Paper SAS3195-2019

# Months to Weeks: SAS® Credit Scoring Express Implementation

Abhijit S. Joshi, SAS Institute Inc., Pune, India

## ABSTRACT

SAS® Credit Scoring is a flagship product from SAS that helps you reduce credit losses and boost your overall business performance by making better, data-driven credit decisions on both the origination and servicing sides of your business. One of the key benefits of the solution is the reduction in time to market, which delivers significant Return on Investment (ROI). Recently, the solution has undergone significant enhancements that reduce its implementation time from several months to several weeks—enabling financial institutions to realize their time-to-market ROI even sooner. Based on real-life project experiences, this paper outlines how implementation can be fast-tracked through an improved assessment process and a phased implementation approach. Moreover, the use of an automation toolkit optimizes implementation efforts, which helps further reduce time frames. This combination of processes and a toolkit enables you to leverage the existing data infrastructure at a site, thereby avoiding long project cycles that are associated with building data warehouses or data marts that feed into the solution. This paper also provides existing and new customers of SAS Credit Scoring with an update on recent enhancements to the solution and a roadmap to faster ROI.

## INTRODUCTION

Credit decisions are getting more and more complex by the day. When it comes to profit making for money lending organizations, credit decisions can be tricky. Being too conservative or too aggressive can result in lost business opportunity or increased losses. Balancing both risk and opportunity requires analytics and intelligence, but also the proper infrastructure to operationalize decisions efficiently. SAS® Credit Scoring provides such an infrastructure – it is a comprehensive solution for development, maintenance, and monitoring of credit risk models and scorecards.

In a nutshell, for resolving business problem, statistical models are built. These models use analytical base table, created by using data from different sources. Now each of these data sources must be configured in the SAS Credit Scoring solution before the process of model building begin. In previous implementation approach, configuring the solution was the major pain area and used to take significant amount of time in overall process.

SAS Credit Scoring has following workspaces:

SAS® Analytical Data Set Builder (a component of SAS Credit Scoring Solution):

- Data Preparation: In this workspace, you build analytical base tables (ABTs) by creating or importing variables, which are used by modelers to build credit risk models. A centralized GUI-based repository is created for maintaining different entities (filters, variable definitions, and bins).
- Configuration: In this workspace, you configure solution components such as Purpose, Subject of Analysis, Data Sources, and so on, to make the solution ready to use.

SAS Credit Scoring

- Model Specification: This workspace enables you to import models, create judgmental scorecards, and manage model details such as expected values.
- Backtesting: Models can be backtested and validated to ensure that they will perform as expected on actual out-of-time data.

- Model Deployment: In this workspace, you can create and manage scoring templates.
- Model Monitoring: Here you can view model-monitoring reports that measure the stability, performance, and calibration of all predictive models.

This paper mainly addresses the Configuration workspace in SAS Analytical Data Set Builder and presents a method for expediting the implementation timelines.

A full implementation of SAS Credit Scoring was required in previous versions of the solution and included the following high-level steps:

1. Gathering requirements and assessing the solution
2. Installing SAS Credit Scoring at customer's machine
3. Mapping the customer data warehouse to the SAS® Detail Data Store
4. Writing an extract, transform, load (ETL) program to populate the detail data store (DDS)
5. Modifying the DDS to accommodate customer variables if required
6. Updating the DDS to foundation mart ETL program, if required
7. Configuring SAS Credit Scoring

When the above approach is followed, implementation timelines are elongated, and in some cases they take several months. SAS recognized that not all customers required the full implementation and thus re-designed the solution to enable faster timelines for those customers. Other challenges faced during implementation are discussed in the next section

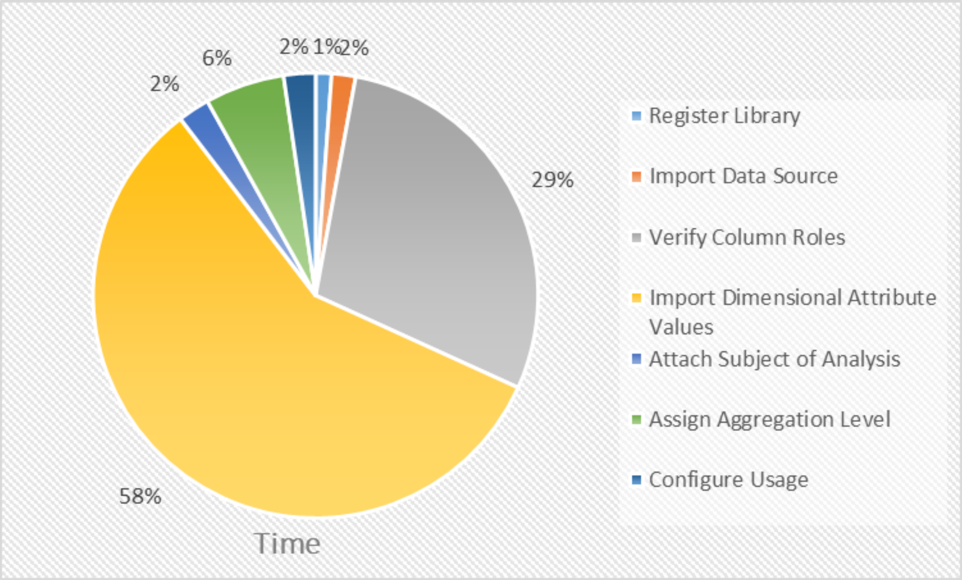
## IMPLEMENTATION CHALLENGES

The following were some of the issues that in some cases caused longer implementation cycles:

- substantial customizations needed in the DDS to support customer requirements
- significant time spent in populating the foundation mart based on this customized DDS
- configuring external data sources directly was not possible or was time consuming and required several manual steps

To deal with these challenges, SAS first enabled external data sources to be directly configurable, without the need for the DDS. For customers who did not require the DDS, this reduced the implementation time significantly. In addition, we created a utility to accelerate the configuration of the external data sources and adopted an improved implementation methodology to further reduce time lines.

