Paper SAS1826-2018

Is Your Data Viable? Preparing Your Data for SAS® Visual Analytics 8.2

Gregor Herrmann, SAS Institute Inc.

ABSTRACT

We all know that data preparation is crucial before you can derive any value from data through visualization and analytics. SAS® Visual Analytics on SAS® Viya® comes with a new rich HTML5 interface on top of a scalable compute engine that fosters new ways of preparing your data upfront. SAS® Data Preparation that comes with SAS Visual Analytics brings new capabilities like profiling, transposing or joining tables, creating new calculated columns, and scheduling and monitoring jobs. This paper guides you through the enhancements in data preparation with SAS Visual Analytics 8.2 and demonstrates valuable tips for reducing runtimes of your data preparation tasks. It covers integrating existing SAS® 9 tools and programs in your data preparation efforts for SAS Viya.

INTRODUCTION

Sources of data have gone beyond the boundaries of IT-managed enterprise data warehouses. Organizations are facing the flux of ad hoc sources that business users need to make more informed decisions.

One of the key criteria for the successful use of a BI application is being able to import users' ad hoc data sources in a self-service manner for data analysis without depending on IT resources. In addition to access to these ad hoc data sources, there is an increasing need to enhance data suitable for the needs of analysis, without the need for the IT department to make changes to the centralized data source, which can often take a long time.

Business solutions that allow data access and data manipulation for business analysts are gaining more traction. SAS Visual Analytics comes with capabilities that can empower specifically enabled users to bring their own data into the environment and to further refine it by modifying existing data items or adding new data items. The goal of this self-service data management capability is to provide a managed, yet self-service way for users to provision and prepare their own data without always having to rely on IT. The subsequent sections of this document dive into more details about these data preparation tasks. In addition, integrating existing SAS 9 data preparation jobs into a workflow for making data available for analysis in SAS Visual Analytics 8.2 is covered.

SAS VISUAL ANALYTICS ON SAS VIYA TECHNOLOGY OVERVIEW

SAS Visual Analytics delivers analytical visualizations that use intelligent ways to help business analysts and nontechnical users to see patterns and trends and to identify opportunities for further analysis. SAS Visual Analytics is backed by the power of SAS Viya, which is available to users in a self-service and approachable manner. SAS Visual Analytics enables the creation and dissemination of dashboards, reports, and the results of investigative exploration, either to the web or to native mobile applications.

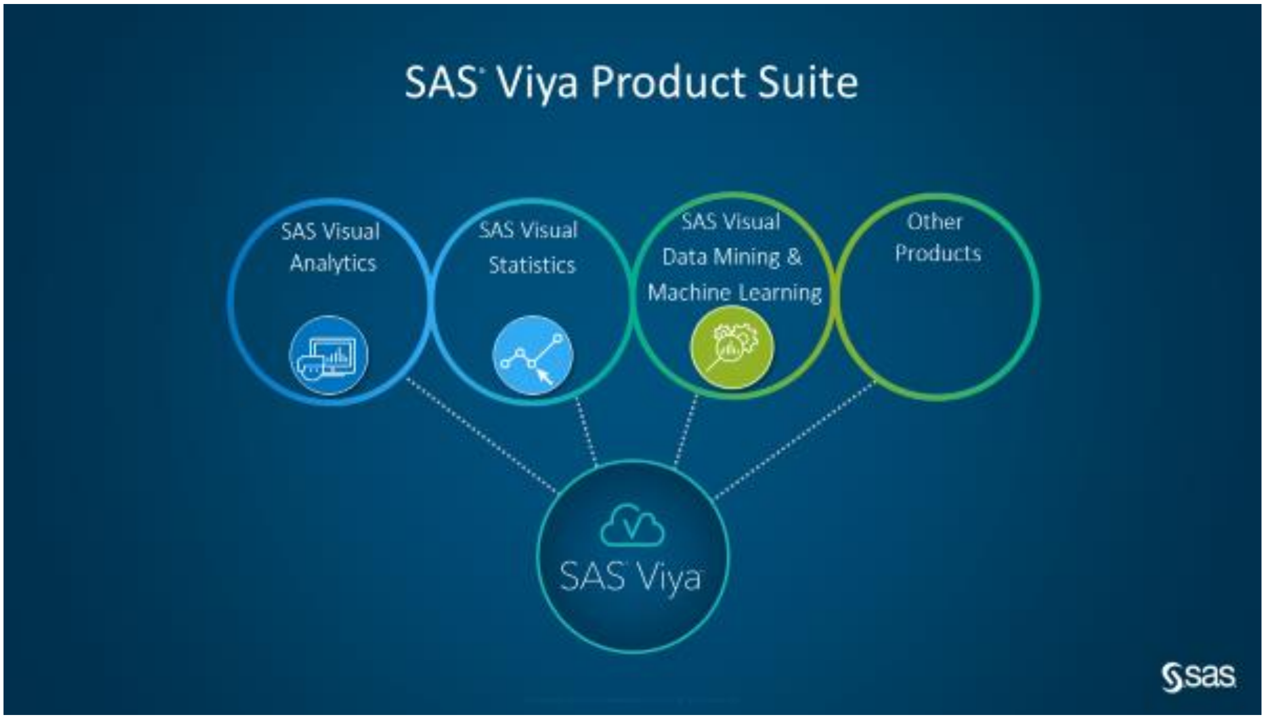


Figure 1. Analytical Visualizations

SAS Visual Analytics content can be augmented by advanced statistical methods and machine learning algorithms in one unified HTML5 interface by using additional modules of the SAS Viya product suite. SAS Visual Analytics includes the capability to prepare data before making it available to users, an interface for exploring your data (often known as data discovery), and an interface for building highly interactive and visual reports and dashboards.

