

# SAS® Visual Forecasting: A Cloud-Based Time Series Analysis and Forecasting System

Jerzy M. Brzezicki and Joseph H. Katz, SAS Institute Inc.

## ABSTRACT

SAS® Visual Forecasting, based on SAS® Viya®, is the next generation SAS® product for forecasting. It provides a new resilient, distributed, scripting environment for cloud computing that provides time series analysis, automatic forecast model generation, automatic variable and event selection, and automatic model selection. SAS Visual Forecasting features a new graphical interface that is centered on the use of pipelines, a new microservices-based architecture, and a new fast, scalable, and elastic in-memory server environment based on SAS® Cloud Analytic Services (CAS). It provides end-to-end capabilities to explore and prepare data, apply various modeling strategies, compare forecasts, override statistical forecasts, and visualize results. The workflow framework for model generation and forecasting is shared with SAS® Visual Data Mining and Machine Learning and SAS® Visual Text Analytics. Forecast analysts and data scientists can also access the power of SAS Visual Forecasting through a flexible and powerful programming environment.

## INTRODUCTION

As the newest generation of SAS forecasting products, SAS Visual Forecasting provides a new resilient, distributed, time series analysis and scripting environment for cloud computing that is fine-tuned for conducting advanced time series analysis, modeling, and forecasting. SAS Visual Forecasting exploits the speed, scalability, and elasticity of the SAS Viya in-memory engine.

One of the key innovative concepts of SAS Visual Forecasting, which allows it to work on such an enormous scale, is the efficient “shuffling” of the time series data across nodes in a distributed cluster. Distributed file systems usually break up large files into chunks and store these chunks across many nodes. This arrangement works well for some analytical processes, but it is not appropriate for time series analysis. Time series data need to be processed by the order of their associated timestamps. In a nondistributed environment, data are sorted by their timestamps (time IDs) and accumulated to the desired frequency before the analysis is performed. In a distributed cluster, this step would negate the benefit of having distributed data. To rectify this issue, SAS Visual Forecasting “shuffles” the data so that each time series, uniquely identified by its combination of classification (BY) variable values, is copied into the memory of a single worker node or onto a single thread of a worker node. No sorting is ever required.

Forecasters can encounter huge data sets. A large retailer might have tens of thousands of products to forecast across hundreds or thousands of stores. But they no longer need to sort their historical data because SAS Visual Forecasting will handle their data automatically and in shared memory.

At the core of SAS Visual Forecasting is the TSMODEL procedure, which provides advanced support for time series analysis, time series decomposition, time series modeling, and signal analysis and anomaly detection (which is useful for the Internet of Things). Designed to be highly scalable, SAS Visual Forecasting makes one pass through the data to do multiple analyses. Its scripting language is optimized and compiled based on the machine it runs on, and the scripts are distributed to the available hardware. SAS Visual Forecasting provides large-scale automatic forecast model generation, automatic variable and event selection, automatic parameter optimization, automatic champion model selection from a set of candidate models, and forecast generation.

SAS Visual Forecasting is an HTML5 web-based application that uses REST (representational state transfer) API microservices. The graphical user interface supports a process workflow approach via pipelines for large-scale automatic modeling and forecasting. This process workflow approach to modeling is part of a consistent end-to-end common-experience framework, called Model Studio, which is shared with SAS Visual Data Mining and Machine Learning and SAS Visual Text Analytics.

For forecast analysts and data scientists who prefer to write code, the new forecasting procedures provide a programming entry point. Coders can also modify the code from within the user interface to customize the modeling strategy nodes to meet their specific needs. SAS Visual Forecasting introduces an open platform that utilizes the REST API microservices. These APIs and their underlying actions can be called from Python, R, Java, and Lua in addition to the SAS language.

SAS Visual Forecasting is also a collaborative work environment that enables sharing of reusable components such as nodes, pipeline templates, and projects. You can create custom modeling strategy nodes and pipeline templates and share them with your colleagues within Model Studio. You can download projects from one installation and upload them to another installation. However, the data sources are not included with the project download and must be moved independently because of the potential size of the data tables.

