizations to a JSON file from the source environment and import them into the target environment to preserve them.

3. Copy data to the target environment or, if using external data sources, make sure that the data source is accessible from the target environment.

4. Re-create content in the target environment as follows:

   - Make sure that your sas-admin CLI is set up. If you have a profile and have signed in on the target machine, specify the URL for the target machine as the endpoint in your profile.
   - If you have not created a profile and signed in on the target machine, do the following:
     - Create a profile that references your target SAS machine as the endpoint.
     - Sign in with an account that is a member of the SAS Administrators group in your SAS environment.
   - If you are not already there, navigate to the CLI executable files.

   **TIP** The files are on your SAS Viya machine (details). If you downloaded the CLI, the files are in your download location.

   - Run the SAS-ADMIN CAS CASLIBS CREATE PATH command to create a path-based caslib in the target environment using the information stored in the JSON file that you created in the source environment.

     For example, the following command creates the Sales caslib in the target environment:

     ```
sas-admin cas caslibs create path --source-file /tmp/salescaslib.json
```

     If the caslib already exists in the target environment, it will not be re-created. If the path does not exist in the target environment, it can be manually created prior to running the import or the --CREATE-DIR option can be added to the command to automatically create it. Keep in mind that the --CREATE-DIR option does not create subdirectories.

   - To re-create access controls (caslib authorizations), run the SAS-ADMIN CAS CASLIBS REPLACE-CONTROLS command to overwrite the access controls for the Sales caslib with those in the salescasauths.json file.

     For example, the following command replaces the access controls for the Sales caslib with those in the salescasauths.json file:

     ```
sas-admin --output json cas caslibs replace-controls --server serverA --name Sales --source-file /tmp/salescasauths.json
```

5. In the target environment, verify results and make any necessary post-promotion adjustments.
The transfer plug-in

Promote Resources That Are Not Stored in Folders

To promote resources that do not reside in folders such as groups and rules, you can use the transfer plug-in to the sas-admin CLI to promote resources using resource URIs. Specifically, you can create a JSON file that identifies the URIs of the items to export, use the transfer plug-in to export the JSON file, and then use it again to import the resulting package file to the target environment. Here are the steps:

1 Review the information in "Promotion within SAS Viya: Reference". For successful promotion, you must understand process, scope, and effects.

2 Export content from the source environment to a package file as follows:
   - Make sure that your sas-admin CLI is set up. If you have a profile and have signed in on the source machine, specify the URL for the source machine as the endpoint in your profile.
   - If you have not created a profile and signed in on the source machine, do the following:
     - Create a profile that references your source SAS machine as the endpoint.
     - Sign in with an account that is a member of the SAS Administrators group in your SAS environment.
     - If you are not already there, navigate to the CLI executable files.
       - Run the SAS-ADMIN IDENTITIES LIST-GROUPS command to display information about the existing groups. Identify the group that you want to export, and record the ID of the group. You will use the ID to construct the URI of the group.
         - For example, the following command displays information about existing groups:
           ```
sas-admin --output text identities list-groups
           ```
         - If you are exporting a general authorization rule that is associated with the group, retrieve the ID from SAS Environment Manager as follows:
           - Open SAS Environment Manager in the source environment. In the side menu, under SAS Environment Manager, select
           - Using the Rules Filter, locate the rule that you want to export. From the table on the right, right-click the rule, and select Properties. The ID of the rule is the last item in the Rule Properties window. You will use the ID to construct the URI of the rule.
           - Create the JSON file that defines what is being exported. This JSON file is used as input to the transfer plug-in using the --REQUEST option. The format of the JSON file is documented in the transfer plug-in.
             - The ITEMS option in the JSON file precedes the list of URIs, enumerating all resources that are being exported. The IDs of the group and authorization rule that you recorded in prior steps are the last part of their respective URIs.
             - The first part of the URI is the REST endpoint of the resource, as follows:
If you are exporting something other than a group or role and you do not know the REST endpoint of the resource, you can find information at developer.sas.com or contact SAS Technical Support for assistance.

Here is an example of a JSON file. The ITEMS option precedes the URIs for a group and an authorization rule.

```json
{
  "version": 1,
  "name": "My custom groups",
  "description": "Export Viya Custom groups and Rule",
  "items": [
    "/identities/groups/Finance Content Developers",
    "/authorization/rules/00b08a2f-847f-4c2a-98b5-e1711860d02d"
  ]
}
```

The URIs in the JSON file consist of the following parts:

<table>
<thead>
<tr>
<th>REST Endpoint</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>/identities/groups/</td>
<td>Finance Content Developers</td>
</tr>
<tr>
<td>/authorization/rules/</td>
<td>00b08a2f-847f-4c2a-98b5-e1711860d02d</td>
</tr>
</tbody>
</table>

- Run the SAS-ADMIN TRANSFER EXPORT command with the --REQUEST option to export the resources enumerated in the JSON file that you created.

  For example, the following command exports the resources enumerated in the export_groups.json file in the /tmp/ directory.

  ```bash
  sas-admin transfer export --request @/tmp/export_groups.json
  ```

  The first character following the --REQUEST option must be the @ sign. See the transfer plug-in for more information.

  **Note:** Note the ID of the package that is returned with the output of the command.

  3 Run the SAS-ADMIN TRANSFER DOWNLOAD command to download the package to a JSON file that you can import to your target environment. Use the ID of the package from the prior step as input to the command.

  For example, the following command downloads the package with the specified ID to the package_groups.json file:

  ```bash
  sas-admin transfer download --id 2916d2cb-14bd-4418-9aff-2d90feb4397e --file /tmp/package_groups.json
  ```

  4 Copy the JSON file to the target environment.
To import the package with SAS Environment Manager, see "Import SAS Viya Resources". If a group with the same name already exists in the target environment, it is not imported. If a rule with the same ID exists, then the contents of the rule are updated.