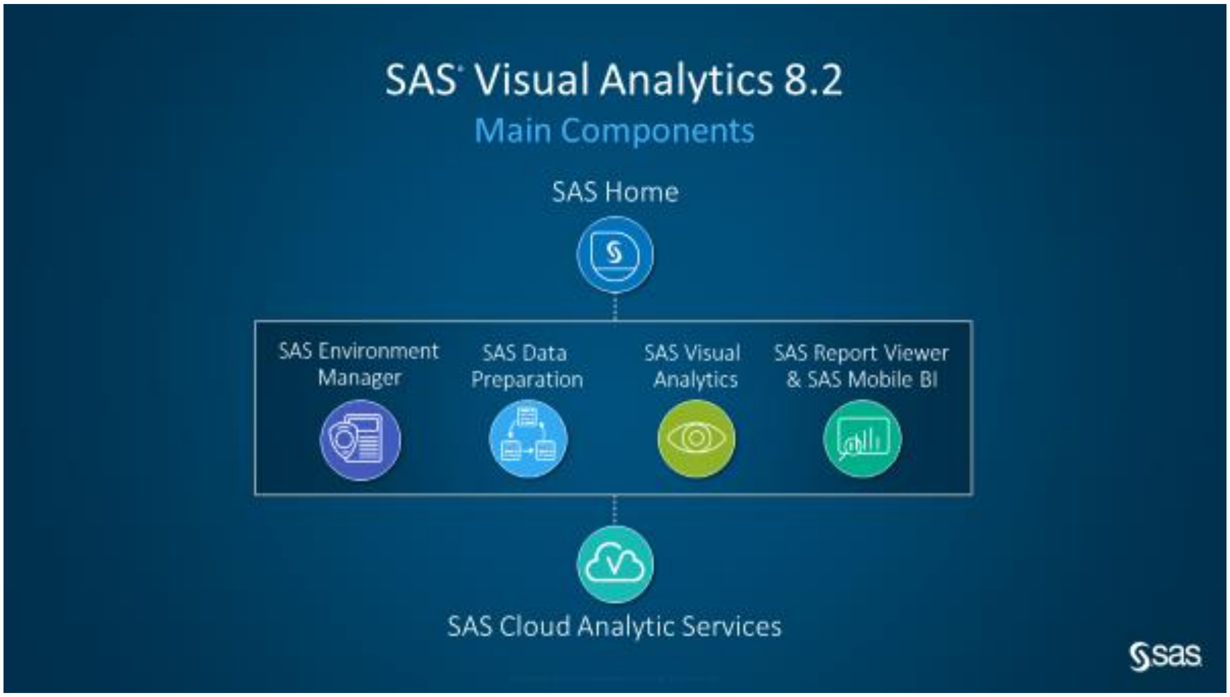


Figure 2. SAS Visual Analytics Main Components

Because SAS Visual Analytics 8.2 is the second release on SAS Viya, it is important to understand the technical differences from the previous versions of SAS Visual Analytics running on SAS 9. Let's have a closer look at the underlying architecture:

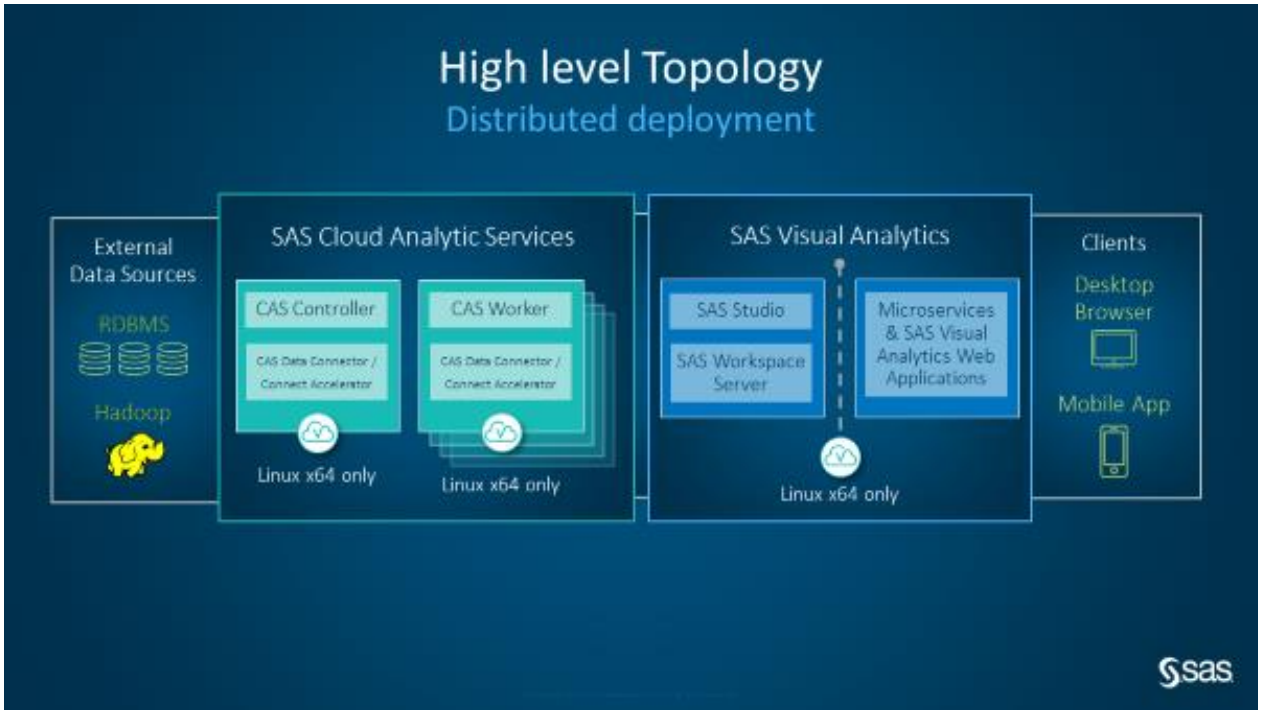


Figure 3. SAS Visual Analytics Topology

Compared to SAS Visual Analytics 7.*x*, the compute engine, SAS® LASR Analytic Server, has been replaced in SAS Visual Analytics in SAS Viya by the CAS server. You can call CAS the next generation in-memory compute server that brings new capabilities like failover and resiliency. The main difference from the SAS LASR Analytic Server is that data preparation is no longer executed by a SAS program on the head node. The CAS server brings its own data preparation capabilities, spreading its workload to all workers across a distributed environment. This allows data management transformations on larger data volumes with short execution times because the operations are done in parallel across the worker nodes.

DATA ACCESS

Data preparation always starts with accessing data. SAS Visual Analytics on SAS Viya supports a large variety of file formats in a standard configuration. Data can be imported from your local file system or from social media, or you can choose from data that is already available on the server.

SUPPORTED FILE FORMATS

Importing Local Files

Local files are files that can be accessed via the operating system of the machine on which you are running your browser to access SAS Visual Analytics. The following file formats can be imported from your local file system:

- Comma-separated values (CSV) files or TXT files.
- SAS data sets (SASHDAT or SAS7BDAT). SAS data set views (SAS7BVEW) cannot be loaded into CAS tables.