Figure 1. Typical Implementation Time DistributionFigure 1 shows a typical distribution of time spent on data management tasks during implementation. To reduce timelines, it was clear that these tasks would need to be shortened.



| | |
|---|---|
| Library - 1 | Dimensions - 5 in each dataset |
| Data Sets - 10 | Dim. Values - 10 each |
| Columns - 50 each | |

Figure 1. Typical Implementation Time Distribution

## EXPRESS IMPLEMENATION APPROACH

Based on our experiences deploying the solution on many sites, we re-designed the implementation approach to enable faster completion. The following sections describe each step in detail.

1. Systematic assessment of the solution
2. Understanding customer requirements
3. Conducting an information workshop
4. Providing an implementation plan
5. Providing implementation support
6. Using the automation toolkit
7. Using the accelerator file
8. Using the usage file
9. Using SAS programs
10. Creating a default point-in-time ABT

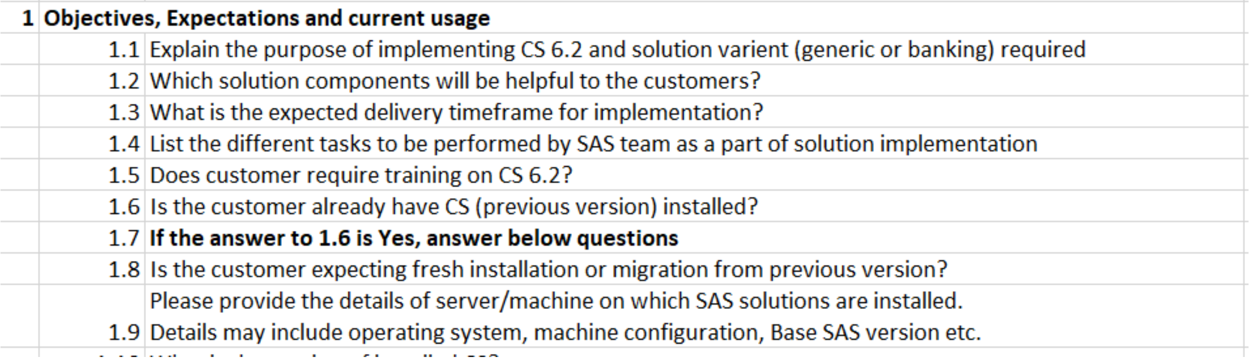
### SYSTEMATIC SOLUTION ASSESSMENT

Solution assessment is the step where we gather relevant information from the customer. We typically cover the following information:

- Challenges faced in previous versions
- Capacity at which the customer is using the solution

- Number and size of analytical base tables (ABTs) available and required
- Number of models
- Number and complexity of input data sources
- Number of concurrent users
- Most useful or frequently used solution components and so on

To expedite the overall assessment process, we send a questionnaire to the customer before any consultant visits. Once all the information is gathered, the Consulting team then starts preparing for the detailed requirement-gathering session. For example, Figure 2 shows the questions targeted toward understanding the customer's basic requirements.



| | | |
|---|---|---|
| **1** | **Objectives, Expectations and current usage** | |
| | 1.1 | Explain the purpose of implementing CS 6.2 and solution varient (generic or banking) required |
| | 1.2 | Which solution components will be helpful to the customers? |
| | 1.3 | What is the expected delivery timeframe for implementation? |
| | 1.4 | List the different tasks to be performed by SAS team as a part of solution implementation |
| | 1.5 | Does customer require training on CS 6.2? |
| | 1.6 | Is the customer already have CS (previous version) installed? |
| | **1.7** | **If the answer to 1.6 is Yes, answer below questions** |
| | 1.8 | Is the customer expecting fresh installation or migration from previous version? |
| | 1.9 | Please provide the details of server/machine on which SAS solutions are installed.<br>Details may include operating system, machine configuration, Base SAS version etc. |

Figure 2. Sample Assessment Questionnaire

## UNDERSTANDING CUSTOMER REQUIREMENTS

Requirement gathering is completed through multiple sessions depending on the complexity of the requirements and data. One useful exercise is to discuss the questionnaire with customers in order to gauge priority and to understand the reasoning behind their answers.

In the requirement-gathering session, we specifically target the following items for collecting information:

- data sources
- tables
- columns
- ABT if available
- source-to-target mapping
- models and their significant variables
- types of scorecards
- frequency of scoring, and so on

A clear understanding of the customer's existing system results in better planning capability.

## CONDUCTING AN INFORMATION WORKSHOP

Sometimes customers want to assume responsibility for implementing the solution. In such cases, SAS conducts a workshop to enable its customers. We have found that, using customer-provided sample data in the workshop helps them understand each step much better.

## PROVIDING AN IMPLEMENTATION PLAN

Once the requirements are clear, consultants draft the implementation plan. Our strategy is to convert the requirements into a phased delivery approach. There might be multiple phases depending on the requirements. The following is an example scenario:

1. Phase 1: If the customer already has the base table that they are using for model development, we suggest taking that base table as is into the SAS Credit Scoring environment and creating ABTs with point-in-time variables to shorten timelines.
2. Phase 2: First, consider all the high-priority behavioral scorecard models that have overlapping variables. Then, build the ABT with variables that cover all the models.
3. Phase 3: Do the same exercise for application scorecard models. The time required to complete this phase is dependent on the number and complexity of the required variables. It typically lasts from several days to weeks.

The implementation plan is then discussed with the customer and finalized.