To import the package with the transfer plug-in, you must first delete any group that already exists in the target environment with the same name.

- Remove a group with the same name with the SAS-ADMIN IDENTITIES DELETE-GROUP command.
- Run the SAS-ADMIN TRANSFER UPLOAD command to upload the JSON file to the target environment.

For example, the following command uploads the package_groups.json file:

```
sas-admin transfer upload --file /tmp/package_groups.json
```

**Note:** Note the ID of the package that is returned with the output of the command.

- Run the SAS-ADMIN TRANSFER IMPORT command to import the package to the target environment. Use the ID of the package from the prior step as input to the command.

For example, the following command imports the package with the specified ID:

```
sas-admin transfer import --id id
```

In the target environment, verify results and make any necessary post-promotion adjustments.

---

**Promotion within SAS Viya: Reference**

**Participating Resources: Promotable with the Transfer Plug-in**

The following resources participate in promotion in SAS Viya by using the transfer plug-in:

- comments
- data views
- files
- folders
- graph templates
- images
- jobs
  - job requests
  - job definitions
  - job flows
  - job actions
Note: The Schedule service schedules job requests, but the promotion of schedules is not currently supported.

- Model Studio projects
- reports
- SAS Data Studio data plans
- statistical models
- themes

**IMPORTANT** For any resource that is promoted, the creator of the resource in the source environment might be changed to the person who is performing the promotion in the target environment. This is especially important if authorization settings are promoted because they might become invalid if the resource creator changes.

Other Participating Resources

The following resources participate in promotion within SAS Viya by using other relevant CLIs:

- authorizations (CAS access controls on promoted caslibs and tables; direct rules on other promoted objects)
- data sources

**IMPORTANT** For any resource that is promoted, the creator of the resource in the source environment might be changed to the person who is performing the promotion in the target environment. This is especially important if authorization settings are promoted because they might become invalid if the resource creator changes.

Details: Reports That Contain ASTORE Tables

Some reports contain a new type of table called an analytic store (ASTORE) table. ASTORE tables are created internally. They are different from the source tables that users include in a mapping file before importing. Currently, tables cannot be promoted. Consequently, ASTORE tables are not transferred to the target environment during promotion.

When you open a report that was just promoted, and the report contains an ASTORE table, you receive a message about a missing data source. The message is accompanied by the name of the ASTORE table that is missing.

To resolve this, you must manually copy the ASTORE table from the source machine to the target machine.

1. Connect to each source machine with SSH (secure shell), and search for this directory:

   `/opt/sas/viya_or_tenant/config/data/cas/default/vamodels`
The value of `viya_or_tenant` is either `viya`, which is the default, or is the name of the tenant if this is a multi-tenant deployment.

**TIP** If you have a multi-host or distributed deployment, check for this directory on each host and each deployment, starting with the SAS Cloud Analytic Services controller first.

2 Locate the missing `ASTORE` table.

3 Copy the `ASTORE` table to the target machine as follows:

   ```bash
   scp -p /opt/sas/viya_or_tenant/config/data/cas/default/vamodels/ASTORE_table
   target_machine:/opt/sas/viya_or_tenant/config/data/cas/default/vamodels/
   ```

4 Open and verify the report in SAS Visual Analytics.

5 Repeat steps 1 through 3 for additional missing `ASTORE` tables.

---

**Details: Jobs**

Job promotion can involve the promotion of a job request, a job definition, and a job schedule.

If the job object (request, definition, or schedule) is in a SAS Environment Manager folder, you can promote it using SAS Environment Manager.

If the job object is not in a SAS Environment Manager folder, it is promoted as follows:

1 Make sure that your `sas-admin` CLI is set up. If you have a profile and have signed in on the source machine, specify the URL for the source machine as the endpoint in your profile. Sign in with an account that is a member of the SAS Administrators group in your SAS environment.

   If you have not created a profile and signed in on the source machine, do the following:
   - **Create a profile** that references your source SAS machine as the endpoint.
   - **Sign in** with an account that is a member of the SAS Administrators group in your SAS environment.

2 Retrieve the URI of the job definition. Use the `JOB DEFINITIONS LIST` command to retrieve a list of all job definitions. Locate the ID of the job definition for which you need the job definition URI. Use the `JOB DEFINITIONS SHOW --ID` command to retrieve the job definition URI for the job with the associated ID.

   ```bash
   sas-admin job definitions list
   sas-admin job definitions show --id job-id
   ```

3 You must manually build the job request URI. The format of the job request URI is as follows:

   `/jobExecution/jobRequests/job-request-ID`

   You can retrieve the job request ID by examining the properties of the job in SAS Environment Manager or by using the job plug-in to the `sas-admin` CLI.

   Using SAS Environment Manager:
   - From SAS Environment Manager, click the **Jobs** page.
   - Click the **Scheduling** tab.
c Right-click on the job for which you need the job request ID, and select **Properties**. The ID on the **General** tab of the **Job Properties** dialog box is the job request ID.

Using the job plug-in:

a Create your profile, and sign in to SAS Viya.

b Issue the following command: `sas-admin job requests list`

c Locate the job request ID from the list of jobs.

See [SAS Note 63592](#) for more information.

4 Use the URI of the job definition and the URI of the job request to **export** content to a package file using the transfer plug-in to the sas-admin CLI.

5 **Import** the resulting JSON package with the transfer plug-in to the sas-admin CLI.

### Details: Themes

Theme promotion involves the promotion of the individual theme as well as associated graphic files and animated graphic files (logo, initial loading, and transition). SAS recommends that you use SAS Theme Designer to promote themes. See “Exporting and Importing a SAS Theme” in *SAS Theme Designer: User’s Guide* for more information.