- Microsoft Excel workbook (XLSX) files and Excel 97-2003 workbook (XLS) files. You cannot import XLST, XLSB, XLSM, or other Excel file types. You cannot import pivot tables. To import native Microsoft Excel files, SAS Data Connector to PC File Formats is required.

Importing Social Media Data

With regard to social media, SAS Visual Analytics on SAS Viya supports the following data imports:

- Twitter
- Facebook
- Google Analytics
- YouTube
- Google Drive

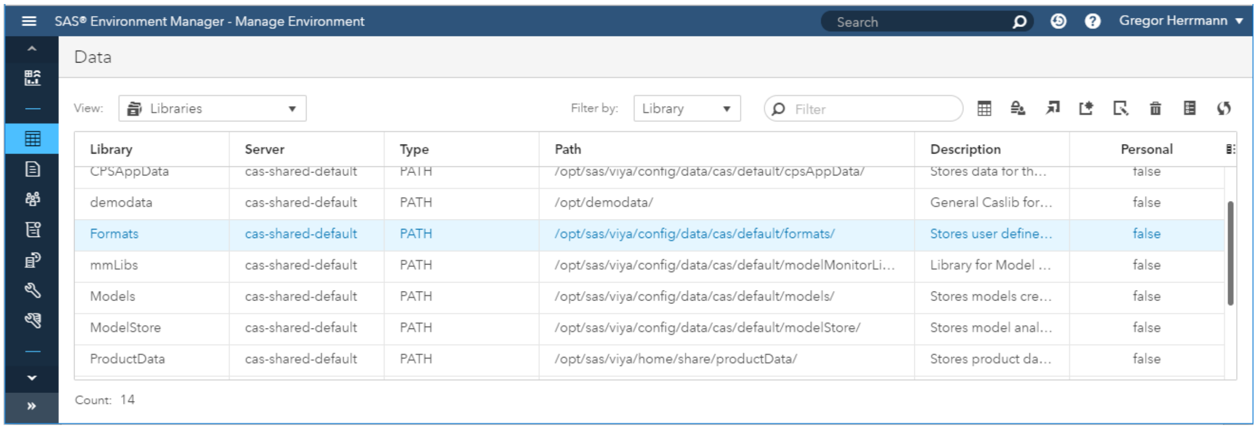
To load data from the different social media channels, you must allow SAS Visual Analytics to access your account.

Accessing Server Files

If your data is already loaded onto the CAS server, it can be accessed via the **Available** data pane in the **Open Data Source** window. (See Figure 5.) Data that is already physically stored on the CAS server, but not yet loaded into memory, can be opened using the **Data Sources** pane of the same window. After clicking on **Data Sources**, a list of available CAS libraries is displayed. Drilling down into one of these libraries shows all available tables within that library. The icon beneath the table name indicates the table type.

- CAS table (a table already in the specific CAS format with extension .sashdat)
- Physical table (a text file usually in CSV format or a SAS 9 file with extension .sas7bdat)
- In-memory table (a table already loaded into memory on the CAS server; this file does not have any extension)

If you are loading SAS 9 tables that contain user-defined formats, they must be made available to your CAS environment before loading. The easiest way is to create a CAS table that contains all your user-defined formats, and then save it to the appropriate caslib. You can see the default settings for the caslib formats in SAS Environment Manager in Figure 4. SAS Environment Manager comes with a nice interface to check the availability of user-defined formats and make modifications if necessary.

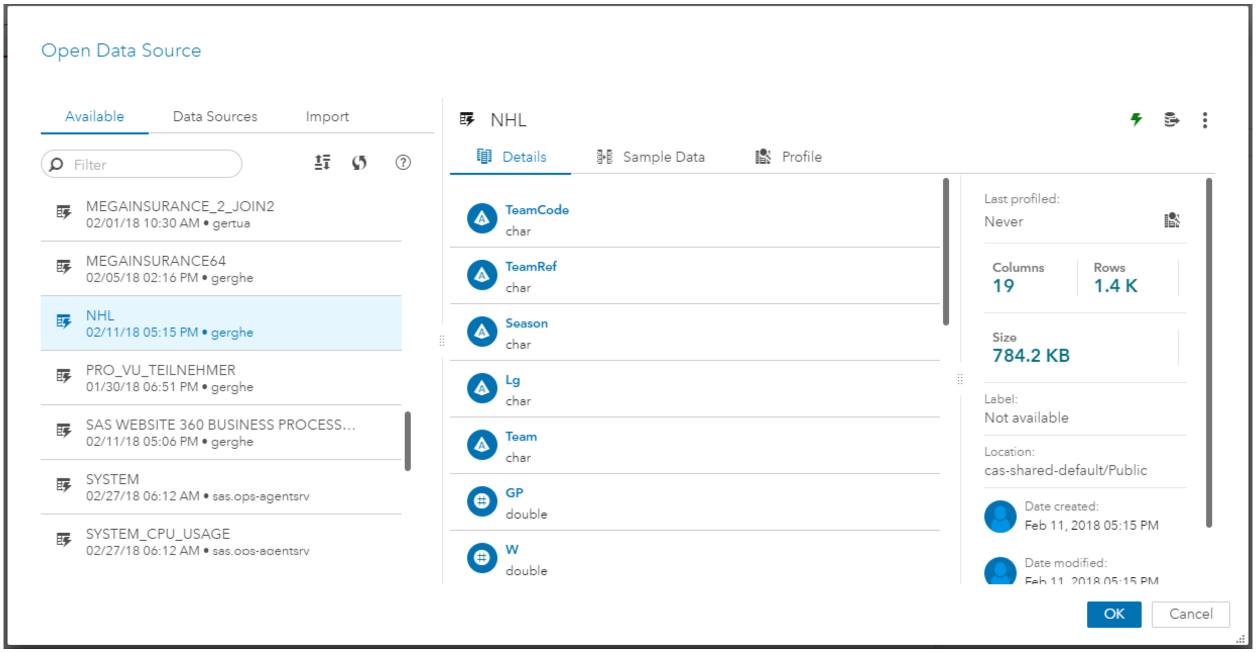


Library	Server	Type	Path	Description	Personal
CPSAppData	cas-shared-default	PATH	/opt/sas/viya/config/data/cas/default/cpsAppData/	Stores data for th…	false
demodata	cas-shared-default	PATH	/opt/demodata/	General Caslib for…	false
Formats	cas-shared-default	PATH	/opt/sas/viya/config/data/cas/default/formats/	Stores user define…	false
mmLibs	cas-shared-default	PATH	/opt/sas/viya/config/data/cas/default/modelMonitorLi…	Library for Model …	false
Models	cas-shared-default	PATH	/opt/sas/viya/config/data/cas/default/models/	Stores models cre…	false
ModelStore	cas-shared-default	PATH	/opt/sas/viya/config/data/cas/default/modelStore/	Stores model anal…	false
ProductData	cas-shared-default	PATH	/opt/sas/viya/home/share/productData/	Stores product da…	false

Count: 14

Figure 4. Default Caslibs in SAS Environment Manager

When you have selected the table that you want to open, you get a short summary of the table, including the number of rows and columns. You can switch to the **Sample Data** or **Profile Panel** to see more details of the content of your selected data source.