## PROVIDING IMPLEMENTATION SUPPORT

In cases where customers are going to implement the solution themselves, SAS can provide full implementation support. To minimize issues, it is important that customers follow the implementation plan. We also provide best practice guidelines, issue resolution, required training, and workshops.

## USING THE AUTOMATION TOOLKIT

Manually configuring data sources can take a lot of time. To expedite the process, we have introduced an automation toolkit, which can perform all the required steps for configuring external data sources with minimal manual intervention. It can also create the ABT with direct variables (this corresponds to Phase 1 in "Providing an Implementation Plan"). SAS codes and REST APIs are used to create the automation toolkit.

Figure 3 illustrates the steps that are required for configuring an external data source. These steps are described in detail in the discussion of the automation approach.

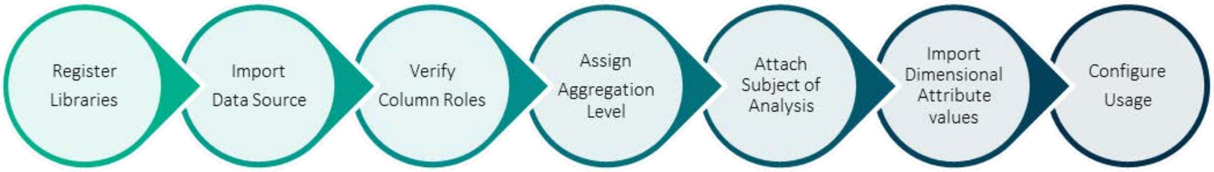


Figure 3. External Data Source Configuration Steps

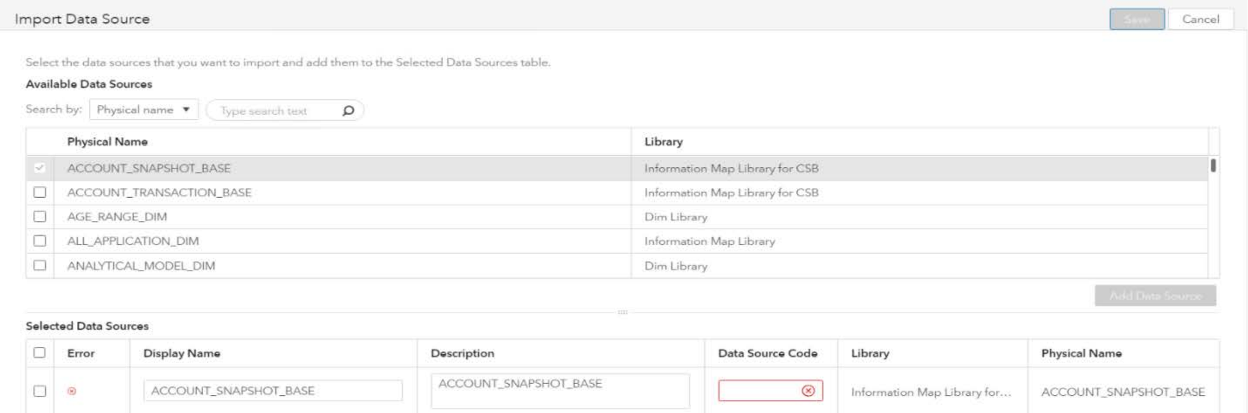
1. Register Libraries

   The first step in configuring an external data source is to configure a library. Previously, in order to configure a library, you had to manually insert a new record in the APDM.LIBRARY_MASTER table, which is now automated using the automation toolkit.

2. Import Data Source

   After the library is configured, all the data sources underneath it, which are not yet configured, are available for configuration. In the approach previously in use, a three-character unique code had to be provided manually while importing a new data source, making it difficult to keep track of all the used codes. This problem is resolved using the automation toolkit.

The automation toolkit accepts the input in a pre-defined format, in which a user can provide details about data sources that must be configured. When the required SAS code is executed, all the mentioned data sources get imported. The code takes care of unique codes for all the data sources. Display 1 shows the screen for importing a new data source manually.



Select the data sources that you want to import and add them to the Selected Data Sources table.

Available Data Sources

Search by: Physical name ▾ Type search text

|  | Physical Name | Library |
|---|---|---|
| ☑ | ACCOUNT_SNAPSHOT_BASE | Information Map Library for CSB |
| □ | ACCOUNT_TRANSACTION_BASE | Information Map Library for CSB |
| □ | AGE_RANGE_DIM | Dim Library |
| □ | ALL_APPLICATION_DIM | Information Map Library |
| □ | ANALYTICAL_MODEL_DIM | Dim Library |

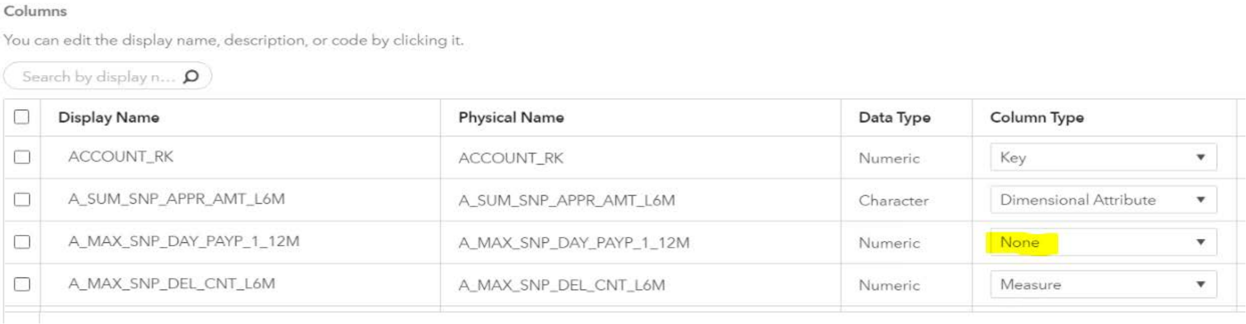
Selected Data Sources

| □ | Error | Display Name | Description | Data Source Code | Library | Physical Name |
|---|---|---|---|---|---|---|
| □ | ⊗ | ACCOUNT_SNAPSHOT_BASE | ACCOUNT_SNAPSHOT_BASE | ⊗ | Information Map Library for… | ACCOUNT_SNAPSHOT_BASE |