SAS Visual Forecasting contains a new Overrides Module that enables you to influence or modify the statistical forecasts. Using a faceted search filter structure, you create filters that are based on a combination of attributes such as location, brand, category, size, color, sentiment, quality, and so on. Override instructions are defined by the filter and one or more time periods, and they affect all time series that are contained within the filter. These attribute-based filters are not limited to the variables that are used for hierarchical forecast modeling.

Proper data preparation is a prerequisite for all modeling activities. Data preparation could involve activities such as joining or merging a number of tables, computing calculated columns, and converting a character date field to a numeric date field. You can use SAS® Data Studio to perform the necessary data preparation activities and then seamlessly move into Model Studio. This integration makes it easy to transition from the data preparation activity to the modeling activity within SAS Visual Forecasting.

## EVOLUTION OF SAS FORECASTING SOFTWARE

### TIME SERIES FORECASTING SYSTEM

The Time Series Forecasting System (TSFS) was an interactive point-and-click application that was introduced with SAS 6.12 in November 1996. It had no BY-group support, and you could forecast only one time series at a time. Interactive modeling capabilities with support of ARIMA and ESM models together with analysis views were available within the application.

### SAS® FORECAST SERVER

SAS Forecast Server, released with SAS 9.1.3 in April 2006, includes a client-server application called SAS® Forecast Studio, an application called SAS® Time Series Studio, and more than 10 high-performance forecasting procedures. The high-performance procedures (HPFDIAGNOSE, HPFENGINE, HPFARIMASPEC, HPFESMSPEC, HPFEVENTS, HPFEXMSPEC, HPFIDMSPEC, HPFSELECT, HPFRECONCILE, HPFTEMPRECON, and HPFUCMSPEC) cover the full forecasting cycle: diagnostics, automatic model generation and selection, events support, and interactive modeling. With built-in BY-group support, they provide support for an automated forecasting process for large-scale data and support for modeling hierarchies. The procedures were first released in SAS 9.1.3 in December 2004.

SAS Forecast Studio simplifies and automates the lifecycle of a forecasting process. It automatically accumulates data according to the detected time interval and prepares the data for hierarchical aggregation. It selects significant input variables, automatically generates forecasting models for each time series, and selects the champion model. Each step of the forecasting process has a view associated with it: a hierarchy view to navigate the data, a forecasting view to review forecasts, a series view to examine series properties, a modeling view to analyze and compare models, a scenario analysis view to run multiple “what if ...” analyses, and an exceptions view to filter forecasts. SAS Forecast Studio supports hierarchical forecasting with reconciliation. You can view the SAS code that is generated by most of the user interface actions. Forecasting processes can be run in a production environment by using SAS batch code. In addition, a set of utility SAS macros helps maintain the life cycle of forecasting projects.

SAS Time Series Studio is an application that focuses on extensive time series exploration capabilities. Its user interface provides advanced and intuitive tools for time series exploration. Results of time series exploration and data preparation can be easily integrated with SAS Forecast Server. Combining both SAS Forecast Studio and SAS Time Series Studio provides a very powerful set of tools for forecasting modelers.

## **SAS VISUAL FORECASTING**

SAS Visual Forecasting, released with SAS Viya 3.2 in December 2017, is the latest SAS forecasting product. An evolution over the SAS Forecast Server application in the direction of cloud computing, it exploits all aspects of the scalable architecture of SAS Viya. The forecasting process is supported by a new set of procedures and CAS actions. The key procedure, PROC TSMODEL, provides a framework for time series analysis and forecasting in SAS Viya. You can execute steps of the forecasting process, such as diagnostics and forecasting, in a single pass through data that reside in memory on a CAS server.

The evolving trend of SAS forecasting software shows that each generation supports larger volumes of data than the previous generation. In addition, each generation covers more aspects of the forecasting process and is focused on an approachable analytical life cycle. The latest release of SAS Visual Forecasting focuses on collaboration among SAS analytical tools, integration in an open ecosystem, support for multiple languages through REST APIs, unconstrained navigation and modeling techniques that facilitate the power of unstructured data and their attributes, and parallel execution of differently configured forecasting processes through multiple pipelines.