MEGAINSURANCE_2_JOIN2
02/01/18 10:30 AM • gertua

MEGAINSURANCE64
02/05/18 02:16 PM • gerghe

NHL
02/11/18 05:15 PM • gerghe

PRO_VU_TEILNEHMER
01/30/18 06:51 PM • gerghe

SAS WEBSITE 360 BUSINESS PROCESS...
02/11/18 05:06 PM • gerghe

SYSTEM
02/27/18 06:12 AM • sas.ops-agentsrv

SYSTEM_CPU_USAGE
02/27/18 06:12 AM • sas.ops-agentsrv

NHL

TeamCode	char
TeamRef	char
Season	char
Lg	char
Team	char
GP	double
W	double

Last profiled:
Never

Columns
19

Rows
1.4 K

Size
784.2 KB

Label:
Not available

Location:
cas-shared-default/Public

Date created:
Feb 11, 2018 05:15 PM

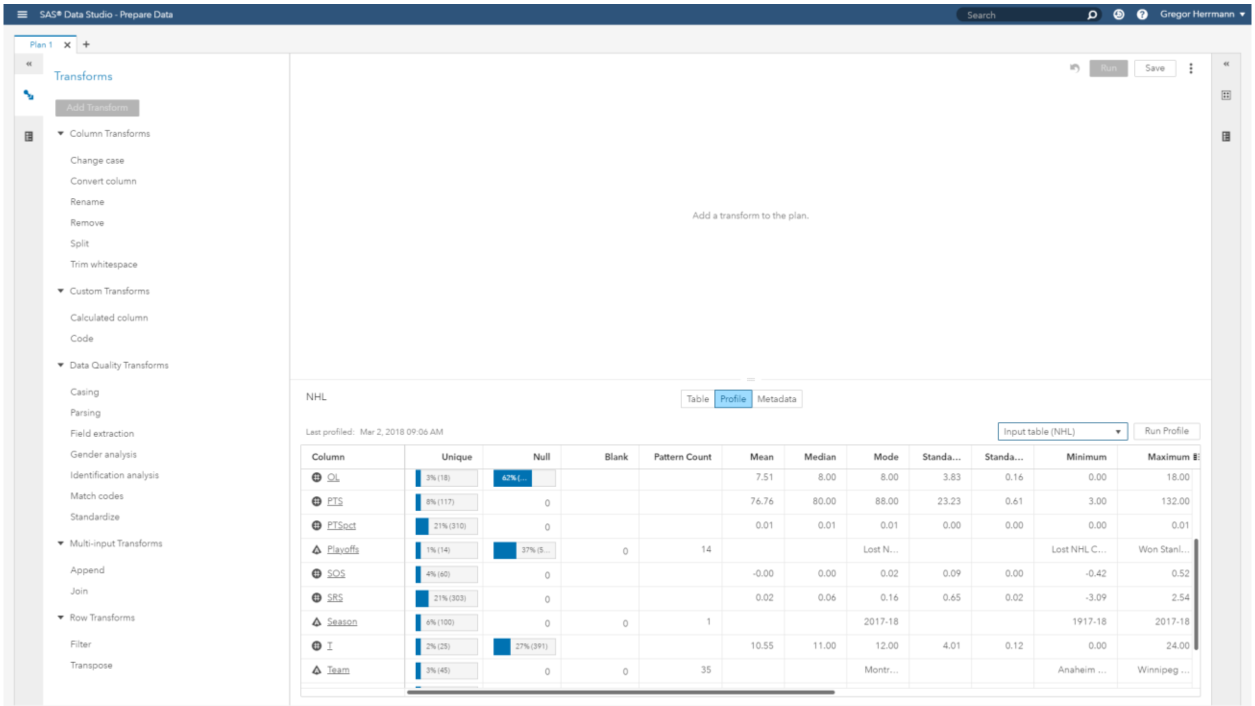
Date modified:
Feb 11, 2018 05:15 PM

Figure 5. Open Data Source

DATA PREPARATION IN SAS VISUAL ANALYTICS 8.2

BASIC CONCEPTS

The SAS® Data Studio interface enables you to prepare and view data. Data preparation tasks in SAS Visual Analytics are stored as data plans. A plan is a collection of data transforms or actions performed on a table. SAS® Data Studio provides a convenient way for you to prepare data in tables, to keep track of the changes that you make to tables, and to modify or view the history of actions that you made to tables. If you start with a new plan, your first action is always adding a table to a plan. If you are not familiar with the content of your table, you should run a profile on your table. The profile gives you a good overview and might contain some hints if you are facing data quality issues (for example, variables that contain a lot of missing values).



Last profiled: Mar 2, 2018 09:06 AM

Column	Unique	Null	Blank	Pattern Count	Mean	Median	Mode	Standa...	Standa...	Minimum	Maximum
OL	3% (18)	62% (...			7.51	8.00	8.00	3.83	0.16	0.00	18.00
PTS	8% (117)	0			76.76	80.00	88.00	23.23	0.61	3.00	132.00
PTSpct	21% (310)	0			0.01	0.01	0.01	0.00	0.00	0.00	0.01
Playoffs	1% (14)	37% (5...	0	14			Lost N...			Lost NHL C...	Won Stanl...
SOS	4% (60)	0			-0.00	0.00	0.02	0.09	0.00	-0.42	0.52
SRS	21% (303)	0			0.02	0.06	0.16	0.65	0.02	-3.09	2.54
Season	6% (100)	0	0	1			2017-18			1917-18	2017-18
T	2% (25)	27% (391)			10.55	11.00	12.00	4.01	0.12	0.00	24.00
Team	3% (45)	0	0	35			Montr...			Anaheim ...	Winnipeg ...