Display 1. Import Data Source

3. Verify Column Roles

After importing the data source, each of its columns is assigned a role (column type). For example, if the ACCOUNT_RK column is assigned a KEY role, then it is used as a key column, and similarly, if BALANCE is assigned a MEASURE role, then it is used as a fact in calculations. The column role is assigned as NONE if the variable naming convention is not according to the SAS class codes. The automation toolkit assigns the correct role if a variable role is NONE. The SAS code checks the data type of that variable and assigns it as MEASURE for numeric and DIMENSIONAL ATTRIBUTE for character. In previous implementations, a user had to manually assign the correct role. Display 2 shows an example of a column that is assigned the NONE role.



Columns

You can edit the display name, description, or code by clicking it.

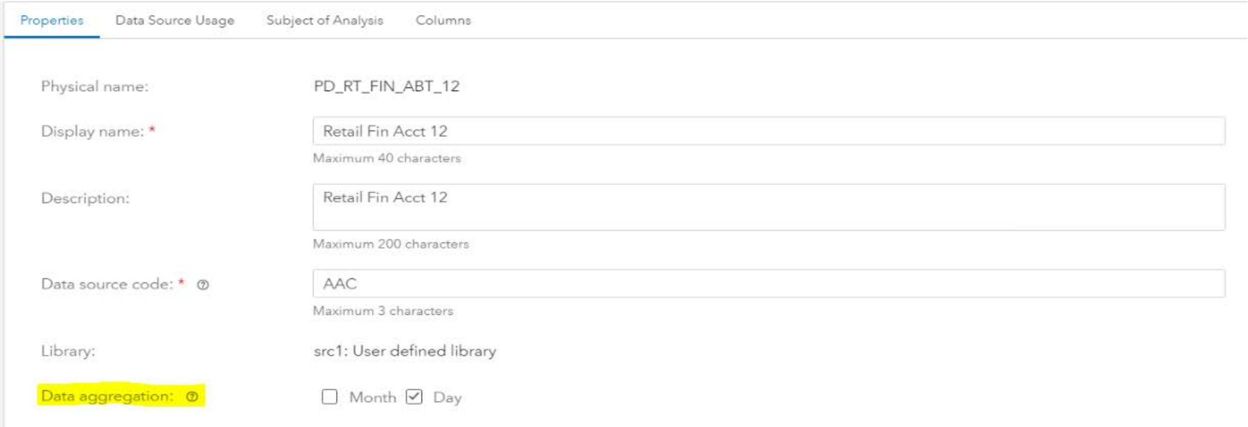
Search by display n…

| □ | Display Name | Physical Name | Data Type | Column Type |
|---|---|---|---|---|
| □ | ACCOUNT_RK | ACCOUNT_RK | Numeric | Key ▾ |
| □ | A_SUM_SNP_APPR_AMT_L6M | A_SUM_SNP_APPR_AMT_L6M | Character | Dimensional Attribute ▾ |
| □ | A_MAX_SNP_DAY_PAYP_1_12M | A_MAX_SNP_DAY_PAYP_1_12M | Numeric | None ▾ |
| □ | A_MAX_SNP_DEL_CNT_L6M | A_MAX_SNP_DEL_CNT_L6M | Numeric | Measure ▾ |

Display 2. Verify Column Roles

4. Assign Aggregation Level

To identify whether the data source supports daily, monthly, or both roll-up methods, a data aggregation approach must be configured for that data source. Previously, a user had to manually configure this step, but now with the automation toolkit, it can be done automatically. Display 3 shows the window for configuring the data aggregation level.