### Additional Details

#### Promote Content

<table>
<thead>
<tr>
<th>Action</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Promote a folder.</td>
<td>Any supported objects that the folder contains (including reports, comments, files, and subfolders) are automatically included in the promotion. Or, you can select one or more objects within the folder to promote (for example, a report).</td>
</tr>
<tr>
<td>Action</td>
<td>Result</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Promote objects that are not stored in folders.</td>
<td>Objects that are not stored in folders do not surface when performing an export in SAS Environment Manager. These objects can be promoted with the transfer plug-in as follows:</td>
</tr>
<tr>
<td></td>
<td>- Retrieve the URI of the object. If there is an associated CLI (for example, identities or authorizations), you can use the CLI to retrieve the URI.</td>
</tr>
<tr>
<td></td>
<td>- Use the URI to export the object with the transfer plug-in.</td>
</tr>
<tr>
<td></td>
<td>- Import the resulting JSON package with the transfer plug-in.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong>: You can export the object with the transfer plug-in, and then import it with the Import wizard.</td>
</tr>
<tr>
<td></td>
<td>See <a href="#">jobs, caslibs and access controls</a>, and custom groups for more information about promoting items that are not stored in folders.</td>
</tr>
<tr>
<td>Promote from SAS Viya to SAS Viya.</td>
<td>When promoting with the transfer plug-in, general authorization rules are included as part of the export package and applied to objects when imported. If a resource already exists in the target environment, the target resource is updated and authorizations are merged.</td>
</tr>
<tr>
<td>Perform an upgrade or a promotion of a report from an earlier release of SAS Visual Analytics to the current release.</td>
<td>The report is converted every time you access it.</td>
</tr>
<tr>
<td></td>
<td>In some cases, this might cause a delay when opening the report in the current release of SAS Visual Analytics. In other cases, the information bar of the report does not retain all content when printing the report. To prevent a delay in the future, follow these steps:</td>
</tr>
<tr>
<td></td>
<td>1. Open the report in the current release of SAS Visual Analytics.</td>
</tr>
<tr>
<td></td>
<td>2. Save the report in the current release of SAS Visual Analytics. The report is saved in the current release’s format, which eliminates the need to run the conversion process in the future.</td>
</tr>
</tbody>
</table>

### Export Content

<table>
<thead>
<tr>
<th>Action</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Export a folder in SAS Visual Analytics.</td>
<td>In SAS Visual Analytics, export must be supported by all members of a folder in order to export the folder. To work around this, you must individually export the items in the folder for which export is supported.</td>
</tr>
<tr>
<td>Action</td>
<td>Result</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Export or import an object, and then rename the exported or imported</td>
<td>Subsequent exports or imports of an object overwrite the renamed object and replace it with the original object. The renamed object gets overwritten because it has the same URI as the original object. An object’s URI is key in an export or import.</td>
</tr>
<tr>
<td>object.</td>
<td></td>
</tr>
</tbody>
</table>

**Import Content**

<table>
<thead>
<tr>
<th>Action</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import a Model Studio project.</td>
<td>Model Studio project owners must sign in to the target system before any of their projects can be promoted.</td>
</tr>
<tr>
<td>Import any object that has been exported with the transfer plug-in or</td>
<td>The Import wizard can import any object that has been exported with the transfer plug-in or with SAS Environment Manager.</td>
</tr>
<tr>
<td>with SAS Environment Manager.</td>
<td></td>
</tr>
<tr>
<td>Import a folder when another folder with the same name already exists.</td>
<td>When you import a folder, an underlying request is issued to create the folder. It is possible that a request to create a folder with the same name can occur. When this happens, you must address the duplicate folder issue, and then retry the import.</td>
</tr>
<tr>
<td>Import an object into a folder where there is an existing object with</td>
<td>Two objects with the same name but different URIs might result.</td>
</tr>
<tr>
<td>the same name but a different URI.</td>
<td></td>
</tr>
</tbody>
</table>

**Process Flow Examples**

Promotion within SAS Viya extracts certain content and metadata from the source SAS Viya environment and loads them to a target SAS Viya environment. For example, you might use this type of promotion to copy content from a test environment to a production environment.

Here are two workflows for promotion within SAS Viya:
Here are details about the preceding figure:

- The shaded boxes list artifacts of the adjacent activity or command.
- The lowercase verbs are commands in the transfer plug-in.
- If you use the wizard, the package file is exported to the download location on the source machine. Before you import the package file, you must copy it to a location that is available on the target machine.
- You repeat the entire workflow for any content that you are promoting.

Additional Documentation

For additional promotion documentation related to specific products, see the following:

<table>
<thead>
<tr>
<th>Product</th>
<th>Promotion Documentation</th>
</tr>
</thead>
</table>
| Model Studio                   | “Promotions and Upgrades within SAS Viya” in SAS Visual Data Mining and Machine Learning: Advanced Topics  
Promotion: Import Using the Import Wizard

The import phase of the promotion process can be performed using the Import wizard in SAS Environment Manager 3.3 or later.

Introduction

The Import wizard contains the following steps:

- Select the source file that contains the content to be promoted.
- Map content. You can do one of the following:
  - Create a new mapping file. Select target servers, target caslibs, and target tables.
  - Use an existing mapping file.
  - Use an existing mapping file, and then select different target servers, target caslibs, and target tables.

If you are importing an SPK file that contains substitution properties, change substitution property mappings if needed.

Note: If you are using an existing mapping file and making changes, then you must choose the existing mapping file first, and then make changes. If you choose the existing mapping file after you make changes, the changes to the existing mappings are erased.

- Import the content into the target environment.

Note: You must close the Import wizard before attempting an additional import.