Figure 5. Viewing a profile in SAS Data Studio

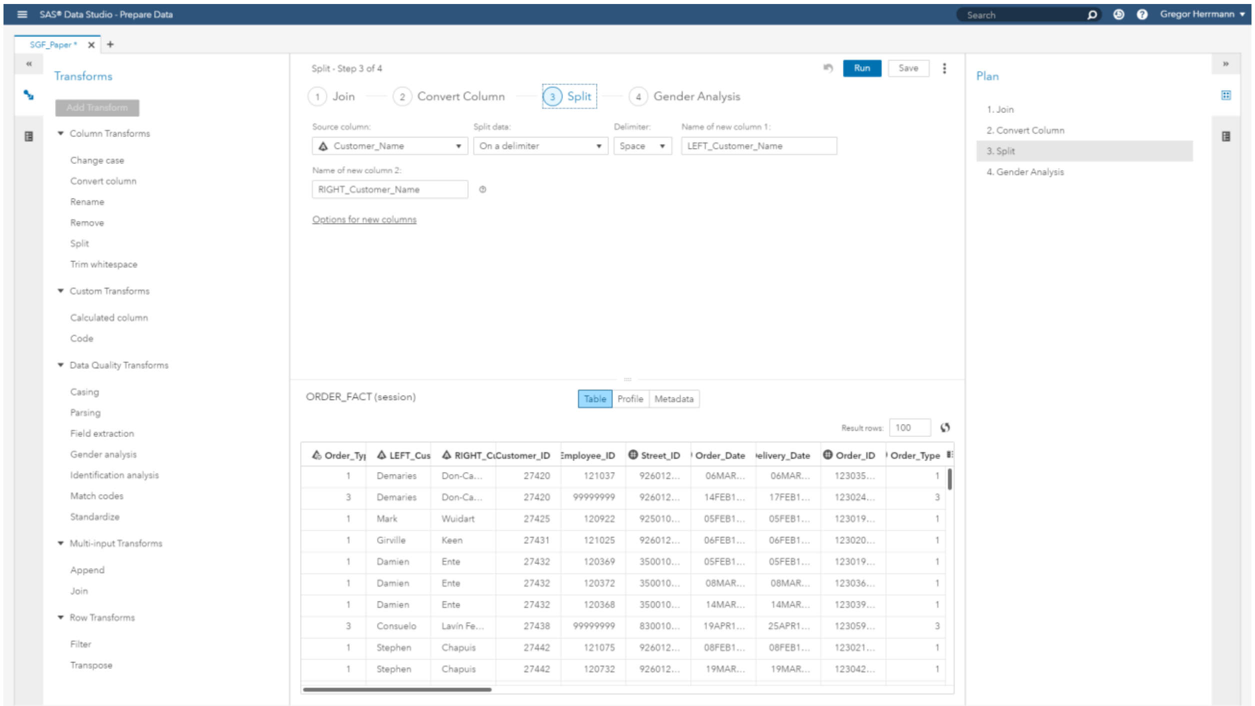
DATA TRANSFORMS

You can choose from a large variety of data transforms to perform desired operations on your existing data. The transforms range from simple transforms like Rename, Change case, or Trim whitespace to more complex actions like Join or Transpose. They cover the most frequently used data transformations in terms of self-service data preparation. A sequence of data transforms can be saved as a data plan. Every transform requires initial modifications to be able to run the action. After all transforms have been executed successfully, you can save your plan.

If SAS® Data Preparation is licensed at your site, you can access the data quality transforms and integrate one or more of them into your data plan.

The data quality transforms use SAS® Quality Knowledge Base (QKB), which is a collection of locales and other information that is referenced during data analysis and data cleansing. The data quality transforms apply a QKB locale and a definition to a selected source column. Definitions define data formats for specific types of content and data cleansing. For example, a parse definition for a street address describes how a street address can be parsed into identifiable segments.

A locale reflects the language and linguistic conventions of a geographic region. These conventions can include word order or language selection for the country or region.



Split - Step 3 of 4

1 Join — 2 Convert Column — 3 Split — 4 Gender Analysis

Source column: Customer_Name
Split data: On a delimiter
Delimiter: Space
Name of new column 1: LEFT_Customer_Name
Name of new column 2: RIGHT_Customer_Name
Options for new columns

Plan: 1. Join; 2. Convert Column; 3. Split; 4. Gender Analysis

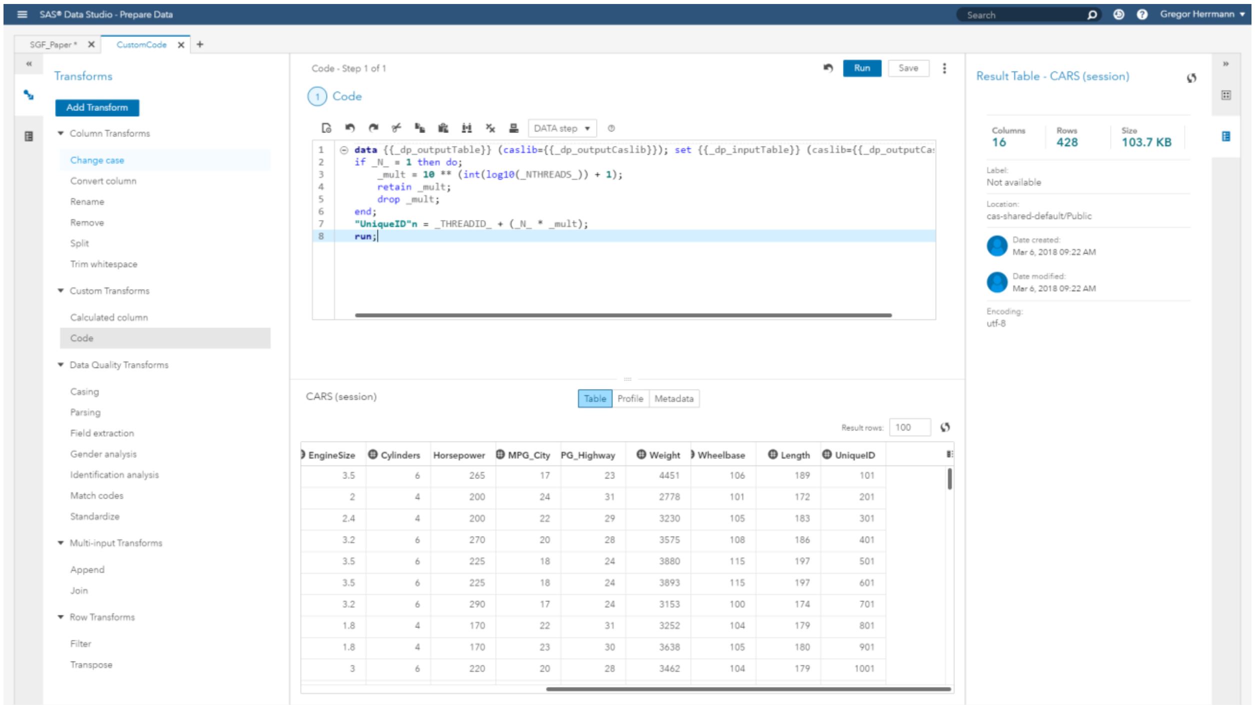
ORDER_FACT (session)

