SAS® Clinical Standards Toolkit 1.7
Migration Guide
# Contents

**Using This Book** .................................................. vii

**Chapter 1 • Installation Locations** ........................................ 1
  Introduction ........................................................................ 1
  Isasroot ........................................................................... 1
  Global Standards Library ..................................................... 2
  Sample Study Library .......................................................... 3

**Chapter 2 • SAS Deployment Wizard** ...................................... 5
  Introduction to the SAS Deployment Wizard .............................. 5
  Upgrade to SAS Clinical Standards Toolkit 1.7 in SAS 9.4 .......... 6
  Upgrade to SAS Clinical Standards Toolkit 1.7 By
    Installing SAS 9.4 in Conjunction with SAS 9.3 .................. 9

**Chapter 3 • Tools to Identify Differences between Versions** ........ 13
  Introduction ........................................................................ 14
  Compare Current User Customizations to the
    Previous Version of the SAS Clinical Standards
    Toolkit Using Checksums .................................................... 14
  Compare Two Registered Controlled Terminology Packages ......... 21
  Compare Two Codelist Sources ............................................ 22
  Compare Two Macro Autocall Libraries .................................. 25
  Compare Two Property Files ................................................ 26
  Compare Two Folder Hierarchies .......................................... 28
  Copy a Folder Hierarchy ..................................................... 33

**Chapter 4 • Tools to Migrate Source Metadata from CRT-DDS 1.0.0 to Define-XML 2.0.0** .......... 35
  Introduction ........................................................................ 35
  Overview ........................................................................... 36
  Study Source Metadata Migration Driver Programs ................ 37
Chapter 5 • Frequently Asked Questions

Installation Issues

Migration Process

Standards Management

Library Issues

Other Issues
Using This Book

Audience

This document provides guidance on migrating from one version of the SAS Clinical Standards Toolkit to another.

The intended audience is users who have installed and customized a previous version of the SAS Clinical Standards Toolkit and now want to install version 1.7.

If you are installing the SAS Clinical Standards Toolkit for the first time and it is version 1.7, or if you have installed previous versions of the SAS Clinical Standards Toolkit but did not customize them, you do not need to review this document or use any of the information when you install the SAS Clinical Standards Toolkit 1.7.

However, if you plan to subsequently customize the SAS Clinical Standards Toolkit, this document offers a preview of the issues that you might want to consider in the future as you migrate to later versions of the SAS Clinical Standards Toolkit.
viii Using This Book
## Installation Locations

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td>!sasroot</td>
<td>1</td>
</tr>
<tr>
<td>Global Standards Library</td>
<td>2</td>
</tr>
<tr>
<td>Sample Study Library</td>
<td>3</td>
</tr>
</tbody>
</table>

### Introduction

The installation process of the SAS Clinical Standards Toolkit puts files in these locations:

- !sasroot
- global standards library
- sample study library

### !sasroot

Generic (cross-standard or standard-independent) framework SAS macros are installed in the `!sasroot/cstframework/sasmacro` directory (Microsoft Windows) or in the `!sasroot/sasautos` directory (UNIX).
These primary autocall locations are the same for both SAS 9.3 and SAS 9.4 as well as for previous versions of the SAS Clinical Standards Toolkit and version 1.7. During installation of the SAS Clinical Standards Toolkit, SAS performs the following tasks:

- adds macros that are new to version 1.7
- overwrites macros that have been modified in version 1.7
- leaves in place macros that have been deprecated with version 1.7

**CAUTION!** If the timestamp of a macro in `!sasroot` is later than the timestamp of the SAS Clinical Standards Toolkit 1.7 macro to be installed, the installation process does not replace the macro in the `!sasroot` location. If you have modified `!sasroot` macros, you assume responsibility for reconciling your changes with newer versions of those macros. Your code management processes must include backing up files so that you do not risk having your changes overwritten when newer macros are installed. There are several tools that help you identify changes in the framework autocall macros.

**TIP** Best Practice Recommendation: Do not modify the macros in the autocall locations. Make a copy of each macro to modify, modify the copy, and save the copy in another folder. Add the folder to the SAS autocall path.

---

**Global Standards Library**

The primary function of the SAS Clinical Standards Toolkit global standards library is to provide the metadata that defines each standard that is supported by the SAS Clinical Standards Toolkit.

The location of the global standards library is specified during installation, but it can be moved to another location. By default, this location is set to the `C:\cstGlobalLibrary` directory (Microsoft Windows) or to the `/usr/local/cstGlobalLibrary` directory (UNIX).

If the global standards library of a previous version of the SAS Clinical Standards Toolkit was installed in the default location, the installation process asks you to specify another
location for the global standards library for version 1.7. (For more information, see Chapter 2, “SAS Deployment Wizard,” on page 5.) You can rename or move the global standards library for the previous version before starting the installation of version 1.7.

The **global standards library directory**/metadata directory contains the cumulative metadata for every standard that is registered to the SAS Clinical Standards Toolkit. The **metadata/standards.sas7bdat** data set contains records for each supported standard. The number of records varies with each release because support for new standards is added or deprecated.

If you have customized a previous version of the SAS Clinical Standards Toolkit to add your own standards, you must register those customized standards to the SAS Clinical Standards Toolkit version 1.7. This process is described in the “Framework” chapter in the *SAS Clinical Standards Toolkit User’s Guide*.

---

**Sample Study Library**

The SAS Clinical Standards Toolkit sample study library includes a sample folder for each standard. Each sample folder contains the files that represent a sample SAS implementation of the specific standard, sample study data sets, and sample programs and results that illustrate SAS Clinical Standards Toolkit functionality.

The location of the sample study library is specified during installation, but it can be moved to another location. By default, this location is set to the **C:\cstSampleLibrary** directory (Microsoft Windows) or to the **/usr/local/cstSampleLibrary** directory (UNIX).

As a part of each standard definition that is supplied by SAS, the **global standards library directory/standards/standard and version/control/standards.sas7bdat** data set contains a column named studylibraryrootpath. By default, this column has been set to the root path of the sample study for that standard. You can use this column to point to the root path of another study hierarchy within your organization. If you have modified this path for any of the standards supplied by SAS, you must ensure that mapping is retained after migrating from a previous version of the SAS Clinical Standards Toolkit to version 1.7.
Introduction to the SAS Deployment Wizard

This chapter describes the installation of the SAS Clinical Standards Toolkit 1.7 on Windows 7 Enterprise x64 (Service Pack 1) using the SAS Deployment Wizard in two different scenarios:

- Upgrade to SAS Clinical Standards Toolkit 1.7 in SAS 9.4
- Upgrade to SAS Clinical Standards Toolkit 1.7 by installing SAS 9.4 in conjunction with SAS 9.3 that already has a previous version of the SAS Clinical Standards Toolkit installed
Note: Usually, you use only one of these scenarios, depending on your situation.

Upgrade to SAS Clinical Standards Toolkit 1.7 in SAS 9.4

Overview

If you have modified any file that is part of a previous version of the SAS Clinical Standards Toolkit, your code management processes should include file backups. File backups minimize these risks:

- A modified file is deleted when you uninstall the previous version of the SAS Clinical Standards Toolkit.
- A modified file is overwritten when you install the SAS Clinical Standards Toolkit 1.7.

SAS recommends that you uninstall the previous version of the SAS Clinical Standards Toolkit in SAS 9.4 before installing version 1.7 in SAS 9.4.

If you do not uninstall the previous version of the SAS Clinical Standards Toolkit in SAS 9.4, you risk having files that contain deprecated macros (that are not part of the SAS Clinical Standards Toolkit 1.7) left behind in the !sasroot/cstframework/sasmacro directory. Here are examples of macros that were present in previous releases of the SAS Clinical Standards Toolkit that do not exist in version 1.7:

- cst_createds.sas
- cst_createemptytables.sas
- cst_getstandardmetadatas.sas
- cstcheck_java.sas
- cstutil_createunixsubdir.sas
- csutilwriteresultsintro.sas
CAUTION! Driver programs in a previous version of the SAS Clinical Standards Toolkit that call these macros and that are migrated to version 1.7 might give unexpected results.

Uninstall a Previous Version of the SAS Clinical Standards Toolkit

To uninstall a previous version of the SAS Clinical Standards Toolkit, perform the following steps:

1. In the Windows Control Panel, uninstall SAS 9.4. The Choose Language dialog box appears.

2. Select a language, and then click OK. The Select SAS Products to Uninstall page of the SAS Deployment Wizard appears.

3. Select these products:
   - SAS Clinical Standards Toolkit Global Standards Library
   - SAS Clinical Standards Toolkit Sample Library
   - SAS Clinical Standards Toolkit Framework JAR File
   - SAS Foundation 9.4

4. Click Next. SAS prepares to uninstall the products. The Checking System page appears.

5. Click Next, and then click Start. Once the uninstall process is complete, the !sasroot/cstframework/sasmacro folder is deleted.