When you promote content with the Import wizard, the ensuing exchange depends on whether you are importing an SPK or JSON package file. Certain features in the wizard such as substitutions and access controls apply only to SPK files. Based on the type of package file, the wizard determines which features apply, and it surfaces only the relevant features.
Import SAS 9 Resources

**IMPORTANT** Order matters. Import resources in the following order: first groups, then libraries, then tables, and then reports (explorations).

1. Make sure that the exported packages and initial mappings are available. See “Promotion from SAS 9: Tasks”.

2. On the Content page in SAS Environment Manager, click \(|\) and import using the Import wizard.

3. In the Import file field, click \(\) to navigate in your file system to the SPK file that you want to import. Select it.

   If this is the first time you have imported content, the first package that you need to import is the identities package file that you used as input to the SAS-ADMIN TRANSFER GENERATE-CONTENT-MAPPING command in a preliminary step.

   Click Open.

   The content in the source file is compared to the content in the target environment, and package content is displayed in the tree on the left. If any objects in the source file already exist in the target environment, they will be displayed with \(\)\(\)\(\). You can select an object from the source file to display its properties.

4. At the top of the window, click Mapping to proceed to the mapping step. The ensuing exchange depends on the contents of the SPK file. If the file contains substitution properties, then the Substitution Properties view is displayed and populated with values. If the file contains table mappings, then the Table Mappings view is displayed and populated with values. If the file contains additional mappings, then the Additional Mappings view is displayed and populated with values. Any views for which there are no applicable values in the SPK file are grayed out.

   **Note:** If nothing is displayed after clicking Mapping, then the package might not be fully unpacked. To resolve this, click Select source again, and then click Mapping.

   If table mappings are available, the table is populated as follows:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Mappings are generated for all objects.</td>
</tr>
<tr>
<td></td>
<td>() precedes libraries with missing tables.</td>
</tr>
<tr>
<td>Mapping</td>
<td>() indicates new mappings.</td>
</tr>
<tr>
<td>Exists</td>
<td>() indicates that target server, caslib, and table exist.</td>
</tr>
<tr>
<td></td>
<td>() indicates that one or more target items do not exist.</td>
</tr>
</tbody>
</table>

   If you want to use an existing mapping file, click \(\) in Combine with an existing mapping file to navigate in your file system to an existing mapping file.
If this is the first time you have imported content, you should specify the ContentMappings.json file that was created as a result of running the SAS-ADMIN TRANSFER GENERATE-CONTENT-MAPPING command in a preliminary step.

If you add a mapping file, ☐ in the Mapping column indicates mappings from the mapping file. You can change mappings by clicking on a target item, and then choosing a different target from the drop-down list.

- If substitution properties are available, mappings are generated for all objects in the source file by default. You can change substitutions by clicking on values in the New Value column.
- If additional mappings are available, mappings are generated for all objects in the source file by default. You can change mappings by clicking on a target item, and then choosing a different target from the drop-down list.
- Click to save the mapping values. The Save Mapping File window appears. The name of the resulting mapping file defaults to the name of the package file. If you combined it with an existing mapping file, the name defaults to the name of the mapping file that you combined it with. The mapping file is saved to the download location specified in your browser settings.

5 At the top of the window, click Finish to complete the promotion.

Click Import at the top right of the window. If every object was successfully mapped, then the Results table is displayed by default with a list of the objects that were imported. If any objects were not mapped, then the Reminders tab shows the list of unmapped objects appears by default.

6 Click Close at the top right of the window. If you have not yet imported or saved the mappings from the previous step, then you will be prompted to save the mappings.

---

**Note:** The mapping file is saved by default to the download location specified in your browser settings. If you want the mapping file to be saved in a different location, you must manually move it after you exit the Import wizard.

---

**See Also**

"Participating Resources"

---

**Import SAS Viya Resources**

1 Complete the prerequisite steps in “Promotion within SAS Viya: Tasks” to create SAS Viya package files. Then, SAS recommends that you promote libraries first, followed by reports.

2 On the Content page in SAS Environment Manager, click ☑, and import using the Import wizard.

3 In the Import file field, click ☐ to navigate in your file system to the JSON file that you want to import.

   Click Open.

   The content in the source file is compared to the content in the target environment, and the package content is displayed in the tree on the left. If any objects in the source file already exist in the target environment, they will be displayed with ☑. You can select an object from the source file to display its properties and dependences. If the Dependencies heading is preceded by ☐, then one or more of the dependencies do not exist in the target environment. You can replace the references to the missing dependencies in the mapping step.
4 At the top of the window, click **Mapping** to proceed to the mapping step. The ensuing exchange depends on the contents of the JSON file. If the file contains table mappings, then the **Table Mappings** view is displayed and populated with values. If the file contains additional mappings, then the **Additional Mappings** view is displayed and populated with values. Any views for which there are no applicable values in the JSON file are grayed out.

Note: If nothing is displayed after clicking **Mapping**, then the package might not be fully unpacked. To resolve this, click **Select source** again, and then click **Mapping**.

- If table mappings are available, the table is populated as follows:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
</table>
| Source  | Mappings are generated for all objects.  
          | □ precedes libraries with missing tables. |
| Mapping | ◊ indicates new mappings. |
| Exists  | ○ indicates that target server, caslib, and table exist.  
          | □ indicates that one or more target objects do not exist. |

If you want to use an existing mapping file, click [□] in **Combine with an existing mapping file** to navigate in your file system to an existing mapping file.

If you add a mapping file, [□] in the **Mapping** column indicates mappings from the mapping file. The name of the resulting mapping file in **Save** changes to mapping-file-name.

You can change mappings by clicking on a target item, and then choosing a different target from the drop-down list.

Note: The import process proceeds even if there are missing dependencies in the target environment. You can remediate anything that fails to import at a later time.