Result rows: 100

Order_Typ	LEFT_Cus	RIGHT_C	Customer_ID	Employee_ID	Street_ID	Order_Date	elivery_Date	Order_ID	Order_Type
1	Demaries	Don-Ca...	27420	121037	926012...	06MAR...	06MAR...	123035...	1
3	Demaries	Don-Ca...	27420	99999999	926012...	14FEB1...	17FEB1...	123024...	3
1	Mark	Wuidart	27425	120922	925010...	05FEB1...	05FEB1...	123019...	1
1	Girville	Keen	27431	121025	926012...	06FEB1...	06FEB1...	123020...	1
1	Damien	Ente	27432	120369	350010...	05FEB1...	05FEB1...	123019...	1
1	Damien	Ente	27432	120372	350010...	08MAR...	08MAR...	123036...	1
1	Damien	Ente	27432	120368	350010...	14MAR...	14MAR...	123039...	1
3	Consuelo	Lavín Fe...	27438	99999999	830010...	19APR1...	25APR1...	123059...	3
1	Stephen	Chapuis	27442	121075	926012...	08FEB1...	08FEB1...	123021...	1
1	Stephen	Chapuis	27442	120732	926012...	19MAR...	19MAR...	123042...	1

Figure 5. Data Plan

If you want to accomplish more complex data transformations, you can insert a code transform into your data plan. You can choose from two available code languages: CASL and DATA step. Each time you run a plan, table and library names might change. To avoid errors, you must use variables instead of table names and CAS library names in your code. Using variables instead of table names and CAS library names eliminates the possibility that the code will fail due to name changes. You can see the variables in the first line of the **CustomCode** transform in the following screenshot. After executing your CustomCode transform, you can download the log to check correctness.



Code - Step 1 of 1

1 Code

DATA step

```
data {{_dp_outputTable}} (caslib={{_dp_outputCaslib}}); set {{_dp_inputTable}} (caslib={{_dp_outputCa
if _N_ = 1 then do;
    _mult = 10 ** (int(log10(_NTHREADS_)) + 1);
    retain _mult;
    drop _mult;
end;
"UniqueID"n = _THREADID_ + (_N_ * _mult);
run;
```

Result Table - CARS (session)

Columns: 16
Rows: 428
Size: 103.7 KB
Label: Not available
Location: cas-shared-default/Public
Date created: Mar 6, 2018 09:22 AM
Date modified: Mar 6, 2018 09:22 AM
Encoding: utf-8

CARS (session)

Result rows: 100

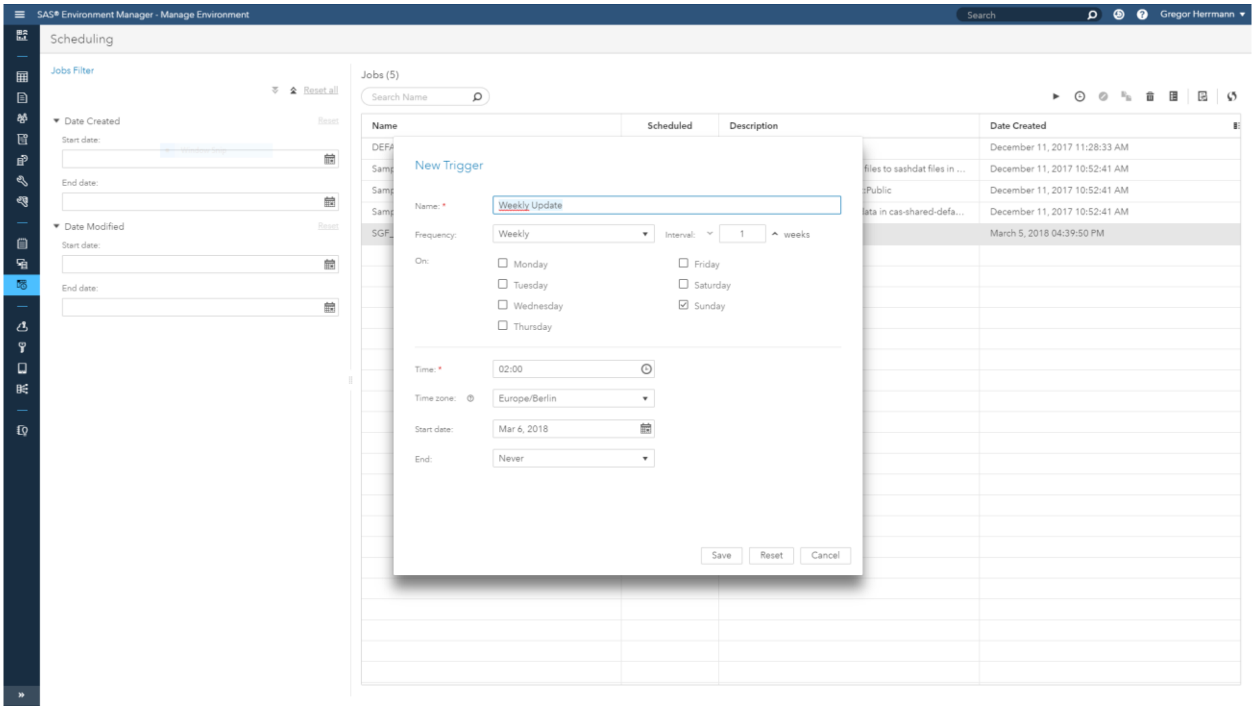
EngineSize	Cylinders	Horsepower	MPG_City	PG_Highway	Weight	Wheelbase	Length	UniqueID
3.5	6	265	17	23	4451	106	189	101
2	4	200	24	31	2778	101	172	201
2.4	4	200	22	29	3230	105	183	301
3.2	6	270	20	28	3575	108	186	401
3.5	6	225	18	24	3880	115	197	501
3.5	6	225	18	24	3893	115	197	601
3.2	6	290	17	24	3153	100	174	701
1.8	4	170	22	31	3252	104	179	801
1.8	4	170	23	30	3638	105	180	901
3	6	220	20	28	3462	104	179	1001

Figure 6. CustomCode Transform

ADDITIONAL CAPABILITIES

From the toolbar, by clicking on the symbol with three dots, a drop-down menu is displayed, providing additional capabilities. You can download both the code and the log of the current data plan, you can change the source table of the data plan, or you can immediately start building a report or developing a model with either your source or your target table.