6. If the !sasroot/cstframework/sasmacro folder has not been deleted automatically, delete it manually.
If it has not been deleted, old macros remain in the folder. If you previously updated any of the macros in this folder, the old macros were not deleted.

**CAUTION!** Ensure that you have backed up the SAS Clinical Standards Toolkit macros that you want to preserve.

If the global standards library folder and the sample study library folder exist, manually delete them.

These folders might contain previous installation information that is not needed. The default locations for these folders are `C:\cstGlobalLibrary` and `C:\cstSampleLibrary`.

**Install the SAS Clinical Standards Toolkit 1.7 for SAS 9.4**

To install the SAS Clinical Standards Toolkit 1.7 for SAS 9.4, perform the following steps:

1. Run `setup.exe`, which is located in the new SAS Software Install Depot. The Choose Language dialog box appears.

2. Select a language, and then click **OK**.

   The Select Deployment Task page of the SAS Deployment Wizard appears.

3. Select **Install SAS software**, and then click **Next**.

   The Select Products to Install page appears.

4. Select **SAS Foundation**, and then click **Next**.

   The Select SAS Foundation Products page appears.

5. Select **SAS Clinical Standards Toolkit Framework**, and then click **Next**.

   The Specify SAS Installation Data File page appears.

6. Enter the path to the SAS installation data file, and then click **Next**.
The Select Language Support page appears.

7 Select one or more languages, and then click Next.

The Specify SAS Clinical Standards Toolkit Libraries page appears.

8 Enter the locations of the global standards library and the sample study library.

Note: The locations cannot be the same. And, the locations cannot already exist; they must be created by the SAS Deployment Wizard.

9 Click Next twice.

The installation begins.

---

Upgrade to SAS Clinical Standards Toolkit 1.7 By Installing SAS 9.4 in Conjunction with SAS 9.3

Overview

This scenario assumes that you have installed SAS 9.3 with a previous version of the SAS Clinical Standards Toolkit. You can keep this installation and install SAS 9.4 with the SAS Clinical Standards Toolkit 1.7 in a separate environment.

Install the SAS Clinical Standards Toolkit 1.7 for SAS 9.4

To install the SAS Clinical Standards Toolkit 1.7 for SAS 9.4, perform the following steps:

1 Run setup.exe, which is located in the new SAS Software Install Depot.

The Choose Language dialog box appears.

2 Select a language, and then click OK.
The Select Deployment Task page of the SAS Deployment Wizard appears.

3 Select **Install SAS software**, and then click **Next**.

The Specify SAS Home page appears because SAS 9.3 is installed. The SAS Deployment Wizard requires you to specify a different location for SASHome for SAS 9.4.

4 Enter a location for SASHome.

On Windows 7 Enterprise x64 (Service Pack 1), the default location is `C:\Program Files\SASHome2`.

5 Click **Next**.

The Select Deployment Type page appears.

6 Select **Install SAS Foundation and Related Software**, and then click **Next**.

The Select Products to Install page appears.

7 Select **SAS Foundation**, and then click **Next**.

The Select SAS Foundation Products page appears.

8 Select **SAS Clinical Standards Toolkit Framework**, and then click **Next**.

The Specify SAS Installation Data File page appears.

9 Enter the path to the SAS installation data file, and click **Next**.

The Select Language Support page appears.

10 Select one or more languages, and then click **Next**.

The Select Regional Settings page appears.

11 Specify the regional settings, and then click **Next**.

The Default Product for SAS File Types page appears.

12 Specify the default product to use to open SAS file types, and then click **Next**.

The Specify SAS Clinical Standards Toolkit Libraries page appears.
13 Enter the locations of the global standards library and the sample study library, and then click **Next**.

**Note:** The locations cannot be the same. And, the locations cannot already exist; they must be created by the SAS Deployment Wizard.

14 Click **Next**.

The Checking System page appears.

15 Click **Next**.

The Deployment Summary page appears.

16 Click **Start**.

The installation begins.

After the installation is complete, the Deployment Complete page appears.

17 Click **Next**.

The Additional Resources page appears.

18 Click **Finish**.
Tools to Identify Differences between Versions

Introduction .................................................................................................................. 14

Compare Current User Customizations to the Previous Version of the SAS Clinical Standards

Toolkit Using Checksums ............................................................................................... 14
Overview ......................................................................................................................... 15
cstutilgeneratechecksums Macro .................................................................................. 16
cstutilcomparechecksums Macro .................................................................................. 16
compare_checksums.sas Driver Program ...................................................................... 17
Examples: Compare Checksum Files for !sasroot ......................................................... 17
Example: Compare Checksum Files for the Global Standards Library ......................... 20
Example: Compare Checksum Files for the Sample Study Library .............................. 20

Compare Two Registered Controlled Terminology Packages .................................. 21
Overview ......................................................................................................................... 21
cstutilcompareregisteredct Macro ............................................................................... 21

Compare Two Codelist Sources ..................................................................................... 22
Overview ......................................................................................................................... 22
cstutilcomparecodelists Macro ..................................................................................... 22

Compare Two Macro Autocall Libraries ...................................................................... 25
Overview ......................................................................................................................... 25
cstutilcompareautocallmacros Macro ......................................................................... 25
# Introduction

SAS supplies tools to identify and report differences between two entities, such as folders or controlled terminology packages.

**Note:** Reconciliation of differences is outside the scope of this document.

The macros mentioned in this chapter are located in the `!sasroot/cstframework/sasmacro` folder.

---

## Compare Two Property Files

- Overview .......................................................... 26
- cstutilcompareproperties Macro .............................. 27

## Compare Two Folder Hierarchies

- Overview .......................................................... 28
- cstutilcomparefolderhierarchy Macro ....................... 28
- Example 1: Compare Global Standards Library
  - Folder Hierarchies ............................................. 28
- Example 2: Compare Global Standards Library Files .......... 31

## Copy a Folder Hierarchy

- Overview .......................................................... 33
- cstutilcopyfolderhierarchy Macro ............................ 33
Clinical Standards Toolkit Using Checksums

Overview

As described in Chapter 1, “Installation Locations,” on page 1, the SAS Clinical Standards Toolkit places files in three locations: !sasroot, the global standards library, and the sample study library. To help you determine whether the files in the installation locations for a previous version of the SAS Clinical Standards Toolkit have been modified from the files that were originally supplied by SAS, SAS supplies checksum files for previous versions.

The cstmigration.zip file contains checksum files for the following versions of the SAS Clinical Standards Toolkit:

<table>
<thead>
<tr>
<th>Version</th>
<th>Checksum File From the Root Path</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>\cstMigration\checksums\productionchecksums\</td>
</tr>
<tr>
<td>1.5</td>
<td>checksums_cst15_cstframework.xml</td>
</tr>
<tr>
<td></td>
<td>checksums_cst15_cstgblstdlib.xml</td>
</tr>
<tr>
<td></td>
<td>checksums_cst15_cstsamplelib.xml</td>
</tr>
<tr>
<td>1.5.1</td>
<td>checksums_cst151_cstframework.xml</td>
</tr>
<tr>
<td></td>
<td>checksums_cst151_cstgblstdlib.xml</td>
</tr>
<tr>
<td></td>
<td>checksums_cst151_cstsamplelib.xml</td>
</tr>
<tr>
<td>1.6</td>
<td>checksums_cst16_cstframework.xml</td>
</tr>
<tr>
<td></td>
<td>checksums_cst16_cstgblstdlib.xml</td>
</tr>
<tr>
<td></td>
<td>checksums_cst16_cstsamplelib.xml</td>
</tr>
<tr>
<td>1.7</td>
<td>checksums_cst17_cstframework.xml</td>
</tr>
<tr>
<td></td>
<td>checksums_cst17_cstgblstdlib.xml</td>
</tr>
<tr>
<td></td>
<td>checksums_cst17_cstsamplelib.xml</td>
</tr>
</tbody>
</table>
cstutilgeneratechecksums Macro

The cstutilgeneratechecksums macro creates a checksum file for the specified folder. You usually specify the folder that contains customized files in your version of the SAS Clinical Standards Toolkit. This checksum file is used by the cstutilcomparechecksums macro to compare against another checksum file.

The following example illustrates the use of the cstutilgeneratechecksums macro to create a checksum file for the folder C:/myCustomGlobalLibrary:

libname custom "<user location for generated checksum XML file>";
%cstutilgeneratechecksums(
   _cstFolder=C:/myCustomGlobalLibrary,
   _cstXMLFile=%sysfunc(pathname(custom))/checksums_cst15_myCustomGlobalLibrary.xml,
   _cstProdCode=cstgblstdlib,
   _cstLabel=CST 1.5 Custom Global Library
);

For more information about the macros and their parameters, see the SAS Clinical Standards Toolkit: Macro API Documentation.

cstutilcomparechecksums Macro

The cstutilcomparechecksums macro compares two checksum files that were created by the cstutilgeneratechecksums macro.