- If additional mappings are available, mappings are generated for all objects in the source file by default. You can change mappings by clicking on a target item, and then choosing a different target from the drop-down list.

- Click [□] to save the mapping values. The Save Mapping File window appears. The name of the resulting mapping file defaults to the name of the package file. If you combined it with an existing mapping file, the name defaults to the name of the mapping file that you combined it with. The mapping file is saved to the download location specified in your browser settings.

5 At the top of the window, click **Finish** to complete the promotion.

Click **Import** at the top right of the window. If every object was successfully mapped, then the **Results** table is displayed by default with a list of the objects that were imported. If any objects were not mapped, then the **Reminders** tab shows the list of unmapped objects.

6 Click **Close** at the top right of the window. If you have not yet imported or saved the mappings from the previous step, then you will be prompted to save the mappings.
Note: The mapping file is saved by default to the download location specified in your browser settings. If you want the mapping file to be saved in a different location, you must manually move it after you exit the Import wizard.

See Also

“Participating Resources: Promotable with the Transfer Plug-in”

Promotion: Import Using the Command-Line Interface

In SAS Viya, the import phase of the promotion process can be performed using the transfer plug-in to the sas-admin CLI.

Prerequisites

Note: You must have administrator privileges to promote data. For details and instructions, see SAS Viya Administration: Identity Management.

Before using the transfer commands to perform imports or exports, do the following:

1. Familiarize yourself with the general command-line interface documentation.
2. Follow the instructions to set up your environment.

Note: Use the Help from within the transfer plug-in to learn about the available commands, subcommands, and options. For more information, see “Integrated Help” in SAS Viya Administration: Using the Command-Line Interfaces.

See Also

- “Administrative CLI Plug-Ins” in SAS Viya Administration: Using the Command-Line Interfaces
- “Command-Line Interface: Preliminary Instructions” in SAS Viya Administration: Using the Command-Line Interfaces

Import SAS 9 Resources

IMPORTANT  Order matters. Import resources in the following order: first groups, then libraries, then tables, and then reports (explorations).
1. Make sure that the exported packages and initial mappings are available. See "Promotion from SAS 9: Tasks".

2. From the target machine, make sure that you have completed the prerequisite steps to use the CLI. See "Command-Line Interface: Preliminary Instructions" in SAS Viya Administration: Using the Command-Line Interfaces.

3. Navigate to the CLI executable files.

```
TIP The files are on your SAS Viya machine (details). If you downloaded the CLI, the files are in your download location.
```

4. Run the SAS-ADMIN TRANSFER UPLOAD command to upload the SAS 9 package file from your local machine to the target SAS Viya environment.

For example, the following command uploads a SAS 9 package file called myPackage.spk. Because the --PROFILE option is not specified and the SAS_CLI_PROFILE environment is not set, the file is uploaded to the SAS Viya environment that is specified in your default profile. The --MAPPING option creates a mapping file that you can edit to specify substitution values for data libraries and other connections. The file is written in YAML format.

```
sas-admin transfer upload --file MyPackage.spk --mapping MyPackageMapping.yml
```

```
TIP For those familiar with administering SAS 9, the SAS Viya mapping file is analogous to the SAS 9 substitution properties file.
```

5. Open the mapping file using a text editor, and make changes as needed. If no path was specified in the --MAPPING option, the file is located in the directory where the UPLOAD command was executed.

- If this is the first time you have promoted content, you must combine the initial mapping file (the ContentMappings.json file that you previously created using the SAS-ADMIN TRANSFER GENERATE-CONTENT-MAPPING command) with the current mapping file. You can do this manually or you can combine them with the Import wizard.

- If you have only tables or both libraries and tables in your SPK package, then you must manually add mappings to import ACEs to SAS Viya. When using the Import wizard, the individual table mappings are generated automatically.

To promote a package file that contains reports and explorations, on the target: lines in the mapping file, enter the table name, caslib, and server name where each table will reside in SAS Viya. An example of such a mapping file in JSON format is shown below:

```
{
   "version": 1,
   "connectors": {
      "table": [
         {
            "resourcename": "",
            "source": "default",
            "target": ",HPS,cas-shared-mpp"
         },
         {
            "resourcename": "",
            "source": "\<path_to_table\>/ORION_STAR_SCHEMA(Table)",
            "target": "ORION_STAR_SCHEMA"
         }
      ]
   }
}
```
Note:

- If you are promoting Base libraries, you must add some code to the `substitutions` section of the mapping file. Add the `name:` line with the value of `CASServer` and the `value:` line with the name of your specific CAS server.

Here is a JSON example of the `substitutions` section of a mapping file with these values added:

```json
{
   "name": "CASServer",
   "value": "CAS-server-name"
}
```

When promoting Base libraries with the Import wizard, the `name:` and `value:` lines are automatically added to the mapping file.

- The information about the `target:` line below the `source:` `default` line applies to any other table in the file for which one or more of the values were omitted from the `target:` line. For more information about the mapping file, see "Mapping Files".

6. Use the SAS-ADMIN TRANSFER LIST command to obtain the package ID of the package that you just uploaded. You can also obtain the package ID of the package from the output of the SAS-ADMIN TRANSFER UPLOAD command from the preceding step.

7. Run the SAS-ADMIN TRANSFER IMPORT command to import the package to SAS Viya. Here is an example:

   ```bash
   sas-admin transfer import --id 9d4ca052-1788-4e68-b558-a29069d22451
   --mapping MyPackageMapping.yml
   ```

Import SAS Viya Resources

**IMPORTANT** Order matters. Import resources in the following order: first libraries and then other content (reports).
Follow these steps to promote content from SAS Visual Analytics 8.x to the current release:

1. Make sure that the exported packages and initial mappings are available. See “Promotion within SAS Viya: Tasks”.

2. From the target machine, make sure that you have completed the prerequisite steps to use the CLI. See “Command-Line Interface: Preliminary Instructions” in SAS Viya Administration: Using the Command-Line Interfaces.