Physical name: PD_RT_FIN_ABT_12

Display name: * Retail Fin Acct 12
Maximum 40 characters

Description: Retail Fin Acct 12
Maximum 200 characters

Data source code: * AAC
Maximum 3 characters

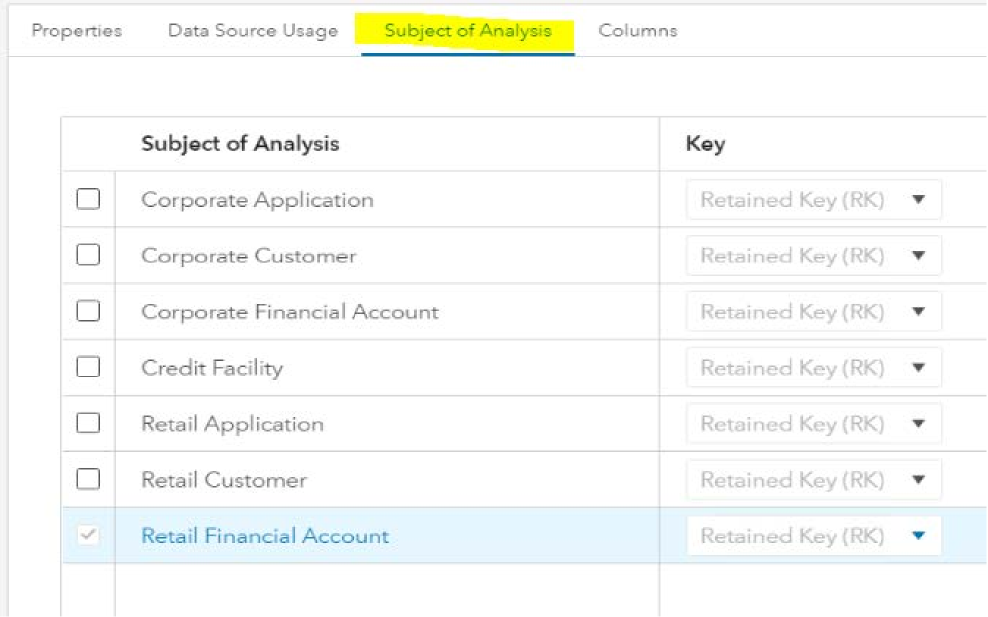
Library: src1: User defined library

Data aggregation: ☐ Month ☑ Day

Display 3. Assign Data Aggregation Level

5. Attach Subject of Analysis

   The subject of analysis (SOA) provides the information about the granularity of the subject. For example, data can be imported at the account level or at the customer level. It must be assigned to the data source for making it available for variable creation. Previously, this was also a manual step, but now the automation toolkit can assign the primary SOA by using inputs from the accelerator file. Display 4 shows the window where you configure the SOA.



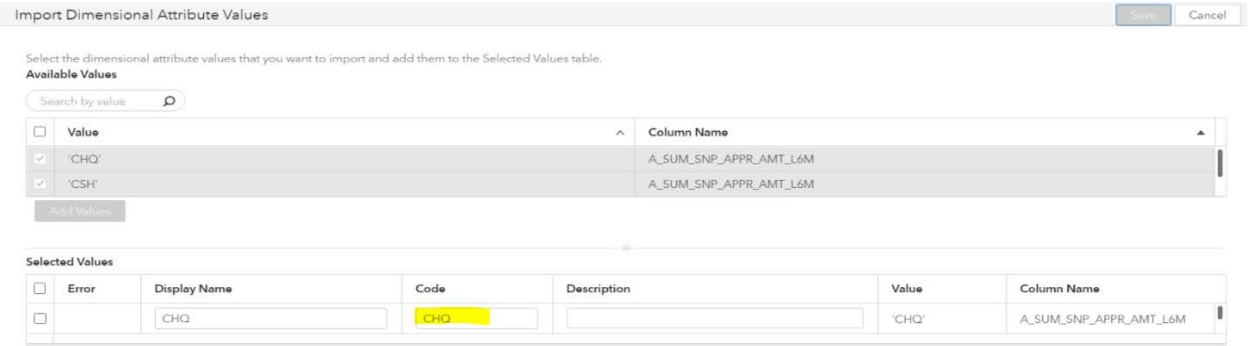
| | Subject of Analysis | Key |
|---|---|---|
| ☐ | Corporate Application | Retained Key (RK) |
| ☐ | Corporate Customer | Retained Key (RK) |
| ☐ | Corporate Financial Account | Retained Key (RK) |
| ☐ | Credit Facility | Retained Key (RK) |
| ☐ | Retail Application | Retained Key (RK) |
| ☐ | Retail Customer | Retained Key (RK) |
| ☑ | Retail Financial Account | Retained Key (RK) |

Display 4. Attach Subject of Analysis

6. Import Dimensional Attribute Values

   Data source variables that are of type DIMENSIONAL ATTRIBUTE require a list of unique values. For example, the dimensional attribute variable TRANSACTION TYPE can have a value of CASH, CHEQUE, and so on as its values. These values are not available when the variable is created in SAS Analytical Data Set Builder; they are not available until the user imports them.

   The automation toolkit accepts the input from a user that defines which of all the dimensional attribute values get imported for a specified variable. Previously, importing these values was a manual task. Display 5 shows the window in which you import dimensional attribute values.



Select the dimensional attribute values that you want to import and add them to the Selected Values table.

Available Values

| □ | Value | Column Name |
|---|---|---|
| ☑ | 'CHQ' | A_SUM_SNP_APPR_AMT_L6M |
| ☑ | 'CSH' | A_SUM_SNP_APPR_AMT_L6M |

Selected Values

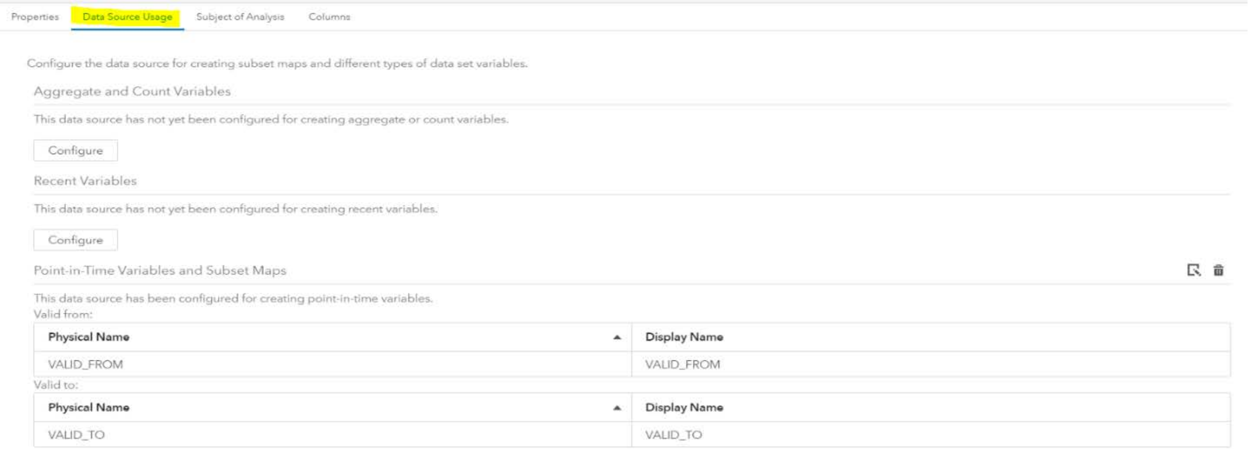
| □ | Error | Display Name | Code | Description | Value | Column Name |
|---|---|---|---|---|---|---|
| □ |  | CHQ | CHQ |  | 'CHQ' | A_SUM_SNP_APPR_AMT_L6M |

Display 5. Import Dimensional Attribute Values

7. Configure Usage

   You can create variables from only those data sources whose USAGE property is configured. Thus, it is very important to correctly configure the usage.