If you choose **Save As** from the drop-down menu, you can specify the location and the name of your target table and whether the table should be replaced if it already exists. To run a data plan on a regular basis in batch, you must create a job. After giving your job a name and saving it, the job can be scheduled to run in the background from SAS Environment Manager. In the current version, only time-based triggers are available.



SAS® Environment Manager - Manage Environment

Scheduling

Jobs Filter

Date Created
Start date:
End date:

Date Modified
Start date:
End date:

Jobs (5)

Name	Scheduled	Description	Date Created
DEFA			December 11, 2017 11:28:33 AM
Samp		files to sashdat files in ...	December 11, 2017 10:52:41 AM
Samp		:Public	December 11, 2017 10:52:41 AM
Samp		lata in cas-shared-defa...	December 11, 2017 10:52:41 AM
SGF_			March 5, 2018 04:39:50 PM

New Trigger

Name: * Weekly Update

Frequency: Weekly Interval: 1 weeks

On:
□ Monday
□ Tuesday
□ Wednesday
□ Thursday
□ Friday
□ Saturday
☑ Sunday

Time: * 02:00

Time zone: Europe/Berlin

Start date: Mar 6, 2018

End: Never

Figure 6. Scheduling a job

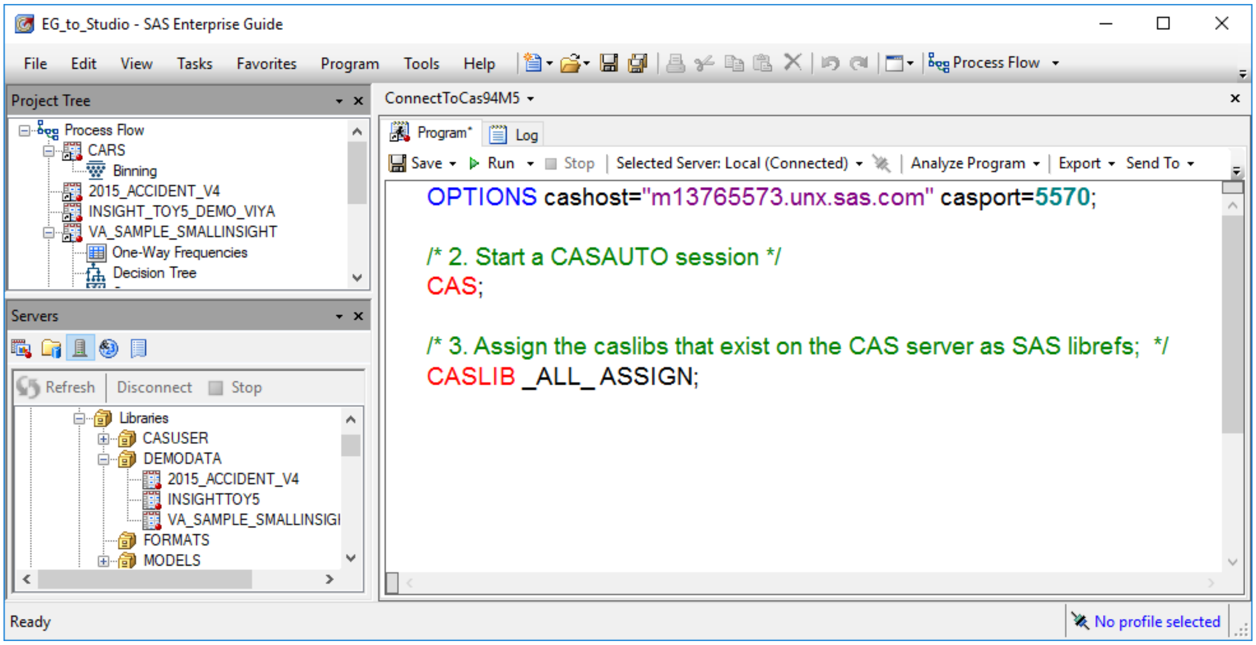
USING SAS 9 PROGRAMS AND TOOLS

You had a closer look at how data preparation works in SAS Visual Analytics on SAS Viya. Many of you, however, might have large amounts of existing SAS programs that do data preparation for you. What if you want to use these valuable assets?

Up to SAS 9.4M4, there was a way to execute SAS programs in a SAS Viya environment using SAS/CONNECT. It required SAS/CONNECT in both the SAS 9 and the SAS Viya environment, and it was not very easy to use. Beginning with SAS 9.4M5, executing programs in CAS from an existing SAS session got much easier. Let's dive deeper into it. The following screenshots show SAS® Enterprise Guide® as the interface to execute the SAS programs. The same code examples can be used from SAS® Studio or the Display Manager.

Connecting to CAS

To be able to execute any code on the CAS server, you have to make a connection first, which requires an identity that can authenticate against the operating system of your CAS server. In this example, connection is made from a Windows 10 laptop to a CAS server running on Linux using an .authinfo file that contains the credential information. It is basically a text file with a userid and encrypted password. A similar mechanism is available for other operating systems as well. The three lines of code in the screenshot establish a connection to a CAS server, start a CAS session, and make all caslibs available in the SAS session in SAS Enterprise Guide.