3. Navigate to the CLI executable files.

   **TIP** The files are on your SAS Viya machine (details). If you downloaded the CLI, the files are in your download location.

4. Run the SAS-ADMIN TRANSFER UPLOAD command to upload the package file to the target SAS Viya environment. Make sure that the profile specifies the correct host information for the target SAS Viya environment.

   For example, the following command uploads a package file called VAcontent.json to the target SAS Viya environment (that is, the SAS Viya environment that is specified in the profile called Prod).

   ```
   sas-admin transfer --profile Prod upload --file VAcontent.json
   ```

   **Note:** You can use the --MAPPING option to create a mapping file that you can edit to specify substitution values.

   The ID of the created package file is displayed at the end of the output. Make a note of (or copy) the package ID. You will need it in the next step.

5. Run the SAS-ADMIN TRANSFER IMPORT command to import the uploaded package to the SAS Viya target environment. Make sure that the profile specifies the correct host information for the target SAS Viya environment. Specify the package ID.

   For example, the following command imports the package file that was uploaded in the previous step to the target SAS Viya environment that was specified in the profile named Prod:

   ```
   sas-admin transfer --profile Prod import --id 9d4ca052-1788-4e68-b558-a29069d22451
   ```

   **Note:** You can use the --MAPPING option to specify the mapping file containing substitution values that you created in the upload step.

6. To see the uploaded SAS Visual Analytics content, open SAS Environment Manager in the target SAS Viya environment. In the side menu ☰, and select Manage Environment ➜ Content.

See Also

- The transfer plug-in
- “Mapping Files” on page 44
- “Substitution Properties” on page 47
Check Logs

The transfer plug-in to the sas-admin CLI interacts with many SAS Viya services. To view log messages from the transfer plug-in, you need to check the service logs of the SAS Viya services. For example, /opt/sas/viya/config/var/log/import9 contains SAS 9 import logs and /opt/sas/viya/config/var/log/transfer contains transfer plug-in logs. For more information, see SAS Viya Administration: Logging.

Promotion: How To (CLI)

The following examples assume that you have already signed in to SAS Viya at the command line. See "Command-Line Interface: Preliminary Instructions" in SAS Viya Administration: Using the Command-Line Interfaces.

CLI Examples: Transfer

The following examples are for the transfer plug-in to the sas-admin CLI.

Examples

**Example:** See the commands and subcommands that are available for the TRANSFER IMPORT command.

```
sas-admin transfer import --help
```

**Example:** Export a new transfer package from an object that has the resource URI /reports/reports/faa7f5f2-0822-4ca0-9f92-23bda3e02738. Name the package Export Report.

```
sas-admin transfer export --name "Export Report"
   --resource-uri "/reports/reports/faa7f5f2-0822-4ca0-9f92-23bda3e02738"
```

**Example:** Display the contents of a transfer package that is named Export Report and include import and export history of the package.

```
sas-admin transfer list --name "Export Report"
sas-admin transfer show --id transfer-package-ID --history
```

Note: Beginning with SAS Viya 3.4 in July 2019 and later deployments, the TRANSFER SHOW command was updated to show more information about the package contents.

```
1 sas-admin transfer list --name "Export Report"
2 sas-admin transfer show --id transfer-package-ID --history
```

1 List information about the transfer package and record the transfer package ID.

2 Show import and export history of the specified package. You get a list of import and export tasks for the package. Choose the number of the import or export task for which you want history. In this example, task 1 was selected.

```
Num   Type     StartedTimeStamp           CompletedTimeStamp         State
```
Enter (q)uit or a number to get task details> 1

You get a summary of results for the selected task. Choose the number of the item that you want information about. In this example, item 4, SAS Environment Manager, was selected.

Summary results of import:

Total tasks: 4  Succeeded: 4  Failed: 0  Completed with errors: 0  Completed with warnings: 0  Skipped: 0

<table>
<thead>
<tr>
<th>Num</th>
<th>Name</th>
<th>State</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Dashboard Items</td>
<td>Completed</td>
<td>/folders/folders/16d6016b-ca0a-438a-b502-538742bef506</td>
</tr>
<tr>
<td>2</td>
<td>Application Activity</td>
<td>Completed</td>
<td>/reports/reports/ecec39ad-994f-4055-8e40-4360f410bc6e</td>
</tr>
<tr>
<td>3</td>
<td>Products</td>
<td>Completed</td>
<td>/folders/folders/707b2a8c-b1f8-4253-9a07-32d10234a26c</td>
</tr>
<tr>
<td>4</td>
<td>SAS Environment Manager</td>
<td>Completed</td>
<td>/folders/folders/ac7814c4-eb57-45fd-ae8a-d3add0b57075</td>
</tr>
</tbody>
</table>

Enter (q)uit or a number to get task details> 4

You get details about the selected item.

Name        SAS Environment Manager
State       Completed
Started     2019-07-12T12:39:38.515Z
Completed   2019-07-12T12:39:41.055Z
Result      /folders/folders/ac7814c4-eb57-45fd-ae8a-d3add0b57075
Location    /Products/SAS Environment Manager
Message     The resource was successfully promoted.

Example: Get the mapping information for the transfer package that is named Export Report from the previous example.

```
sas-admin transfer list --name "Export Report"
sas-admin transfer get-mapping --id transfer-package-ID --mapping /tmp/map.txt
```

1 List the ID for the transfer package that you want to export.

2 Issue the command to retrieve the mapping information for the Export Report transfer package that has the specified ID. Write the file to the map.txt file.