   The automation toolkit helps the user to configure the usage of the data sources in a short time. In Display 6, which shows the window for configuring data source usage, the data source is configured for creating POINT-IN-TIME type of variables. Previously, this was a manual activity.



Configure the data source for creating subset maps and different types of data set variables.

Aggregate and Count Variables

This data source has not yet been configured for creating aggregate or count variables.

Recent Variables

This data source has not yet been configured for creating recent variables.

Point-in-Time Variables and Subset Maps

This data source has been configured for creating point-in-time variables.

Valid from:

| Physical Name | Display Name |
|---|---|
| VALID_FROM | VALID_FROM |

Valid to:

| Physical Name | Display Name |
|---|---|
| VALID_TO | VALID_TO |

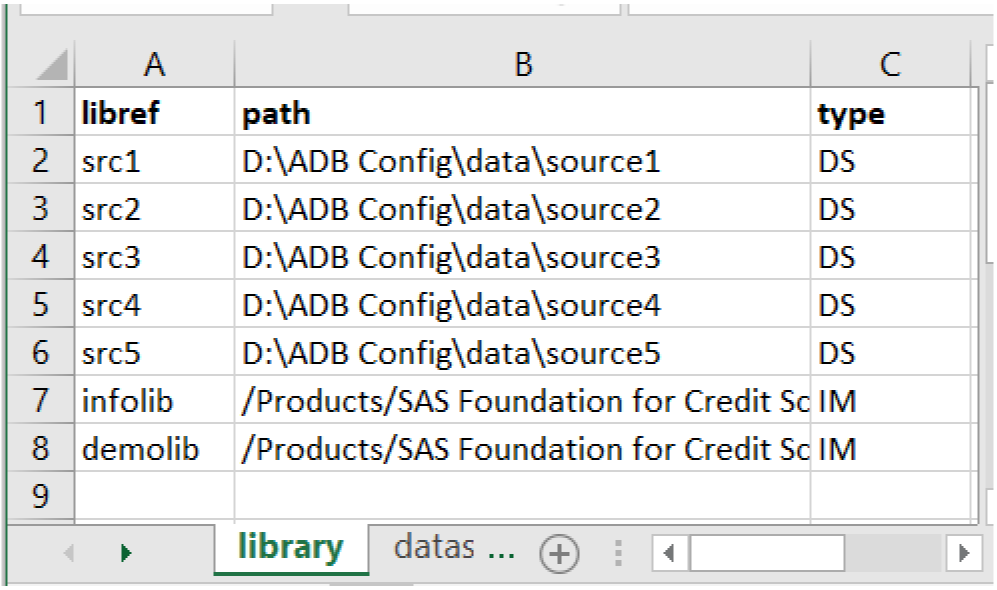
Display 6. Configure Usage

## USING THE ACCELERATOR FILE

The backbone of this automation toolkit is the accelerator file (input.xlsx), which takes input from the user in a pre-defined format. This file is used in the SAS programs that use REST APIs to communicate with SAS Credit Scoring and perform the required action.

The accelerator file has four sheets:

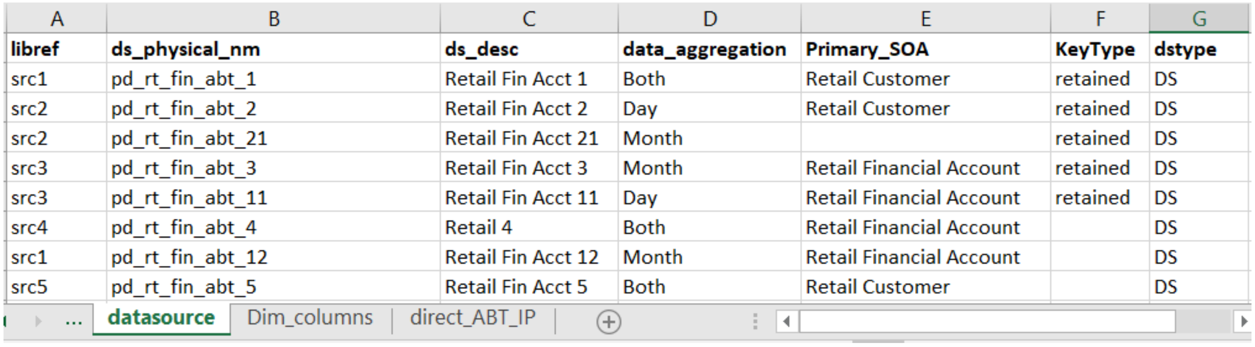
- library: This sheet accepts input for configuring the libraries. Required columns are libref, path, and type. The path must be the absolute path of the library. The library could be a Base SAS library for SAS data sets and views (type = DS), or it could be an information map library (type = IM). Display 7 shows sample data populated in the library sheet.



| libref | path | type |
|---|---|---|
| src1 | D:\ADB Config\data\source1 | DS |
| src2 | D:\ADB Config\data\source2 | DS |
| src3 | D:\ADB Config\data\source3 | DS |
| src4 | D:\ADB Config\data\source4 | DS |
| src5 | D:\ADB Config\data\source5 | DS |
| infolib | /Products/SAS Foundation for Credit Sc | IM |
| demolib | /Products/SAS Foundation for Credit Sc | IM |

Display 7. Accelerator File: library

- datasource: This sheet accepts input for configuring the data source and the data aggregation level, as well as for attaching the subject of analysis. Display 8 shows the sample data populated in the datasource sheet.