The following example illustrates the use of the cstutilcomparechecksums macro to compare the checksum file created in the previous example to the checksum file for the SAS Clinical Standards Toolkit 1.6 as supplied by SAS:

%cstutilcomparechecksums(
   _cstBaseXMLFile=%sysfunc(pathname(chksums))/checksums_cst16_cstlib.xml,
   _cstCompXMLFile=%sysfunc(pathname(chksums))/checksums_cst15_myCustomGlobalLibrary.xml,
   _cstCompResults=work.cstlib,
   _cstOutReportPath=%sysfunc(pathname(chksums)),
   _cstOutReportFile=cstlib_myCustomGlobalLibrary_16.html
);

For more information about the macros and their parameters, see the SAS Clinical Standards Toolkit: Macro API Documentation.
**compare_checksums.sas Driver Program**

The compare_checksums.sas driver program illustrates using the cstutilgeneratechecksums macro and the cstutilcomparechecksums macro to compare the checksum files for !sasroot, the global standards library, and the sample study library. The compare_checksums.sas driver program is located in the *global standards library directory*/standards/cst-framework-1.7/utilities/checksums folder.

The checksum files for these versions of the SAS Clinical Standards Toolkit are compared in the compare_checksums.sas driver program:

- 1.5 to 1.5.1
- 1.5.1 to 1.6
- 1.5 to 1.6

**Note:** The results of running the compare_checksums.sas driver program are located in the *global standards library directory*/standards/cst-framework-1.7/utilities/checksums/results folder. The results shown in the following examples assume that the installed files have not been customized.

**Examples: Compare Checksum Files for !sasroot**

The following example in the compare_checksums.sas driver program compares the checksum files for the cstframework autocall library in !sasroot of the SAS Clinical Standards Toolkit 1.5 to 1.5.1:

```sas
%cstutilcomparechecksums(
   _cstBaseXMLFile=%sysfunc(pathname(chksums))/checksums_cst15_cstframework.xml,
   _cstCompXMLFile=%sysfunc(pathname(chksums))/checksums_cst151_cstframework.xml,
   _cstCompResults=work.cstframework,
   _cstOutReportPath=&_cstMigrationPath/results,
   _cstOutReportFile=cstframework_15_151.html
);
```
The results are written to the location that is specified by the `_cstOutReportPath` parameter.

**Note:** This location is used by all of the `cstutilcomparechecksums` macro examples.

The `cstutilcomparechecksums` macro creates two lists: one lists the files that are different and one lists the files that are identical.

The following display shows the list of files that are different:

**Display 3.1 Different Files between the SAS Clinical Standards Toolkit 1.5 and 1.5.1**

<table>
<thead>
<tr>
<th>File</th>
<th>Checksum (CST 1.5)</th>
<th>Checksum (CST 1.5.1)</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>est_createds.sas</code></td>
<td>2cde5fbd39ef3d5c45</td>
<td>7b6776e03c76742152d6e641215c6412</td>
<td>Files are different</td>
</tr>
<tr>
<td><code>estutilcheckfiles.sas</code></td>
<td>3d5f2d3b7c4b748d54</td>
<td>0b5a6c3e5e7a7694612a</td>
<td>Files are different</td>
</tr>
<tr>
<td><code>estutilcheckreferences.sas</code></td>
<td>7e7d6816b7c6d45631</td>
<td>7b6771603c76742152d6e641215c6412</td>
<td>Files are different</td>
</tr>
<tr>
<td><code>estutil_readflags.sas</code></td>
<td>2e54b39472bd565a5e4</td>
<td>0b5a6c3e5e7a7694612a</td>
<td>Files are different</td>
</tr>
<tr>
<td><code>estutil_writeqtems.sas</code></td>
<td>4abcdef56789012345</td>
<td>4abcdef56789012345</td>
<td>Files are different</td>
</tr>
<tr>
<td><code>estutil_writeqtemscramble.sas</code></td>
<td>6e944b44d4c11956b3223014e1d47</td>
<td>0b5a6c3e5e7a7694612a</td>
<td>Files are different</td>
</tr>
<tr>
<td><code>estutil_wrongdatametadatameta.sas</code></td>
<td>0a975f400205d130649</td>
<td>0b5a6c3e5e7a7694612a</td>
<td>Files are different</td>
</tr>
<tr>
<td><code>estutil_datavalidationcode.sas</code></td>
<td>edf6145db5c2311b2044</td>
<td>0b5a6c3e5e7a7694612a</td>
<td>Files are different</td>
</tr>
<tr>
<td><code>estutil_checkfil esframe.sas</code></td>
<td>9b7a81134b2350d1548c3130</td>
<td>0b5a6c3e5e7a7694612a</td>
<td>Files are different</td>
</tr>
</tbody>
</table>

**Note:** The list of files that are identical is not shown.

The `compare_checksum.sas` driver program also compares the checksum files between major versions. Such a comparison identifies more differences. For example, comparing the `cstframework` autocall library for SAS Clinical Standards Toolkit 1.5.1 to 1.6 creates three lists: one lists the files that are different, one lists the files that are missing, and one lists the files that are identical.

The list of the files that are different between major versions is longer than the list of the files that are different between minor versions.

**Note:** Most of the changes to the files that are different between major versions generally involve only the macro headers, not the functionality of the macros. So, the list is not shown.
The following display shows the list of files that are missing between the SAS Clinical Standards Toolkit 1.5.1 and 1.6:

**Display 3.2  Files Missing between the SAS Clinical Standards Toolkit 1.5.1 and 1.6**

Base Directory: C:\Program Files\SASHome\SASFoundation\9.3\cstframework\sasmacro (2014-02-18T17:50:26)
Compare Directory: C:\Program Files\SASHome2\SASFoundation\9.4\cstframework\sasmacro (2014-02-18T17:50:30)
SAS Production Code: cstframework

<table>
<thead>
<tr>
<th>File</th>
<th>Checksum (CT 1.5.1)</th>
<th>Checksum (CT 1.6)</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>cst_created.sas</td>
<td>79f775bfe833777f31a2b9a1026f025ff6</td>
<td></td>
<td>File not found in COMP</td>
</tr>
<tr>
<td>cst_createemptytables.sas</td>
<td>d15bcb3e816d03d05e8a913555053402</td>
<td></td>
<td>File not found in COMP</td>
</tr>
<tr>
<td>cst_getstandardmetadata.sas</td>
<td>76425e71a3d0ab2d072a8a301eab1f057</td>
<td></td>
<td>File not found in COMP</td>
</tr>
<tr>
<td>cstcheck.java.sas</td>
<td>3bb2a783c0e201c8f30e56c74015215</td>
<td>d6f79d9ae6856f629712c06a41e0422</td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstgetmetadatatrusted.standards.sas</td>
<td>c217b4431c3f8b25a538d4439e5805530c9</td>
<td>75c0d26333ec5b7b5c9eb74077efb4f1</td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdiaddataset.sas</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cstdiaddsection.sas</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cstdiaddsection.sas</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cstdiaddsection.sas</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cstdiaddcolumn.sas</td>
<td>374e0b9e6ee9461e1b2f2f7502e797</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdiappendmetadataprocesses.sas</td>
<td>bee6f341f7a3f510e03b77a4e141e226d</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdiappendsnapshot.sas</td>
<td>ad95b5703198602bca2a7010f502d9e</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdiappendmetadataprocesses.sas</td>
<td>4de575ca781d7a285a3510aa53f4172</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdideletedcolumn.sas</td>
<td>15567da59241590b1079d9d98f9d</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdideletednormaldata.sas</td>
<td>94c12fa650164be8b1072f703b5731</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdigegetattribute.sas</td>
<td>e55d69e85957b205f95e430c3d93e662</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdigegetattribute.sas</td>
<td>a852d65064e69e8c0a0583550268</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdigevent.sas</td>
<td>d31257c799821546692d6c87d305bb6</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdigegetcolumnattribute.sas</td>
<td>a29144a5e754b4ae6b102d1271a79f5</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdigetprocessxml.log.sas</td>
<td>4b53f33382bd68d365f8a2a3a11f243</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdigetprocessxml.log.sas</td>
<td>ce14b6297051920c9d909e656440</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdigetprocessxml.log.sas</td>
<td>15b804d77a12b35a452b99a4d44fa</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdigetprocessxml.log.sas</td>
<td>66ef1869d6c67da22e96172675050c8</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdiinterfacetoolsintro.sas</td>
<td>1a033c7b99a9917b598f4b0d6b5ae</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdiinterfacetoolsintro.sas</td>
<td>b372f5b5ed4c0d0471a492b9f7807f4</td>
<td></td>
<td>File not found in BASE</td>
</tr>
<tr>
<td>cstdiinterfacetoolsintro.sas</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This list identifies the files that are no longer in the SAS Clinical Standards Toolkit 1.6 (File not found in COMP) and the files that are new to 1.6 (File not found in BASE).