Example: Upload a package source environment to the target environment. Write the mapping file to the specified location.

```
sas-admin --profile source transfer download --id transfer-package-ID --file /tmp/MyPackage.json
sas-admin --profile target transfer upload --file /tmp/MyPackage.json --mapping /tmp/map.txt
```

1 Download the package to your local machine and store it in a package file that is named MyPackage.json.

2 Upload the MyPackage.json file to the target environment. Write the file to the map.txt file.

Details

You can specify information about the export or import operation that you want to perform using the Transfer service REST API standards, as follows:

- **export**

You can specify information about the export operation that you want to perform using the REQUEST option of the TRANSFER EXPORT command. The option accepts JSON input of type
ExportRequest. The content can be contained in a quoted string or in a file. The filename must begin with the at sign (@). For example:

@filename.json
@/path/filename.json

The content of this option makes up the POST request through which the export package is sent.

Table 9  HTTP POST Export Request Members

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>integer</td>
<td>The schema version number of the JSON media type. This is version 1.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>The name of the export job that is used to export objects from a source system to a transfer package. This is also the name of the transfer package that is being created.</td>
</tr>
<tr>
<td>description</td>
<td>string</td>
<td>A short description of the export job.</td>
</tr>
<tr>
<td>items</td>
<td>list</td>
<td>The list of URIs to include in the transfer package.</td>
</tr>
</tbody>
</table>

Here is a sample JSON file of type ExportRequest. The name of the file is export.json.

```json
{
  "version": 1,
  "name": "My reports",
  "description": "Export of all my reports",
  "items": [
    "/reports/reports/4d083692-3c9a-4f2c-945f-e96fad972036",
    "/folders/folders/d4f1533a-229d-4d45-a5c9-b8a21fbc1e39"
  ]
}
```

You can use either of the following syntax options of the command to export the information:

- `sas-admin transfer export --request @/path-to-file/export.json`

  **Note:** When you include the information in a file, the first character of the file name or file path following the `REQUEST` option must be the at sign (@).

- `UNIX: sas-admin transfer export --request '{ "version": 1, "name": "My reports", "description": "Export of all my reports", "items": [ "/reports/reports/4d083692-3c9a-4f2c-945f-e96fad972036", "/folders/folders/d4f1533a-229d-4d45-a5c9-b8a21fbc1e39" ]}'`

  **Windows:** `sas-admin transfer export --request "{ "version": 1, "name": "My reports", "description": "Export of all my reports", "items": [ "/reports/reports/4d083692-3c9a-4f2c-945f-e96fad972036", "/folders/folders/d4f1533a-229d-4d45-a5c9-b8a21fbc1e39" ]}"`

  **Note:** When running the transfer plug-in in a Windows environment, single quotation marks are not allowed. You must enclose the `REQUEST` option data in double quotation marks, and then escape the embedded double quotation marks with a backslash (\).
You can specify information about the import operation that you want to perform using the REQUEST option of the TRANSFER IMPORT command. The option accepts JSON input of type ImportRequest. The content can be contained in a quoted string or in a file. The file name must begin with an at sign (@). For example:

@filename.json

@/path/filename.json

The content of this option makes up the POST request through which the import package is sent.

### Table 10  HTTP POST Import Request Members

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>integer</td>
<td>The schema version number of the JSON media type. This is version 1.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>The name of the import job that is used to import objects from a transfer package to a target system.</td>
</tr>
<tr>
<td>description</td>
<td>string</td>
<td>A short description of the import job.</td>
</tr>
<tr>
<td>packageUri</td>
<td>string</td>
<td>The package to import.</td>
</tr>
</tbody>
</table>

Here is an example of a JSON file of type ImportRequest that is named import.json:

```json
{
    "version": 1,
    "name": "My reports",
    "description": "import all my reports ",
    "packageUri": "/transfer/packages/a2ef940e-14ac-4960-9a9a-7689702b06f0"
}
```

You can use either of the following syntax options of the command to import the information:

- sas-admin transfer import --request @/path-to-file/import.json

  **Note:** When you include the information in a file, the first character of the file name or file path following the REQUEST option must be the at sign (@).

- UNIX: sas-admin transfer import --request '{ "version": 1, "name": "My reports", "description": "import all my reports ", "packageUri": "/transfer/packages/a2ef940e-14ac-4960-9a9a-7689702b06f0" }'

  **Windows:** sas-admin transfer import --request "{"version": 1, "name": "My reports", "description": "import all my reports ", "packageUri": "/transfer/packages/a2ef940e-14ac-4960-9a9a-7689702b06f0" }"

  **Note:** When running the transfer CLI in a Windows environment, single quotation marks are not allowed. You must enclose the REQUEST option data in double quotation marks, and then escape the embedded double quotation marks with a backslash (\).
Mapping Files

A mapping file is a file in YAML or JSON format that you can edit to specify substitution values for data libraries and connections. The Import wizard requires a mapping file in JSON format. The transfer CLI reads and writes mapping files in YAML or JSON format. The mapping file can contain the following:

- connectors
  Connectors denote a category of items that are being promoted, such as tables, users, and groups. Each connector section contains the following lines: `resourceName:`, `source:`, and `target:`. See “Connectors”.

- substitutions
  Substitutions denote a set of properties that originate from the substitution properties file available in SAS 9.x systems. The `substitutions` section in the mapping file contains the following lines: `resourceId:`, `resourceName:`, and `properties:`.

- options
  Options denote additional items, such as the following:
  - URIs for stored processes on page 15.
  - a setting to prevent the promotion of ACEs. See “Details: Authorizations”.

SAS 9 to SAS Viya Mapping File

Connectors

The mapping file for a SAS 9 to SAS Viya promotion contains connectors such as the following:

- table connector
  The mapping file for a SAS 9 to SAS Viya promotion can contain a list of tables. Each one is preceded by a table connector. The table connector is denoted by the following lines: `connectors:` and `table:`. The `table:` connector maps information about a source table to a target table.