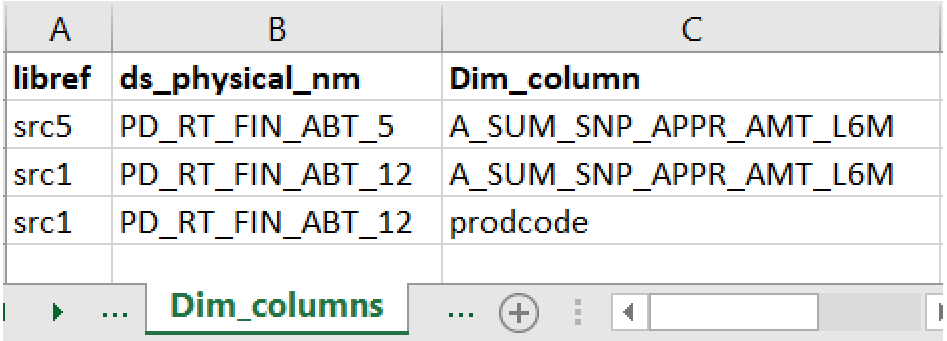


| libref | ds_physical_nm | ds_desc | data_aggregation | Primary_SOA | KeyType | dstype |
|---|---|---|---|---|---|---|
| src1 | pd_rt_fin_abt_1 | Retail Fin Acct 1 | Both | Retail Customer | retained | DS |
| src2 | pd_rt_fin_abt_2 | Retail Fin Acct 2 | Day | Retail Customer | retained | DS |
| src2 | pd_rt_fin_abt_21 | Retail Fin Acct 21 | Month |  | retained | DS |
| src3 | pd_rt_fin_abt_3 | Retail Fin Acct 3 | Month | Retail Financial Account | retained | DS |
| src3 | pd_rt_fin_abt_11 | Retail Fin Acct 11 | Day | Retail Financial Account | retained | DS |
| src4 | pd_rt_fin_abt_4 | Retail 4 | Both | Retail Financial Account |  | DS |
| src1 | pd_rt_fin_abt_12 | Retail Fin Acct 12 | Month | Retail Financial Account |  | DS |
| src5 | pd_rt_fin_abt_5 | Retail Fin Acct 5 | Both | Retail Customer |  | DS |

Display 8. Accelerator File: datasource

Here, ds_desc must contain unique values. Mandatory fields are libref, ds_physical_nm (data source physical name), and ds_desc (data source description). If values for data_aggregation, Primary_SOA, and keyType are not provided, then the user can configure them manually in the SAS Analytical Data Set Builder Configuration workspace within SAS Credit Scoring.

- Dim_columns: The user can provide a dimensional attribute variable name for all the values that need to be imported. Display 9 shows an example of the Dim_columns sheet with sample data.

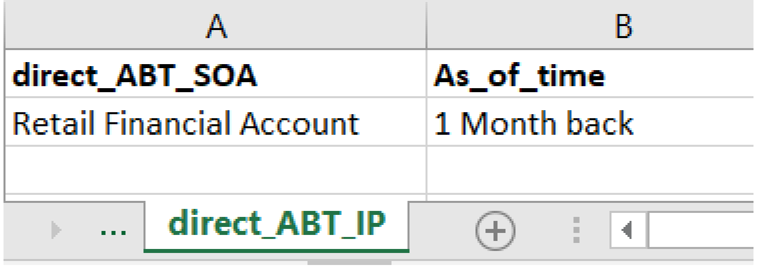


| A | B | C |
|---|---|---|
| **libref** | **ds_physical_nm** | **Dim_column** |
| src5 | PD_RT_FIN_ABT_5 | A_SUM_SNP_APPR_AMT_L6M |
| src1 | PD_RT_FIN_ABT_12 | A_SUM_SNP_APPR_AMT_L6M |
| src1 | PD_RT_FIN_ABT_12 | prodcode |

Display 9. Accelerator File: Dim_columns

- direct_ABT_IP: Once the data sources are configured successfully, the user has an option to create a default ABT with only point-in-time (PIT) variables. Doing so helps the user while creating an ABT; the user can directly import required PIT variables from the shared pool.

  The direct_ABT_IP sheet has only two columns, direct_ABT_SOA and As_of_time. Both are mandatory if a user wants to create a default ABT. Display 10 shows an example of a direct_ABT_IP sheet.

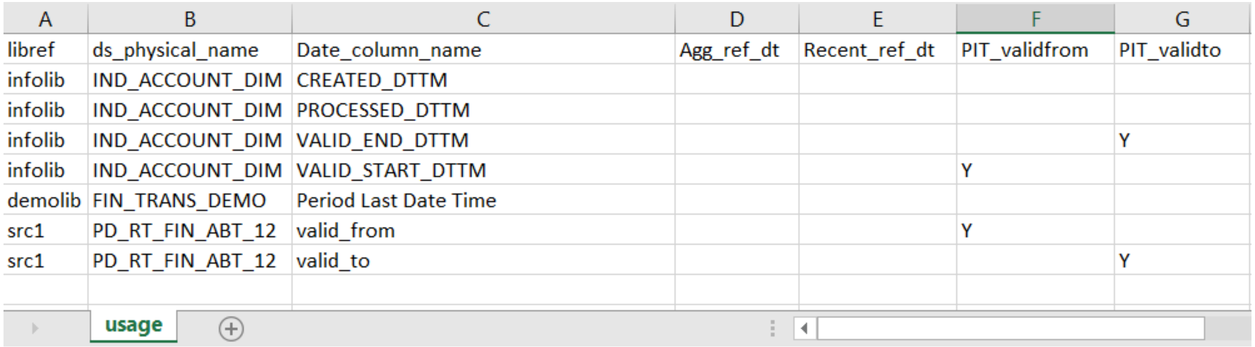


| A | B |
|---|---|
| **direct_ABT_SOA** | **As_of_time** |
| Retail Financial Account | 1 Month back |