**Note:** The list of files that are identical between the two versions is not shown.
Example: Compare Checksum Files for the Global Standards Library

The following example in the compare_checksums.sas driver program compares checksum files for the global standards library of the SAS Clinical Standards Toolkit 1.51 to 1.6:

```
%cstutilcomparechecksums(
   _cstBaseXMLFile=%sysfunc(pathname(chksums))/checksums_cst151_cstgblstdlib.xml,
   _cstCompXMLFile=%sysfunc(pathname(chksums))/checksums_cst16_cstgblstdlib.xml,
   _cstCompResults=work.cstgblstdlib,
   _cstOutReportPath=&_cstMigrationPath/results,
   _cstOutReportFile=cstgblstdlib_151_16.html
);
```

Example: Compare Checksum Files for the Sample Study Library

The following example in the compare_checksums.sas driver program compares checksum files for the sample study library of the SAS Clinical Standards Toolkit 1.51 to 1.6:

```
%cstutilcomparechecksums(
   _cstBaseXMLFile=%sysfunc(pathname(chksums))/checksums_cst151_cstlib.xml,
   _cstCompXMLFile=%sysfunc(pathname(chksums))/checksums_cst16_cstlib.xml,
   _cstCompResults=work.cstlib,
   _cstOutReportPath=&_cstMigrationPath/results,
   _cstOutReportFile=cstlib_151_16.html
);
```
Compare Two Registered Controlled Terminology Packages

Overview

To help you determine whether new controlled terminology packages are available with the latest version of SAS Clinical Standards Toolkit, SAS supplies the cstutilcompareregisteredct macro.

cstutilcompareregisteredct Macro

The cstutilcompareregisteredct macro compares records in one SAS Clinical Standards Toolkit standardsubtypes data set with records in another standardsubtypes data set (such as in two different versions of the SAS Clinical Standards Toolkit). The default folder for the standardsubtypes data set is global standards library directory/standards/cdisc-terminology-SAS Clinical Standards Toolkit version/control.

This example compares the SAS Clinical Standards Toolkit 1.6 standardsubtypes data set to the 1.5 standardsubtypes data set:

```sas
libname newct 'C:\cstGlobalLibrary\standards\cdisc-terminology-1.6\control';
libname oldct 'C:\cstGlobalLibrary15\standards\cdisc-terminology-1.5\control';

%cstutilcompareregisteredct(
   _cstBaseCT=oldct.standardsubtypes,
   _cstNewCT=newct.standardsubtypes,
   _cstRptType=DATASET,
   _cstRptDS=work.CTchanges,
   _cstOverwrite=Y
);
```

For more information about the macros and their parameters, see the SAS Clinical Standards Toolkit: Macro API Documentation.
The following display shows the data set created by the cstutilcompareregisteredct macro (not all columns are shown):

**Display 3.3  Controlled Terminology Package Differences Reported by the cstutilcompareregisteredct Macro**

<table>
<thead>
<tr>
<th>standardversion</th>
<th>description</th>
<th>message</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CDISC-CDASH CDISC CDASH Controlled Terminology, released by NCI on 2011-04-08</td>
<td>Previous CT package found in oldct.standardsubtypes but not found in newct.standardsubtypes</td>
</tr>
<tr>
<td>2</td>
<td>CDISC-CDASH CDISC CDASH Controlled Terminology, released by NCI on 2013-12-20</td>
<td>New CT package found in newct.standardsubtypes but not found in oldct.standardsubtypes</td>
</tr>
<tr>
<td>3</td>
<td>CDISC-QS CDISC QS Controlled Terminology, released by NCI on 2013-12-20</td>
<td>New CT package found in newct.standardsubtypes but not found in oldct.standardsubtypes</td>
</tr>
<tr>
<td>4</td>
<td>CDISC-QS Current QS Controlled Terminology, Copy of 2013-12-20</td>
<td>New CT package found in newct.standardsubtypes but not found in oldct.standardsubtypes</td>
</tr>
<tr>
<td>5</td>
<td>CDISC-SDTM CDISC SDTM Controlled Terminology, released by NCI on 2011-04-08</td>
<td>Previous CT package found in oldct.standardsubtypes but not found in newct.standardsubtypes</td>
</tr>
<tr>
<td>6</td>
<td>CDISC-SDTM CDISC SDTM Controlled Terminology, released by NCI on 2013-12-20</td>
<td>New CT package found in newct.standardsubtypes but not found in oldct.standardsubtypes</td>
</tr>
<tr>
<td>7</td>
<td>CDISC-SEND CDISC SEND Controlled Terminology, released by NCI on 2012-01-03</td>
<td>Previous CT package found in oldct.standardsubtypes but not found in newct.standardsubtypes</td>
</tr>
<tr>
<td>8</td>
<td>CDISC-SEND CDISC SEND Controlled Terminology, released by NCI on 2013-12-20</td>
<td>New CT package found in newct.standardsubtypes but not found in oldct.standardsubtypes</td>
</tr>
</tbody>
</table>

**Compare Two Codelist Sources**

**Overview**

To help you determine the differences between two codelists, SAS supplies the cstutilcomparecodelists macro.

**cstutilcomparecodelists Macro**

The following example compares the SAS Clinical Standards Toolkit 1.6 codelist data set to the 1.5 codelist data set:

```sql
libname newct 'C:\cstGlobalLibrary\standards\cdisc-terminology-1.6\control';
```
libname oldct 'C:\cstGlobalLibrary15\standards\cdisc-terminology-1.5\control';

%cstutilcomparecodelist(_cstFileType=DATASET,_cstBaseCT=oldct.cterms,_cstNewCT=newct.cterms,_cstCompareCL=Y,_cstCLVar=codelist,_cstCompareCLI=Y,_cstCLValueVar=cdisc_submission_value,_cstRptType=_CSTRESULTSDS);

For more information about the macros and their parameters, see the SAS Clinical Standards Toolkit: Macro API Documentation.
The following display shows the data set created by the cstutilcomparecodelists macro:

**Display 3.4  Controlled Terminology Package Differences Reported by the cstutilcomparecodelists Macro**

<table>
<thead>
<tr>
<th>message</th>
<th>actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 Codelist found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=AGESPAN</td>
</tr>
<tr>
<td>3 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=AGESPAN,Value=ADOLESCENT (12-17 YEARS)</td>
</tr>
<tr>
<td>4 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=AGESPAN,Value=ADULT (18-65)</td>
</tr>
<tr>
<td>5 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=AGESPAN,Value=CHILDREN (2-11 YEARS)</td>
</tr>
<tr>
<td>6 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=AGESPAN,Value=ELDERLY (&gt; 65)</td>
</tr>
<tr>
<td>7 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=AGESPAN,Value=IN UTERO</td>
</tr>
<tr>
<td>8 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=AGESPAN,Value=INFANT AND TODDLER (28 DAYS - 23 MONTHS)</td>
</tr>
<tr>
<td>9 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=AGESPAN,Value=NEWBORN (0-27 DAYS)</td>
</tr>
<tr>
<td>10 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=AGESPAN,Value=PRETERM NEWBORN INFANTS</td>
</tr>
<tr>
<td>11 Codelist found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=CCINVCTYP</td>
</tr>
<tr>
<td>12 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=CCINVCTYP,Value=CASUAL CONTACT</td>
</tr>
<tr>
<td>13 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=CCINVCTYP,Value=CONTACT WITH EXPOSURE DURING MEDICAL PROCEDURE</td>
</tr>
<tr>
<td>14 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=CCINVCTYP,Value=CONTACT WITH EXPOSURE IN CROWD SETTING</td>
</tr>
<tr>
<td>15 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=CCINVCTYP,Value=CONTACT WITH MEDICAL RISK FACTOR</td>
</tr>
<tr>
<td>16 Codelist value found in oldct.ctems but not found in newct.ctems</td>
<td>Codelist=CCINVCTYP,Value=HOUSEHOLD CONTACT</td>
</tr>
<tr>
<td>17 Codelist found in newct.ctems but not found in oldct.ctems</td>
<td>Codelist=CCINVCTYP</td>
</tr>
<tr>
<td>18 Codelist value found in newct.ctems but not found in oldct.ctems</td>
<td>Codelist=CCINVCTYP,Value=CASUAL CONTACT</td>
</tr>
<tr>
<td>19 Codelist value found in newct.ctems but not found in oldct.ctems</td>
<td>Codelist=CCINVCTYP,Value=CONTACT WITH EXPOSURE DURING MEDICAL PROCEDURE</td>
</tr>
</tbody>
</table>

Rows 2 through 10 indicate that the AGESPAN codelist (including all values) has been removed from the CDISC-SDTM 201312 controlled terminology package. A comparison of rows 11 through 13 to rows 17 through 19 suggest that the CCINVCTYP codelist has been renamed CCINVTYP.
Compare Two Macro Autocall Libraries

Overview

To help you determine the differences in a library of SAS macros between two versions of the SAS Clinical Standards Toolkit, SAS supplies the cstutilcompareautocallmacros macro.

cstutilcompareautocallmacros Macro

The cstutilcompareautocallmacros macro creates a list of the contents of the directory for each of the two macro libraries.