## **SAS VISUAL FORECASTING END-TO-END FORECASTING PROCESS**

### **TIME SERIES AND ATTRIBUTE DATA TABLES**

SAS Visual Forecasting separates the data into two primary input tables in order to avoid duplication of values and to reduce the overall footprint of the storage required:

- The time series table contains the classification variables (BY variables) that uniquely define each time series, in addition to the time ID variable, the dependent variable, and any potential independent variables to be considered during forecast modeling. The dependent and independent variables are time-varying. That is, the value that is associated with each variable generally changes as the corresponding date changes.
- The attributes table contains information that is specific to the time series but is not time-dependent; it contains characteristics or properties of each time series. Some examples of attribute data include variables such as supplier, product type, venue, and product category. Other examples include variables such as location, brand, category, size, color, sentiment, and quality. Many of these variables can contain unstructured data.

Figure 1 shows the relationship between the time series data table and the attributes data table. The classification (BY) variables in the two tables must match because they link the tables together. In Figure 1, the time series data table contains more than 147,000 rows whereas the attributes data table contains only 1,337 rows. Separating time-varying variables and time-invariant variables into two tables allows for a more compact storage of the data. If the data were kept in one table, that table would contain a tremendous amount of redundancy.

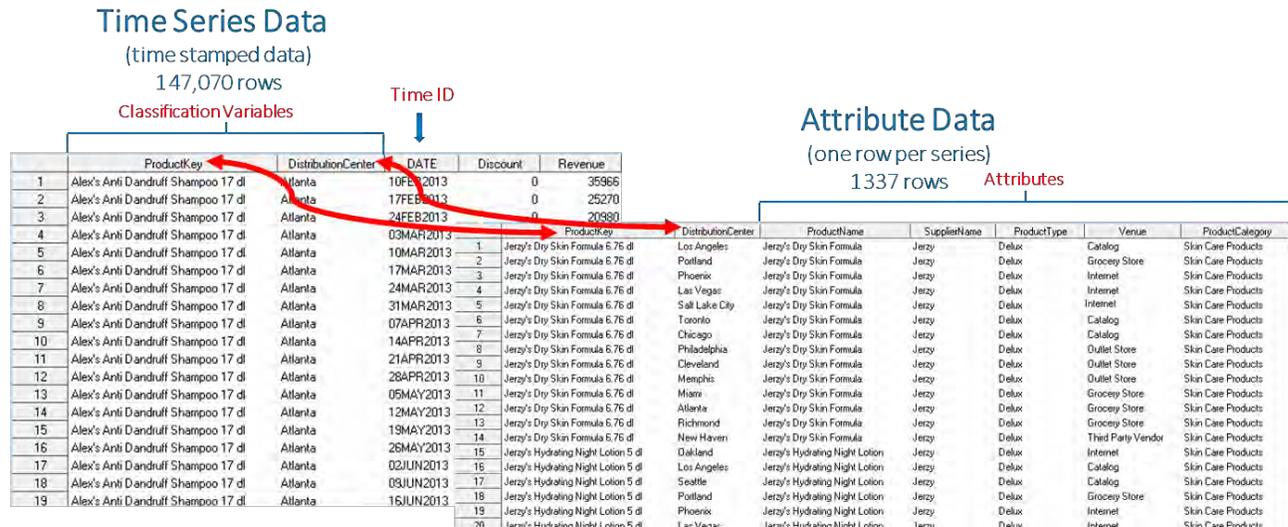


Figure 1. Time Series Data Table and Attributes Data Table

## CREATE A PROJECT

The first step in analyzing your data is to create a project. In the Model Studio user interface (shown in Figure 2), click **New Project**.