Display 10. Accelerator File: direct_ABT_IP

## USING THE USAGE FILE

After the basic configuration is done, the next step is to configure the usage. A user can export the Usage.xlsx file using the SAS code (export_usage.sas) provided as a part of automation toolkit This file contains a list of all the data sources that are configured using the accelerator file and its variables if the type of variable is DATE.



| A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|
| libref | ds_physical_name | Date_column_name | Agg_ref_dt | Recent_ref_dt | PIT_validfrom | PIT_validto |
| infolib | IND_ACCOUNT_DIM | CREATED_DTTM | | | | |
| infolib | IND_ACCOUNT_DIM | PROCESSED_DTTM | | | | |
| infolib | IND_ACCOUNT_DIM | VALID_END_DTTM | | | | Y |
| infolib | IND_ACCOUNT_DIM | VALID_START_DTTM | | | Y | |
| demolib | FIN_TRANS_DEMO | Period Last Date Time | | | | |
| src1 | PD_RT_FIN_ABT_12 | valid_from | | | Y | |
| src1 | PD_RT_FIN_ABT_12 | valid_to | | | | Y |

Display 11. Usage File

As you can see in Display 11, all the data sources are listed along with their libref and all the variables of type DATE. Furthermore, this sheet has four new variables, Agg_ref_dt (Reference date for Aggregate variables), Recent_ref_dt (Reference date for Recent variables), PIT_validfrom (VALID FROM reference date for point-in-time variable) and PIT_validto (VALID TO reference date for point-in-time variables). These four variables are flags that the user has to set to Y if the user wants to configure that data source for a particular usage. For example, in Display 11, consider data source PD_RT_FIN_ABT_12. It has two DATE type variables, valid_from and valid_to. The user can use this data source for creating point-in-time variables if PIT_validfrom and PIT_validto flags are set.

### USING SAS PROGRAMS

Different SAS programs are written for specific implementation tasks. These programs are executed in a pre-defined sequence through the master.sas file.

```
/*Configure library*/
%include "&path.\codes\config_lib.sas";

/*Configure data source*/
%include "&path.\codes\config_ds.sas";

/*Assign variable roles*/
%include "&path.\codes\config_var_roles.sas";

/*Assign aggregation level and subject of analysis*/
%include "&path.\codes\config_agg_soa.sas";

/*Import dimensional attribute values*/
%include "&path.\codes\config_dim_attrib.sas";
```

The code above performs the basic configuration for external data sources. If a data source usage also needs to be configured, then it is recommended that you first export the usage.xlsx sheet, and then run the config_usage.sas program. To do this, you must execute the following two programs:

export_usage.sas: In the exported file, you must update the required flags in usage.xlsx (see the section ‟The Usage File”).

config_usage.sas: This code configures the usage as per usage.xlsx.

### CREATING A DEFAULT POINT-IN-TIME ABT

There is also an option to create a default point-in-time (PIT) or direct ABT that consists of all the variables from all the data sources just configured and whose usage is configured as PIT. Also, the data source must be attached with the subject of analysis (SOA) for which the PIT ABT is being created.

Consider this example. In Display 10 on the direct_ABT_IP sheet of the accelerator file, the direct_ABT_SOA is Retail Financial Account. If there are three data sources A, B, and C that are configured for PIT usage and that have the same SOA as that of PIT ABT, then the default PIT ABT contains all the variables from the three data sets A, B, and C.

Finally, the config_direct_abt.sas code for creating the default PIT ABT must be run. This ABT will be shared across all the users.

## CONCLUSION

The approach described in this paper, along with best practices mentioned, have shown to significantly reduce implementation timelines for some SAS Credit Scoring customers. The improvements that have been noted include the following:

- significant improvement in efficiency (about 90%) through the use of the automation toolkit
- a reduction in errors (typically resulting from manual configuration) via automation
- reduction in overall configuration complexity through usage of the accelerator file
- readily available ABT for PIT variables
- by using a phased approach, the implementation is done in multiple iterations. Each particular feature is available to the customer for production use at the end of each cycle. Thus, the customer can start using SAS Credit Scoring in a very short time.

The combination of a better designed credit scoring solution, along with a more efficient implementation methodology, can result in faster deployment timelines for the solution for our customers—from months to weeks.

## RECOMMENDED READING

*SAS® Credit Scoring 6.3 for Banking: User's Guide*

*SAS® Credit Scoring 6.3 for Banking: Administrator's Guide*

## ACKNOWLEDGMENTS

I would like to thank Deepti Charankar and Rajesh Khanwelkar for encouraging and supporting me to write this paper. I would also like to thank all the consultants from the field who provided their feedback about the product and process, which helped us in figuring out the pain areas and providing solutions to them. Furthermore, I am thankful to Naeem Siddiqi and Andrés Alvarez for providing constructive feedback at every stage to make this paper better. Finally, I would like to acknowledge the following individuals for their inputs in developing automation toolkit:

- Amol Malani
- Sandeep Garg
- Sanchit Singla

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. In addition, you can request the SAS codes for the automation toolkit. Contact the author at:

Abhijit S. Joshi
SAS Research and Development India Ltd.
+91 20 4911 8790
Abhijit.s.joshi@sas.com