The cstutilcompareautocallmacros macro compares the two lists and identifies any file that is contained in one library but not the other library. For each file that is common to both macro libraries, the macro indicates any changes in the macro signature.

The following example compares a customized framework macro library to the framework macro library for the installed version of the SAS Clinical Standards Toolkit:

```sas
%cstutilcompareautocallmacros(
   _cstBasePath=C:\Program Files\SASHome\SASFoundation\9.4\cstframework_1.5customized\sasmacro,
   _cstNewPath=C:\Program Files\SASHome\SASFoundation\9.4\cstframework\sasmacro,
   _cstRptType=_CSTRESULTSDS,
   _cstOverwrite=N
);
```

For more information about the macros and their parameters, see the SAS Clinical Standards Toolkit: Macro API Documentation.
The following display shows the data set created by the cstutilcompareautocallmacros macro (not all columns and rows are shown):

**Display 3.5**  Sample Results Data Set Created with the cstutilcompareautocallmacros Macro

<table>
<thead>
<tr>
<th>message</th>
<th>resultSeverity</th>
<th>keyValues</th>
</tr>
</thead>
<tbody>
<tr>
<td>21 New macro (cstutilkmlvalidate_sas) not found in C:\Program Files\SASHome\SASFoundation\3.4\cstframework\1.5\customized\sasmacro</td>
<td>Info</td>
<td>Macro name=cstutilkmlvalidate_sas</td>
</tr>
<tr>
<td>22 Old macro (cstutilkmlvalidateintro_sas) not found in C:\Program Files\SASHome\SASFoundation\3.4\cstframework\1.5\customized\sasmacro</td>
<td>Info</td>
<td>Macro name=cstutilkmlvalidateintro_sas</td>
</tr>
<tr>
<td>23 New macro parameter(s) detected for: cst_createatablesfordatastandard.sas  see work.macrodiffs</td>
<td>Info</td>
<td>Macro name=cst_createatablesfordatastandard.sas  see work.macrodiffs</td>
</tr>
<tr>
<td>24 New macro parameter(s) detected for: cstutil_deletedataset.sas  see work.macrodiffs</td>
<td>Info</td>
<td>Macro name=cstutil_deletedataset.sas  see work.macrodiffs</td>
</tr>
</tbody>
</table>

Rows 21 and 22 indicate that there are differences in the files in the two macro libraries.

The following display shows that rows 23 and 24 indicate that the macro signatures have changed. These changes are itemized in the WORK.MACRODIFFS data set.

**Display 3.6**  Macro Signature Differences Identified by the cstutilcompareautocallmacros Macro

<table>
<thead>
<tr>
<th>name</th>
<th>parameter</th>
<th>baseSig</th>
<th>newSig</th>
</tr>
</thead>
<tbody>
<tr>
<td>cst_createatablesfordatastandard.sas</td>
<td>_cstWhereClause</td>
<td>%macro cst_createatablesfordatastandard(                          _cstStandard=_cstStandardVersion=, _cstOutputLibrary=, _cstResultsOverrideDS=_cstNumObs=0 ) / des='CST: Creates tables from registered referencemetadata';</td>
<td>%macro cst_createatablesfordatastandard(                          _cstStandard=_cstStandardVersion=, _cstOutputLibrary=, _cstResultsOverrideDS=_cstNumObs=0 ) / des='CST: Creates tables from registered referencemetadata';</td>
</tr>
<tr>
<td>cstutil_deletedataset.sas</td>
<td>_cstLogging</td>
<td>%macro cstutil_deletedataset( _cstDatasetName= ) / des='CST: Delete a SAS data set';</td>
<td>%macro cstutil_deletedataset( _cstDatasetName= ) / des='CST: Delete a SAS data set';</td>
</tr>
<tr>
<td>cstutil_deletedataset.sas</td>
<td>_cstLoggingDS</td>
<td>%macro cstutil_deletedataset( _cstDatasetName= ) / des='CST: Delete a SAS data set';</td>
<td>%macro cstutil_deletedataset( _cstDatasetName= ) / des='CST: Delete a SAS data set';</td>
</tr>
</tbody>
</table>

**Compare Two Property Files**

**Overview**

A key element of the SAS Clinical Standards Toolkit is the set of global macro variables built into and used by the SAS Clinical Standards Toolkit. These global macro variables are primarily defined in property files.
For example, the global standards library directory/standards/cdisc-sdtm-3.1.3-1.7/programs/initialize.properties file includes the following name-value pairs that become global macro variables:

- _cstStandard=CDISC-SDTM
- _cstStandardVersion=3.1.3
- _cstSubjectColumns=studyid usubjid
- _cstTableMetadata=work._csttablemetadata
- _cstColumnMetadata=work._cstcolumnmetadata

To help you determine whether there are changes in the global macro variables from one version of the SAS Clinical Standards Toolkit to another version, SAS supplies the cstutilcompareproperties macro.

**cstutilcompareproperties Macro**

The cstutilcompareproperties macro translates property files into SAS data sets for comparison.

The following example compares the initialize.properties file for CDISC SDTM 3.1.3 to 3.2 in the SAS Clinical Standards Toolkit 1.7:

```sas
%cstutilcompareproperties(
   _cstBasePath=C:\cstGlobalLibrary\standards\cdisc-sdtm-3.1.3-1.7\programs\initialize.properties,
   _cstNewPath=C:\cstGlobalLibrary\standards\cdisc-sdtm-3.2-1.7\programs\initialize.properties
);
```

For more information about the macros and their parameters, see the *SAS Clinical Standards Toolkit: Macro API Documentation*.

The following example macro call outputs a message similar to this one in the SAS log file:

```
[CSTLOGMESSAGE] NOTE:  The following properties differ between the two property files:
Property values different between BASE and NEW.
Name:  _cstStandardVersion Base Value: 3.1.3 New Value: 3.2
```
Compare Two Folder Hierarchies

Overview
To help you determine whether two directories contain the same folder hierarchy (including subfolders), SAS supplies the cstutilcomparefolderhierarchy macro.

**TIP** For the best results, the subfolders in each directory must share a similar naming convention or structure, although this is not required.

cstutilcomparefolderhierarchy Macro
The cstutilcomparefolderhierarchy macro can compare these items:

- **folders**
  The macro identifies the presence or absence of subfolders in the two folder hierarchies.

- **files**
  The macro identifies the presence or absence of files in the two folder hierarchies.

- **data sets**
  The macro identifies, based on the output of PROC COMPARE, the differences in the data sets that are common in the two folder hierarchies.

These comparisons can be run individually or simultaneously.

Example 1: Compare Global Standards Library Folder Hierarchies
The following example compares the SAS Clinical Standards Toolkit 1.5 global standards library folder hierarchy (C:\cstGlobalLibrary15custom) to the 1.6 folder
hierarchy (C:\cstGlobalLibrary) to determine which folders differ in the two versions:

```sas
%cstutilcomparefolderhierarchy(
   _cstBaseFolder=C:\cstGlobalLibrary,
   _cstBaseVersion=1.6,
   _cstCompFolder=C:\cstGlobalLibrary15custom,
   _cstCompVersion=1.5,
   _cstRptDiff=work.foldercomp,
   _cstRptDiffType=FOLDER,
   _cstOverwrite=y,
   _cstOutReportPath=C:\,
   _cstOutReportFile=globallibrarydiff15.htm,
);
```

**Note:** It does not matter which folder is `_cstBaseFolder` or `_cstCompFolder`.

For more information about the macros and their parameters, see the SAS *Clinical Standards Toolkit: Macro API Documentation*. 
The following display shows part of the work.foldercomp data set created by the example:

### Display 3.7  Part of the work.foldercomp Data Set Created by the Example

<table>
<thead>
<tr>
<th>BASE Folder name</th>
<th>COMP Folder name</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
<tr>
<td>C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>C:\cstGlobalLibrary15custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats</td>
<td>Folder present in Base only</td>
</tr>
</tbody>
</table>

**Note:** Data shown might not be on your computer.

The first 19 rows identify the new folders in the base folder hierarchy. These folders represent new content in the SAS Clinical Standards Toolkit 1.6 that is not in 1.5.