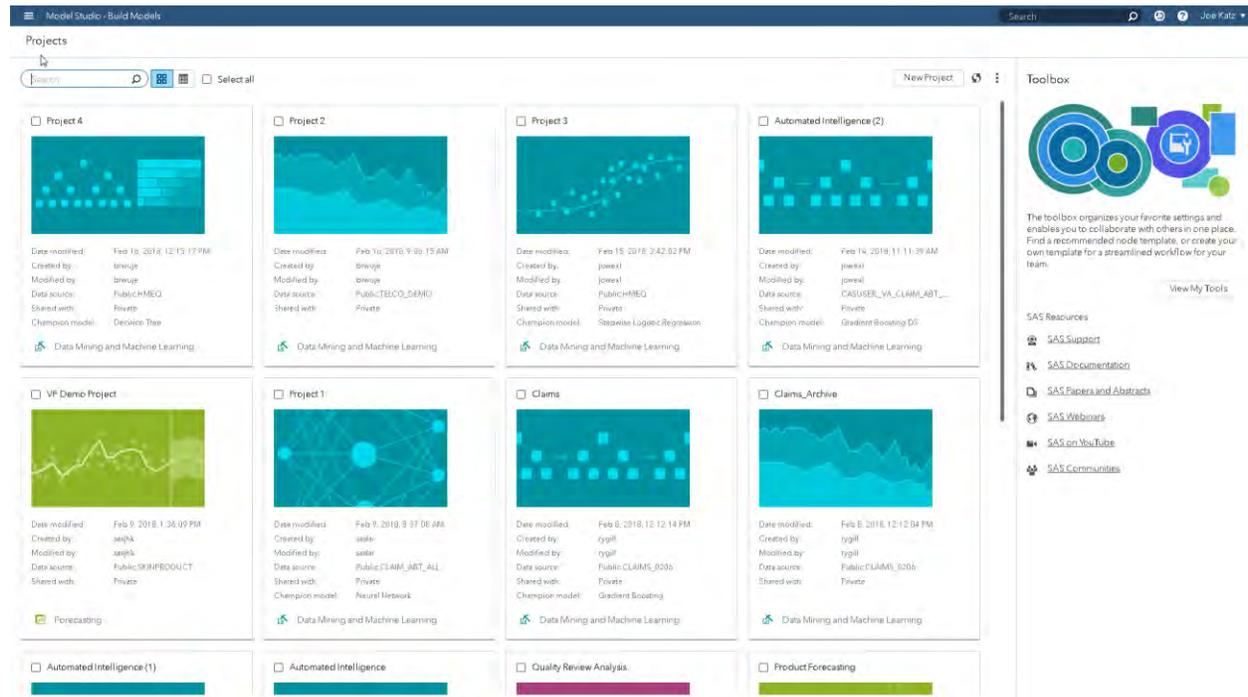


Figure 2. Model Studio Project Window

The New Project dialog box appears, as shown in Figure 3.

The dialog box is titled "New Project" and contains the following fields:

- Name:** A text input field containing "Project 1".
- Type:** A dropdown menu with "Forecasting" selected.
- Data source:** A text input field containing "Public.SKINPRODUCT" and a "Browse" button to its right.
- Description:** A large text area with the placeholder text "Enter description".

At the bottom right, there are two buttons: "Save" (in blue) and "Cancel".

**Figure 3. New Project Dialog Box from Model Studio**

Name your project (a default project name is initially shown), select **Forecasting** in the **Type** list, specify (or browse for) the data you want to analyze in the **Data source** field, and click **Save**. Model Studio then displays the Data tab in a Build Models window, as shown in Figure 4.