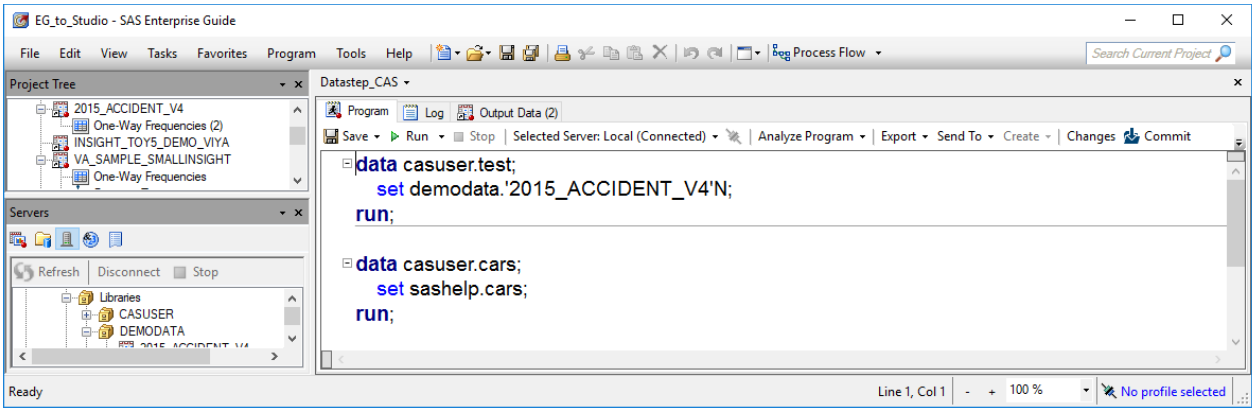


```
OPTIONS cashost="m13765573.unx.sas.com" casport=5570;

/* 2. Start a CASAUTO session */
CAS;

/* 3. Assign the caslibs that exist on the CAS server as SAS librefs;  */
CASLIB _ALL_ ASSIGN;
```

Figure 7. Connecting to CAS

Executing SAS Programs in CAS

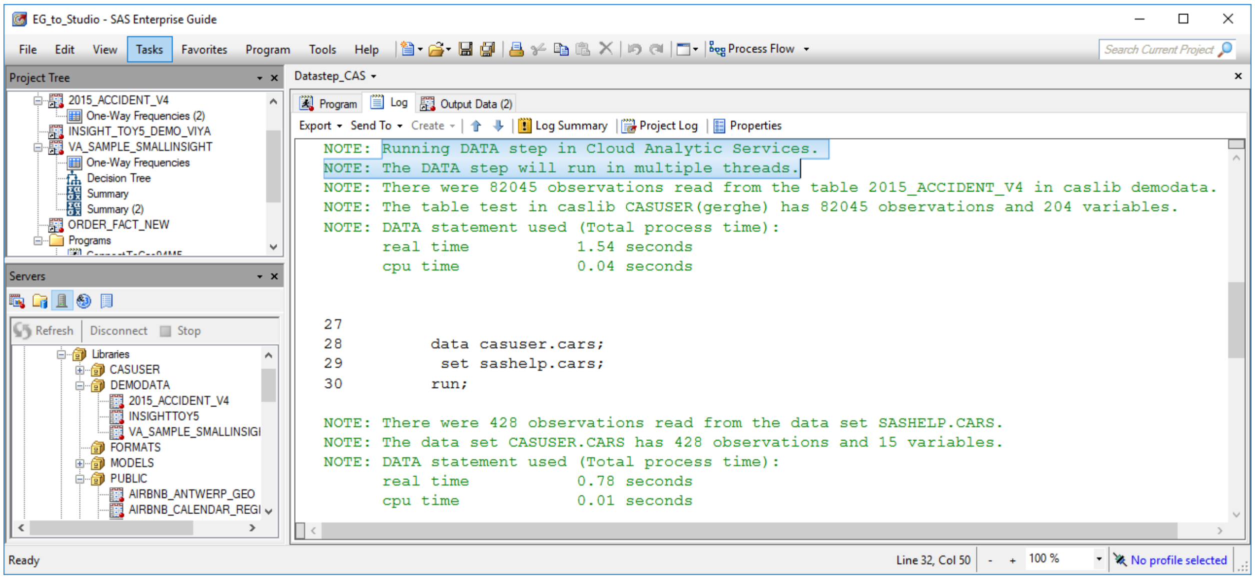
A big advantage of being able to execute SAS programs in CAS is the fact that the DATA step and a large number of SAS procedures can be executed in CAS on a distributed system. This might lower execution times of your SAS programs significantly, especially if your data sizes are large. To make sure your programs execute in CAS, the procedure that you are using must be enabled in CAS and input tables and output tables must reside in a caslib. In the example below, you see two very simple DATA steps. In the first DATA step, both the input table and output table reside in a caslib. In the second DATA step, the input is coming from a SAS library.



```
data casuser.test;
   set demodata.'2015_ACCIDENT_V4'N;
run;

data casuser.cars;
   set sashelp.cars;
run;
```

Figure 8. Executing a DATA step in CAS

If you look at the log, you can clearly see the difference: the first DATA step executes in CAS, whereas the second DATA step runs in a SAS session.



```
NOTE: Running DATA step in Cloud Analytic Services.
NOTE: The DATA step will run in multiple threads.
NOTE: There were 82045 observations read from the table 2015_ACCIDENT_V4 in caslib demodata.
NOTE: The table test in caslib CASUSER(gerghe) has 82045 observations and 204 variables.
NOTE: DATA statement used (Total process time):
      real time           1.54 seconds
      cpu time            0.04 seconds


27
28          data casuser.cars;
29           set sashelp.cars;
30          run;

NOTE: There were 428 observations read from the data set SASHELP.CARS.
NOTE: The data set CASUSER.CARS has 428 observations and 15 variables.
NOTE: DATA statement used (Total process time):
      real time           0.78 seconds
      cpu time            0.01 seconds
```

Figure 9. Examining the Log

As a best practice, upload your data to CAS first if you intend to execute DATA steps or procedures on large files. In terms of data preparation, be aware that PROC SQL is not enabled in CAS. If you intend to run SQL statements on the CAS server, you must use PROC FedSQL instead.

CONCLUSION

The need for tools that enable self-service data preparation capabilities will grow. SAS Visual Analytics on SAS Viya provides an easy-to-use interface for report authors and data scientists to access and prepare data without intervention from IT. The seamless integration of existing SAS programs enables existing customers to move to SAS Visual Analytics on SAS Viya and to benefit from new capabilities for analysis and visualization in a unified HTML5 interface.

RECOMMENDED READING

- Hazejager, Wilbram. 2018. “Data Management in SAS Viya.” *Proceedings of the SAS Global Forum 2018 Conference*. Cary, NC. SAS Institute Inc.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Gregor Herrmann
SAS Institute Inc.
SAS Campus Drive
Cary, NC, 27513
gregor.herrmann@sas.com

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

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