Rows 20, 21, 25, 27, and 29 identify the folders in the comparison folder hierarchy (in the COMP Folder name column) that are not in the base folder hierarchy. (See the highlighted text in the previous display.)

For example, `C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104Extended\formats` (row 20) does not have a corresponding folder structure in 1.6 because this subfolder was created by the user to contain extended controlled terminology packages.
Likewise, C:\cstGlobalLibrary15Custom\standards\cdisc-terminology-1.5\cdisc-cdash\201104\formats (row 21) does not have a corresponding folder structure in 1.6 because it was deprecated and replaced with a newer version (\201312).

**Note:** It is important to identify differences because you must decide whether to migrate these files into the SAS Clinical Standards Toolkit 1.7 or replace them with the newer 1.7 files. There are many factors to consider. For example, do the previous controlled terminology packages contain extended codelists? Are the previous controlled terminology packages customized for your organization or for a set of studies?

### Example 2: Compare Global Standards Library Files

The following example compares the SAS Clinical Standards Toolkit 1.5 global standards library folder hierarchy (C:\cstGlobalLibrary15Custom) to the 1.6 folder hierarchy (C:\cstGlobalLibrary) to determine which files differ in the two versions:

```sas
%cstutilcomparefolderhierarchy(
   _cstBaseFolder=C:\cstGlobalLibrary,  
   _cstBaseVersion=1.6,  
   _cstCompFolder=C:\cstGlobalLibrary15Custom,  
   _cstCompVersion=1.5,  
   _cstRptDiff=work.filecomp,  
   _cstReportType=FILE,  
   _cstOverwrite=y
);
```

**Note:** It does not matter which folder is _cstBaseFolder or _cstCompFolder.

For more information about the macros and their parameters, see the *SAS Clinical Standards Toolkit: Macro API Documentation*. 
The following display shows rows 86 through 98 of the work.filecomp data set created by the example:

**Display 3.8  Rows 86 through 98 of the work.filecomp Data Set Created by the Example**

<table>
<thead>
<tr>
<th>BASE Folder name</th>
<th>COMPN Folder name</th>
<th>Name of File</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>86</td>
<td>C:slocals/Standard/clinical/3.2.1 Validation Control</td>
<td>copied😺</td>
<td>File present in Base only</td>
</tr>
<tr>
<td>87</td>
<td>C:slocals/Standard/clinical/3.2.1 Validation Control</td>
<td>validation_chemistry_check_read.txt</td>
<td>File present in Base only</td>
</tr>
<tr>
<td>88</td>
<td>C:slocals/Standard/clinical/3.2.1 Validation Control</td>
<td>validation_project_use.txt</td>
<td>File present in Base only</td>
</tr>
<tr>
<td>89</td>
<td>C:slocals/Standard/clinical/3.2.1 Validation Control</td>
<td>validation_jhtml_oet.txt</td>
<td>File present in Base only</td>
</tr>
<tr>
<td>90</td>
<td>C:slocals/Standard/clinical/3.2.1 Validation Control</td>
<td>sendall_contacted_reniedtids.txt</td>
<td>File present in Base only</td>
</tr>
<tr>
<td>91</td>
<td>C:slocals/library/Security/standard/2019/standards</td>
<td>consent.txt</td>
<td>File present in Com only</td>
</tr>
<tr>
<td>92</td>
<td>C:slocals/library/Security/standard/2019/standards</td>
<td>client_upload.txt</td>
<td>File present in Com only</td>
</tr>
<tr>
<td>93</td>
<td>C:slocals/library/Security/standard/2019/standards</td>
<td>client_upload_read.txt</td>
<td>File present in Com only</td>
</tr>
<tr>
<td>94</td>
<td>C:slocals/library/Security/standard/2019/standards</td>
<td>consent.xml</td>
<td>File present in Com only</td>
</tr>
<tr>
<td>95</td>
<td>C:slocals/library/Security/standard/2019/standards</td>
<td>client.xml</td>
<td>File present in Com only</td>
</tr>
<tr>
<td>96</td>
<td>C:slocals/library/Security/standard/2019/standards</td>
<td>sendall_contacted_reniedtids.txt</td>
<td>File present in Com only</td>
</tr>
<tr>
<td>97</td>
<td>C:slocals/library/Security/standard/2019/standards</td>
<td>consent.txt</td>
<td>File present in Base only</td>
</tr>
<tr>
<td>98</td>
<td>C:slocals/library/Security/standard/2019/standards</td>
<td>client.xml</td>
<td>File present in Base only</td>
</tr>
</tbody>
</table>

**Note:** Data shown might not be on your computer.

Rows 86 through 90 identify the new files in the base folder hierarchy. These files represent new content in the SAS Clinical Standards Toolkit 1.6 that is not in 1.5.

Rows 91 through 96 identify the files from the two folders in the previous example. (See row 20 and 21 **Display 3.7 on page 30.**) Six files are not in the base folder hierarchy (1.6).

**Note:** It is important to identify differences because you must decide whether to migrate such files into the SAS Clinical Standards Toolkit 1.7. If these files are unchanged and have been replaced by the files in 1.7, you probably do not need to migrate them. However, if you have customized these files for certain studies or organizational standards, you might need to migrate these files.

Rows 97 and 98 identify the new files in the base folder hierarchy. These files represent new content in the SAS Clinical Standards Toolkit 1.6 that is not in 1.5.

The following display illustrates rows 1 through 6 of the work.filecomp data set created by the example:

**Display 3.9  Rows 1 through 6 of the work.filecomp Data Set Created by the Example**

<table>
<thead>
<tr>
<th>BASE Folder name</th>
<th>COMPN Folder name</th>
<th>Name of File</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>C:slocals/library</td>
<td>transaction_log.txt</td>
<td>File present in Base only</td>
</tr>
<tr>
<td>2</td>
<td>C:slocals/library</td>
<td>define_extension.xml</td>
<td>File present in Base only</td>
</tr>
<tr>
<td>3</td>
<td>C:slocals/library</td>
<td>define_extension.xml</td>
<td>File present in Base only</td>
</tr>
<tr>
<td>4</td>
<td>C:slocals/library</td>
<td>define_extension.xml</td>
<td>File present in Base only</td>
</tr>
<tr>
<td>5</td>
<td>C:slocals/library</td>
<td>DDI-M1.2.1.xml</td>
<td>File present in Com only</td>
</tr>
<tr>
<td>6</td>
<td>C:slocals/library</td>
<td>DDI-M1.3-readisation.xml</td>
<td>File present in Base only</td>
</tr>
</tbody>
</table>
Rows 1 through 4 and 6 identify the new files in the base folder hierarchy. These files represent new content in the SAS Clinical Standards Toolkit 1.6 that is not in 1.5.

**Note:** You must decide whether these files must be migrated to 1.7.

Row 5 identifies the ODM1-2-2-clean.xml file that is not in 1.6.

---

## Copy a Folder Hierarchy

### Overview

To help you copy a folder, all of its subfolders, and all content, SAS supplies the `cstutilcopyfolderhierarchy` macro.

Although you can copy a folder hierarchy using operating system commands or tools, the `cstutilcopyfolderhierarchy` macro provides useful options, such as itemization of the folders and files copied and summarization of the actions performed.

### `cstutilcopyfolderhierarchy` Macro

The `cstutilcopyfolderhierarchy` macro creates a list of the folders and files contained by a folder hierarchy. Two data sets are created: one that contains the folders and one that contains the files. You can use these data sets for other programs, such as renaming the folders and files.

The macro offers an option to copy only the source folder hierarchy without also copying the files contained in the source folders.

The following example copies the source folder hierarchy `\d78398\public\cstGlobalLibrary` to `C:\cstGlobalLibrary15`:

```sas
%cstutilcopyfolderhierarchy(
   _cstSourceFolder=\d78398\public\cstGlobalLibrary,
   _cstNewFolder=C:\cstGlobalLibrary15,
   _cstFolderDS=work.folders,
   _cstFileDS=work.files,
   _cstBuildFoldersOnly=N
);
```
For more information about the macros and their parameters, see the SAS Clinical Standards Toolkit: Macro API Documentation.

The example macro call outputs a message similar to this one in the SAS log file:

```
NOTE: [CSTLOGMESSAGE] 191 folders were created.
NOTE: [CSTLOGMESSAGE] 1049 files were copied.
```

The following display shows the data set that contains the folders copied by the example:

**Display 3.10  Sample Data Set That Contains the Folders Copied by the cstutilcopyfolderhierarchy Macro**

The following display shows the data set that contains the files copied by the example:

**Display 3.11  Sample Data Set That Contains the Files Copied by the cstutilcopyfolderhierarchy Macro**
Introduction

SAS supplies tools to support the migration of study source metadata that was created for a CRT-DDS 1.0.0 define.xml file to study source metadata to be used to create a Define-XML 2.0.0 define.xml file.