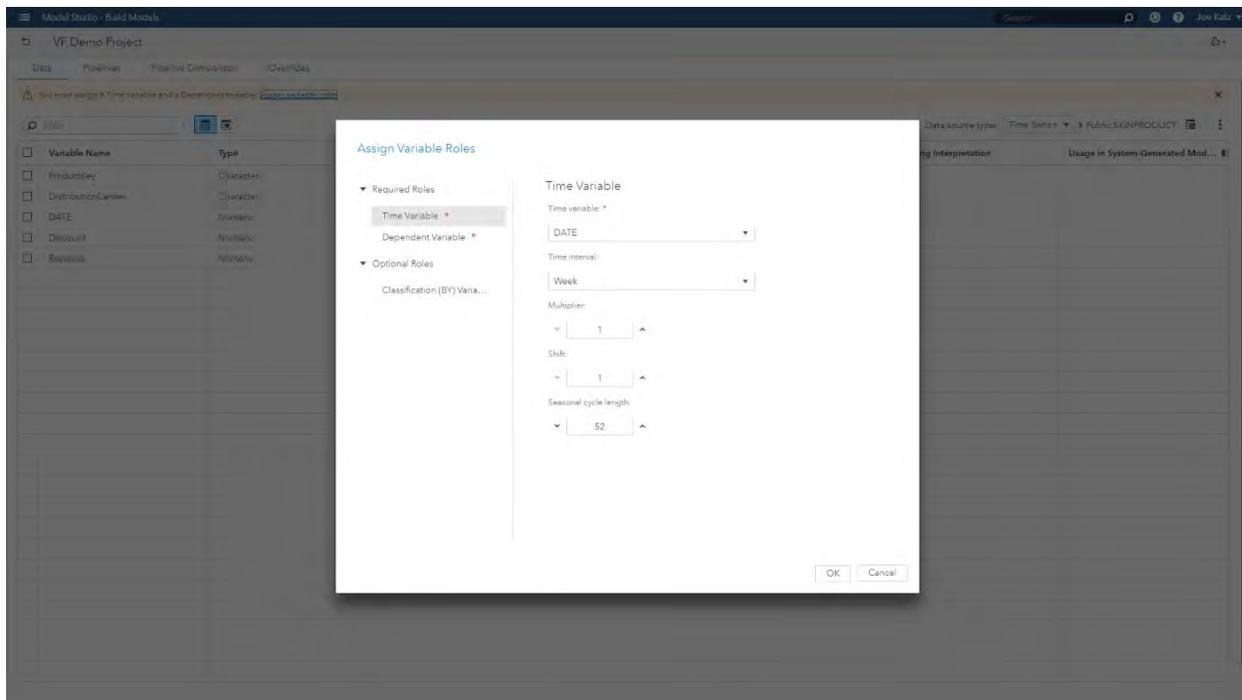
The screenshot shows the "Data" tab in the Model Studio interface. A yellow warning banner at the top states: "You must assign a Time variable and a Dependent variable. [Assign variable roles](#)".

Below the banner is a table with the following columns: Variable Name, Type, Role, Hierarchy Aggregation, Time Interval Accumulation, Missing Interpretation, and Usage in System-Generated Models. The table contains the following data:

Variable Name	Type	Role	Hierarchy Aggregation	Time Interval Accumulation	Missing Interpretation	Usage in System-Generated Models
<input type="checkbox"/> ProductKey	Character	Unassigned				
<input type="checkbox"/> DistributionCenter	Character	Unassigned				
<input type="checkbox"/> DATE	Numeric	Time				
<input type="checkbox"/> Discount	Numeric	Unassigned				
<input type="checkbox"/> Revenue	Numeric	Unassigned				

**Figure 4. Project Creation Data Tab**

Click **Assign variable roles** to open the Assign Variable Roles dialog box, as shown in Figure 5.



**Figure 5. Assign Variable Roles**

In this dialog box, you select and assign roles for your required variables. These roles include the dependent (target) variable that you are trying to forecast, the time ID variable, the time interval of interest (weekly, monthly, or other), the classification (BY) variables that uniquely identify each time series, and the sequence of the BY variables (from top to bottom) that specify the order to be used for hierarchical modeling.