  The first table in the mapping file that follows the `connectors:` line can be a default source table denoted by the following line: `source: default`. Use the `target:` line for the default source table to specify the values that are inherited by the other tables.
Here is an example of the target information for a table in the mapping file. In this example, a table from the source environment is given the name `tablename` in the target environment, directed to the `cas_library` library in the target environment, and directed to the `cas_server_name` server in the target environment. The `DATA_SOURCE_LOCALE` parameter is optional.

```
target: "tablename, cas_library, cas_server_name, data_source_locale"
```

### user connector

The mapping file for a SAS 9 to SAS Viya promotion can contain a list of users. Each one is preceded by a `user:` connector. The `user:` connector is denoted by the following lines:

```
connectors: and user:
```

Each user that follows the `connectors:` line is denoted by the following lines: `resourceName:`, `source:`, and `target:`. You can use the `user:` connector to map SAS 9 reports whose paths begin with User Folders to the specified user's SAS Viya My Folder location.

Use the `source:` line to specify the name of the user that you want to map in the SAS 9 User Folders location. Use the `target:` line to specify the user ID of the user.

Here is an example of a `user:` connector:

```
"user": [
  {
    "resourceName": " *[@Name='John Smith']",
    "source": "John Smith",
    "target": "josmit"
  },
```

Suppose that you want to promote the SomeReport report from `~/User Folders/Smith/John/My Folder/My Reports` in SAS 9. If you edited the mapping file to contain the user connector in the preceding example, then after the import, the SomeReport report appears in the SAS Viya folder `~/My Folder/My Reports` for the user with the josmit user ID. The import can create a My Folder for josmit even if josmit has not previously logged in. If you did not edit the mapping file with the user connector, then the source path is used as is. All needed folders are created in the SAS Viya environment.

### ContentMappings.json File

The `ContentMappings.json` file is the initial mapping file for SAS 9 to SAS Viya promotions. This mapping file is created by the `SAS-ADMIN TRANSFER GENERATE-CONTENT-MAPPING` command and can be read by the Import wizard. See "The GENERATE-CONTENT-MAPPING Command".

### Sample Mapping File in JSON Format

Here is a sample mapping file in JSON format for a SAS 9 to SAS Viya promotion. This mapping file was created in YAML format by the transfer plug-in and then converted to JSON format with a third-party utility.

```
{
  "version": 1,
  "connectors": {
    "table": [
      {
        "resourceName": "",
        "source": "default",
        "target": "HPS,cas-shared-mpp"
      },
      {
        "resourceName": "",
        "source": "/<path_to_table>/ORION_STAR_SCHEMA(Table)",
        "target": "ORION_STAR_SCHEMA"
      }
    ]
  }
}```
SAS Viya to SAS Viya Mapping File

Connectors
The mapping file for a SAS Viya to SAS Viya promotion contains connectors such as the following:

- table connector

The mapping file for a SAS Viya to SAS Viya promotion can contain a list of tables. Each one is preceded by a table connector. The table connector is denoted by the following lines: `connectors: ` and `table: `. The `table: ` connector maps information about a source table to a target table.

Here is a sample mapping file in JSON format for a SAS Viya to SAS Viya promotion. This mapping file was created by the Import wizard and can be read with the Import wizard or the transfer plug-in.

```json
{
    "version": 1,
    "connectors": {
        "table": [
            {
                "source": "server=<server>;library=<caslib>;table=AUDIT",
                "target": "server=<server>;library=<caslib>;table=AUDIT"
            }
        ]
    }
}
```

Here is a sample mapping file in YAML format for a SAS Viya to SAS Viya promotion. This mapping file was created by the transfer plug-in and can be read only with the transfer plug-in.
The mapping file for a SAS 9 to SAS Viya promotion can contain data plans. Each one is preceded by a datasource connector and a datatable connector. The datasource and datatable connectors are denoted by the following lines: `connectors: datasource:`, and `datatable`. These connectors map information about a source data plan to a target data plan. The values for these connectors must be URI-encoded. For example: `/dataSources/providers/cas/sources/cas-shared-default~fs~Financial%20Data`.

The GENERATE-CONTENT-MAPPING Command

To promote groups and membership information, you must run the SAS-ADMIN TRANSFER GENERATE-CONTENT-MAPPING command before running the Import wizard. This command creates a file named `ContentMappings.json` that you can use as a baseline for future imports.

The transfer plug-in `GENERATE-CONTENT-MAPPING` command extracts the following information from any users, groups, and servers package files that are provided:

- **user ID**
  - Used to look up users in SAS Viya in order to add authorizations to folders, content, libraries, and tables.

- **group**
  - Used to determine whether groups exist in the target SAS Viya system, and to create new groups if not. This is necessary to promote membership information.

- **LASR Signer information**
  - Used to import LASR libraries.

**IMPORTANT** The `ContentMappings.json` file is overwritten every time that the SAS-ADMIN TRANSFER GENERATE-CONTENT-MAPPING command is run. If you want to reuse this file, make sure that you save it and maintain it.

**Note:** Only users with administrative privileges can run the SAS-ADMIN TRANSFER GENERATE-CONTENT-MAPPING command. See *SAS Viya Administration: Identity Management*.

Substitution Properties

You can substitute a different target value for a value that is associated with an object that is being promoted so that the object functions properly in the target system. For example, these values might include physical paths, descriptions, or port numbers. In SAS Viya, these properties are referred to as substitution properties and originate from a SAS 9.x package SPK file.
For example, suppose that you are promoting a Base library, and you need to substitute a different directory path for SAS Viya. You can specify the new directory path for the associated substitution property. Then, when the package file is imported into SAS Viya, the Base library objects are directed to the new directory path.