Note: Be aware that the use of these tools results only in an approximation of Define-XML 2.0.0 study source metadata. No assumptions should be made that the results completely represent the study source metadata. Incomplete reference metadata might not enable imputation of missing metadata. Some metadata must be added or updated through a user-defined process. In particular, ValueList definitions and WHERE clauses are converted from value-level metadata definitions in CRT-DDS 1.0 that have implicit assumptions about the variable that is described or the variable that defines the condition.

The macros mentioned in this chapter are located in the global standards library directory/standards/cdisc-definexml-2.0.0-1.7/macros folder.

## Overview

When the SAS Clinical Standards Toolkit creates a define.xml file for CRT-DDS 1.0.0 or Define-XML 2.0.0, this is the process:

1. Create data sets that contain the study source metadata.
2. Convert the study source metadata into the SAS representation of the CRT-DDS 1.0.0 or Define-XML 2.0.0 model.
3. Extend the SAS representation of the model, if needed.
   - For Define-XML 2.0.0, this step is not usually needed because in most cases the study source metadata can contain the complete metadata content for the creation of the Define-XML 2.0.0 file. The tools in this chapter assume that there are data sets with study source metadata for an ADaM or SDTM study to use to create a CRT-DDS 1.0.0 define.xml file.
   - **Note:** If the SAS representation of the model was extended to create a define.xml file for CRT-DDS 1.0.0, you must ensure that the study source metadata created by this step is added to the Define-XML 2.0.0 study source metadata. The conversion tool uses only the study source metadata data sets, not the SAS representation of the CRT-DDS 1.0.0 model.
4. Create a define.xml file from the SAS representation of the CRT-DDS 1.0.0 or Define-XML 2.0.0 model.

For CRT-DDS 1.0.0, the following study source metadata data sets are defined in the SAS Clinical Standards Toolkit 1.5 or later:

- `source_study`
- `source_tables`
For Define-XML 2.0.0, a new study source metadata data set (source_codelists) has been defined. The source_codelists data set was not part of the CRT-DDS 1.0.0 study source metadata data sets because SAS formats were used as input. The source_codelists data set contains all metadata needed to create the codelists in the define.xml file, including the external codelists (for example, MedDRA and WHODRUG) and NCI metadata (for example, the so-called C-codes). The cstutilgetncimetadata macro creates the source_codelists data set from a list of SAS format catalogs that define the study formats and also a SAS data set that contains CDISC/NCI codelist metadata.

**Study Source Metadata Migration Driver Programs**

To provide an example of migrating CRT-DDS 1.0.0 source metadata to Define-XML 2.0.0 source metadata, SAS supplies the migrate_crtdds_to_definexml_adam.sas and migrate_crtdds_to_definexml_sdtm.sas driver programs. These driver programs are located in the *sample study library directory/cdisc-definexml-2.0.0-1.7/programs* folder.

**Note:** The driver programs for ADaM and SDTM are similar in structure, so this example addresses only the SDTM driver program.

After the driver program performs the initial setup (which you can review in the driver program), the following librefs are defined:

```
%* Define libnames for input
%* Original CRT-DDS v1 source metadata for SDTM 3.1.3;
libname crtdds "&studyRootPath/sascstdemodata/metadata";
```
%* Define libnames for output *
%**********************************************************************************;
%* Migrated Define-XML v2 source metadata;
libname defv2 "&studyOutputPath/derivedstudymetadata_crtdds/";
   %lowcase(&_cstTrgStandard)-&_cstTrgStandardVersion";
%**********************************************************************************;
%* Define formats *
%**********************************************************************************;
%* SDTM Study formats in CST 1.7;
libname studyfmt "&studyRootPath/sascstdemodata/terminology/formats";
%* CDISC-NCI Terminology to be used in CST 1.7;
%cst_getstandardsubtypes(_cstStandard=CDISC-TERMINOLOGY,_cstOutputDS=work._cstStdSubTypes);
 data _null_;
  set work._cstStdSubTypes (where=(standardversion="&_cstTrgStandard" and isstandarddefault='Y'));
  * User can override CT version of interest by specifying a different where clause: *
  * Example: (where=(standardversion="&_cstTrgStandard" and standardsubtypeversion='201104'))*
  call symputx('_cstCTPath',path);
  call symputx('_cstCTMemname',memname);
run;
proc datasets lib=work nolist;
  delete _cstStdSubTypes;
quit;
%* CDISC-NCI Terminology to be used in CST 1.7;
libname ncisdtm "&_cstCTPath";
%* Formats to be used for SDTM;
options fmtsearch = (studyfmt.formats ncisdtm.&_cstCTMemname);

**Note:** You might need to change these librefs.

Some of the CRT-DDS 1.0.0 metadata must be mapped to the values expected by Define-XML 2.0.0. It is likely that you must change the formats based on your specific data values. It is important to use the format names as specified because these formats are used by the conversion macros.

%**********************************************************************************;
%* Create some formats for mapping *
%**********************************************************************************;
proc format;
 value $_cststd
   /* Maps from CRT-DDS values to required Define-XML v2 values */
   "CDISC SDTM"="SDTM-IG"
"CDISC SEND"="SEND-IG"
"CDISC ADAM"="ADAM-IG"
;
value $_cstdom
    /* Map to ItemGroup/@Domain attribute */
    "QSCG" = "QS"
    "QSCS" = "QS"
    "QSMM" = "QS"
;
value $_cstdomd
    /* Map to ItemGroup/Alias[@Context='DomainDescription']/@Name attribute */
    "QSCG" = "Questionnaires"
    "QSCS" = "Questionnaires"
    "QSMM" = "Questionnaires"
;
value $_cstcls
    /* Maps from CRT-DDS values to required Define-XML v2 values */
    "SPECIAL PURPOSE DOMAINS" = "SPECIAL PURPOSE"
    "SPECIAL PURPOSE DATASETS" = "SPECIAL PURPOSE"
    "FINDINGS ABOUT" = "FINDINGS"
    "ADSL" = "SUBJECT LEVEL ANALYSIS DATASET"
    "ADAE" = "ADAM OTHER"
    "BDS" = "BASIC DATA STRUCTURE"
;
value $_cstvlm
    /* For SDTM maps to variables that are being described by Value Level Metadata */
    "EG.EGTESTCD" = "EGORRES"
    "IE.IETESTCD" = "IEORRES"
    "TI.IETESTCD" = "IECAT"
    "LB.LBTESTCD" = "LBORRES"
    "PE.PETESTCD" = "PEORRES"
    "SC.SCTESTCD" = "SCORRES"
    "VS.VSTESTCD" = "VSORRES"
    "SUPPAE.QNAM" = "QVAL"
;
run;

Now, the metadata is converted.

%**********************************************************************************;
%* Define the studyversion macro variable.                                      *
%* This will become the MetaDataVersion/@OID attribute                         *
%* In CRT-DDS this was the source_study.definedocumentname column             *
%* Also define the SASRef macro variable to use for the SASRef column in the  *
%* source_xxx data sets.                                                      *
%**********************************************************************************;
proc sql noprint;
  select definedocumentname, SASRef into :studyversion, :SASRef
Now, the source_codelists table is created. This is a separate process because this table was not in the CRT-DDS 1.0.0 source metadata.
if xmlcodelist ne ''; run;

proc sort data=work.cl_column_value nodupkey;
   by xmlcodelist;
run;

/* Only keep applicable formats. ;
proc sql;
   create table defv2.source_codelists
   as select
      nci.*
   from
      work._cstformats nci, work.cl_column_value cv
   where (upcase(compress(nci.sasformatname, '$')) =
      upcase(compress(cv.xmlcodelist, '$')))
   ;
quit;

Note: The _cstFormatCatalogs parameter is blank. This indicates that the format
catalogs that define the codelists to include in the source_codelists table are taken from
the value specified for the FMTTSEARCH option.

Finally, the metadata for external controlled terminology is added to the
source_codelists data set.