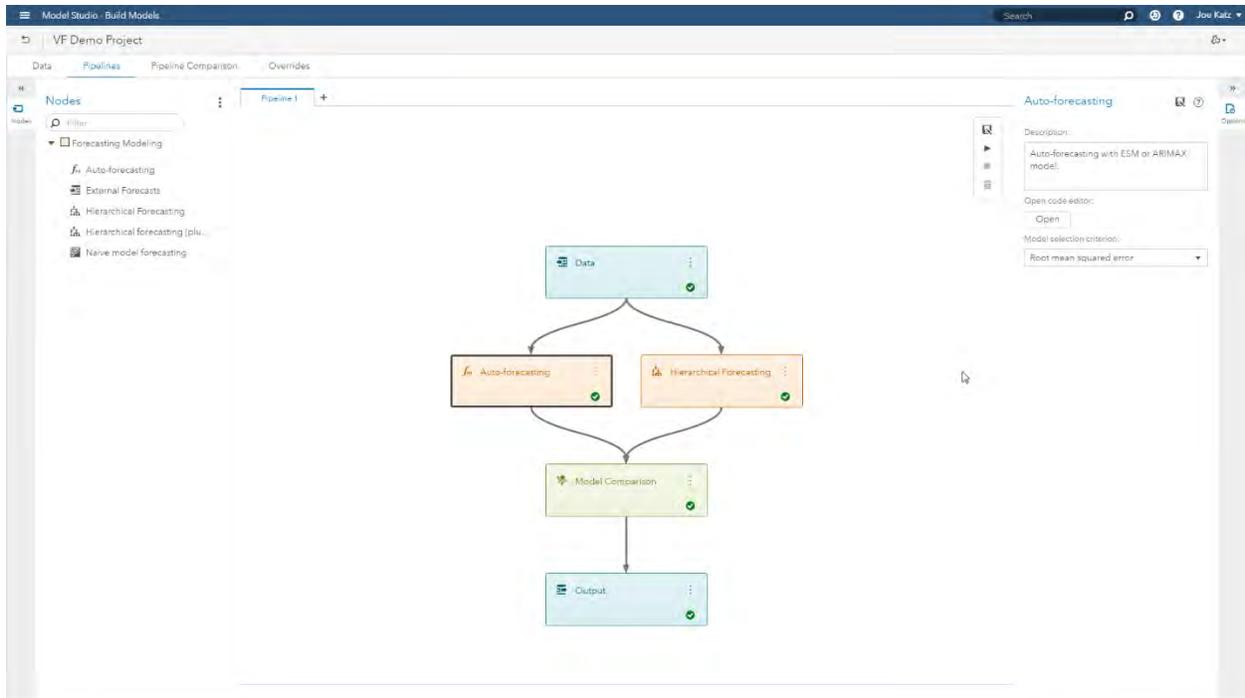
## LAYOUT AND TABS MATCH THE PROCESS STEPS

The SAS Visual Forecasting user interface is designed to guide you through the overall forecasting process. Figure 6 shows the three-panel layout within Model Studio for SAS Visual Forecasting. The **Nodes** panel (on the left) shows the available modeling strategy nodes that can be placed or inserted onto the process flow diagram that is shown on the middle panel. (To place a new node onto the flow diagram, drag it onto the Data node; to insert a new node, click the three vertical dots icon and select **Insert below**.) The properties panel (on the right) displays information about a selected node; it enables you to change some of the parameters for that node and it enables you to open the node to edit the underlying code. Both the nodes and properties panels can be collapsed.

Model Studio for SAS Visual Forecasting has the following four tabs, which represent a high-level view of the sequence of steps that you follow during the forecasting process:

- **Data:** After you create a project, Model Studio opens to this tab. You can make further variable assignments such as changing variables from a role of **Unassigned** to **Independent variable**. After the variable roles have been assigned, you can move to the **Pipeline** tab.
- **Pipeline:** On this tab, you build the process workflow and begin forecast modeling.
- **Pipeline Comparison:** On this tab, you can compare pipelines, as described in the section “Build Pipelines.”
- **Overrides:** On this tab, you can influence or modify the baseline statistical or final forecasts with improved forecasts, as described in the section “Overrides.”

These four tabs are shown in Figure 6 and are described in more detail in the following sections.



**Figure 6. Model Studio Three-Panel Display**

## PREPARE DATA

Properly preparing your data for any analytical activity, whether it be for exploratory data analysis or modeling, is of the utmost importance. In most cases, data preparation activities can consume as much as 80% of the time spent for analytical projects. The same is true for forecasting activities.

You can use SAS Data Studio to perform necessary data preparation activities and then seamlessly move into Model Studio. This integration makes it easy to transition from the data preparation activity to the modeling activity.