%**********************************************************************************;
%* Updates for External Controlled Terminology                                    *
%**********************************************************************************;
proc sql;
insert into defv2.source_codelists
   (sasref, codelist, codelistname, codelistdatatype, dictionary, version,
   studyversion, standard, standardversion)
values ("&SASRef", "CL.AEDICT", "Adverse Event Dictionary", "text", "MEDDRA", "8.0",
   &(studyversion), "&_cstTrgStandard", "&_cstTrgStandardVersion")
values ("&SASRef", "CL.DRUGDCT", "Drug Dictionary", "text", "WHODRUG", "200204",
   &(studyversion), "&_cstTrgStandard", "&_cstTrgStandardVersion")
;
quit;

data defv2.source_columns;
   set defv2.source_columns;
   if table="AE" and column in ("AEDECOD" "AEBODSYS") then xmlcodelist="CL.AEDICT";
   if table="CM" and column in ("CMDECOD" "CMCLAS" "CMCLASCD")
      then xmlcodelist="CL.DRUGDCT";
run;
For more information about the macros and their parameters, see the SAS Clinical Standards Toolkit: Macro API Documentation.
Frequently Asked Questions

Installation Issues .................................................. 44
  Backup Copies .................................................. 44
  Overwrite Previous Versions of the SAS Clinical Standards Toolkit ........................................... 44
  Install the SAS Clinical Standards Toolkit 1.7 in Conjunction with a Previous Version .................. 44
  Upgrade Prior Versions of the SAS Clinical Standards Toolkit ..................................................... 45

Migration Process .................................................. 45
  Order of Migration .............................................. 45

Standards Management ............................................. 46
  Unused Standards ............................................... 46

Library Issues ....................................................... 46
  Delete Sample Study Library ................................ 46
  Maintain Metadata Attributes ............................... 47
  cstGlobalLibrary Standards Folder ....................... 47
  Update Customized Standard Metadata across Global Standards Libraries .............................. 48

Other Issues ........................................................ 48
  Driver Programs ............................................... 48
Installation Issues

Backup Copies

Is it necessary to back up my copy of a previous version of the SAS Clinical Standards Toolkit before I install version 1.7? If I have customized my SAS Clinical Standards Toolkit, what must I change in the configuration file?

If you have customized any of the three locations that the SAS Clinical Standards Toolkit writes to during the installation process, you must back up the files. (For more information about these locations, see Chapter 1, “Installation Locations,” on page 1.)

A new installation of SAS does not typically replace the old configuration file. If you have concerns about modifications that you made to the configuration file, back up the file to ensure that you can restore it.

Overwrite Previous Versions of the SAS Clinical Standards Toolkit

Will an installation of the SAS Clinical Standards Toolkit 1.7 overwrite my previous version?

Only partially. As described in the Chapter 1, “Installation Locations,” on page 1, the SAS Clinical Standards Toolkit places files in three locations. Only files in the !sasroot location can be overwritten.

For a full description of an installation process that requires you to install the global standards library and sample study library to other locations, see the Chapter 2, “SAS Deployment Wizard,” on page 5.

Install the SAS Clinical Standards Toolkit 1.7 in Conjunction with a Previous Version

Can I install the SAS Clinical Standards Toolkit 1.7 in conjunction with a previous version?
Yes. For more information, see “Upgrade to SAS Clinical Standards Toolkit 1.7 By Installing SAS 9.4 in Conjunction with SAS 9.3” on page 9.

Upgrade Prior Versions of the SAS Clinical Standards Toolkit

I am running a previous version of the SAS Clinical Standards Toolkit. Do I need to separately upgrade to each version to get up-to-date to version 1.7?

No. If you did not customize a previous version of the SAS Clinical Standards Toolkit, and you do not require access to any functionality that has been deprecated (such as framework autocall macros and older controlled terminology packages), you can use the full functionality provided by the SAS Clinical Standards Toolkit 1.7.

Note: Deprecations are generally noted in the “What’s New” section of the SAS Clinical Standards Toolkit: User’s Guide for each version.

If you customized a previous version of the SAS Clinical Standards Toolkit, run the migration tools to identify file differences that you might need to address.

Migration Process

Order of Migration

Is there a suggested order to migrate data? Is all or most of the work centered on the SAS Clinical Standards Toolkit global standards library?

Yes, this is the suggested order to migrate data:

1. Migrate user customizations that affect the global standards library. New standards or customized versions of standards are registered in the global standards library. Copy these user customizations into the SAS Clinical Standards Toolkit global standards library folder hierarchy and register them to the current version.

2. If recommended best practices were not followed, migrate user customizations to the $sasroot autocall library.
3 Migrate changes to the sample study library. These changes are least likely to be impacted unless you have chosen to build a sample study for your customized standards or you have modified the driver programs. A typical modification to a driver program is to reference a new sample study root path location.

### Standards Management

#### Unused Standards

Can I delete from the global standards library any unused standards that are supplied with the SAS Clinical Standards Toolkit? If I delete the folders, does this affect any of my metadata files in the framework?

Run the `cst_unregisterstandard` macro for each unwanted standard and standardversion. This macro does not remove the associated folder hierarchies from the global standards library and the sample study library. You can delete the folder hierarchies later.

Some framework sample driver programs that support internal validation include references to all standards and standard versions that are supplied by SAS. These references typically appear in WHERE clauses. These WHERE clauses return fewer records (standards or standard versions) to validate.

### Library Issues

#### Delete Sample Study Library

Do I need the SAS Clinical Standards Toolkit `cstSampleLibrary` or can I delete it?

The sample study library provides sample study implementations (including metadata, data, and code) to demonstrate the SAS Clinical Standards Toolkit functionality for various standards that are supplied by SAS. Without the sample study implementations,
all sample processes and internal validation checks fail. Furthermore, product
documentation assumes its presence.

However, the sample study library is not necessary in a mature SAS Clinical Standards
Toolkit environment in which users have their own processes that point to their own
study data. The studylibraryrootpath column in the global standards library
directory/metadata/standards data set (if used) is probably set to some study
repository root path location outside the sample study library folder hierarchy.

Maintain Metadata Attributes

I have used the default global standards library for metadata attributes in all of my
studies. For example, I populated my source_* data sets with default information. What
must I do to maintain the changes to the source metadata that I have?

Most changes in the reference metadata representations of each CDISC standard
reflect updates made by CDISC. If you do not already have a process in place to
identify and review these changes with each CDISC standard version, you can compare
a previous version of the SAS Clinical Standards Toolkit and version 1.7 reference
metadata for each standard or standardversion of interest. After the changes are
identified and reviewed, update any source_* data sets to reflect the changes.

A suite of tools to identify differences in folder hierarchies and files is described in

cstGlobalLibrary Standards Folder

I notice in the cstGlobalLibrary standards folder that the SAS Clinical Standards Toolkit
version is appended to the folder. I have many changes to the SDTM 3.1.3 standard
directory in my previous version of the SAS Clinical Standards Toolkit. Can I continue to
use that folder and ignore the SDTM 3.1.3 standard folder in version 1.7 because I want
to preserve my changes?

Yes. Consider copying and registering the SDTM 3.1.3 standard folder in your previous
version of the SAS Clinical Standards Toolkit as a custom version of SDTM 3.1.3 in
version 1.7. No structural changes to metadata in the SAS Clinical Standards Toolkit
versions 1.5, 1.6, and 1.7 should interfere with this strategy.
Update Customized Standard Metadata across Global Standards Libraries

I work at a CRO. We have more than 30 global standards libraries that contain customized standard metadata for customers. What is the easiest way for me to update any information that is specific to the SAS Clinical Standards Toolkit 1.7 across these global standards libraries? Is it a manual process? Are there any available tools that can aid in this process?

It is not possible to provide definitive migration guidance without knowing what metadata has been customized and whether these customizations are in the SAS deployment locations for the SAS Clinical Standards Toolkit.

However, your migration process will be simpler if you have standardized your customizations across global standards libraries and if your customizations have not been made to the SAS deployment locations for the SAS Clinical Standards Toolkit.

Other Issues

Driver Programs

Will the driver programs that I created for my studies in a previous version of the SAS Clinical Standards Toolkit run without any changes in version 1.7? If not, what do I need to look for?

Although it is not possible to answer this question with confidence without knowing the details of your driver program customizations, SAS expects most driver programs to work in the most recent versions and the current version with no or minimal modification.

These coding changes might be required, depending on your coding style and macro use:
If you have hardcoded the SAS Clinical Standards Toolkit version or if you have hardcoded the location of the sample study library, consider replacing the hardcoded locations with code such as this:

```sas
%cstutil_setcstsroot;
data _null_;
call symput('studyRootPath',cats("&_cstSRoot",
    "/cdisc-sdtm-&_cstStandardVersion-&_cstVersion/sascstdemodata");
call symput('studyOutputPath',cats("&_cstSRoot",
    "/cdisc-sdtm-&_cstStandardVersion-&_cstVersion/sascstdemodata");
run;
```

If your driver program references deprecated macros or deprecated controlled terminology packages, you must reference alternatives.

If your SASReferences file includes any paths that are no longer valid, correct the paths.
Gain Greater Insight into Your SAS® Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.

support.sas.com/bookstore for additional books and resources.