In the example shown in Figure 7, a character date field has been converted into a numeric date field as required by SAS Visual Forecasting for the time ID variable. After you have saved the new data table, you can navigate seamlessly from SAS Data Studio to Model Studio by clicking the three vertical dots icon and selecting **Actions** → **Saved table** → **Build Models**. The project creation dialog box appears with the data table populated, as illustrated in Figure 3 **Error! Reference source not found.**

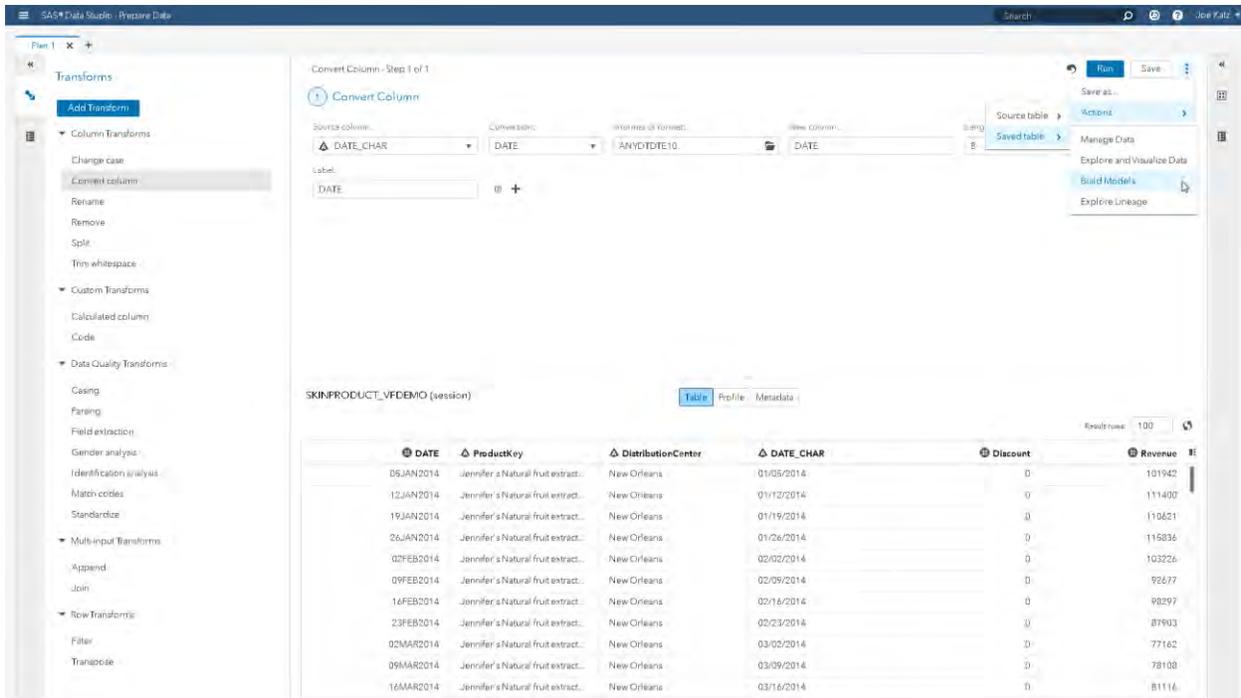


Figure 7. Preparing Data in SAS Data Studio

## EXPLORE TIME SERIES DATA BY USING ATTRIBUTE-BASED FACETED SEARCH

Before you begin any modeling activity, it is important to be able to explore your data. For forecasting, exploration involves looking for trends, seasonality, intermittency, and other patterns.

After you have run the Data node, you can explore the data in the Explore Time Series window, which you can reach by selecting **Explore time series** from the list that appears when you either right-click on the Data node or click the three vertical dots icon. The Explore Time Series window is shown in Figure 8.

On the left side of the window is a list of the attributes from either the default attributes table (BY variables only) or a user-specified attributes table. You can expand the attributes to show a list of attribute values, which are called facets. For example, you can expand the **Product Key** attribute to show four different facets (Jennifer's sunscreen of various types), and you can click **More** to show the full list.

Attributes are variables that represent characteristics about a time series. Some examples include variables such as location, brand, category, size, color, sentiment, and quality. Many of these variables can contain unstructured data. The faceted search filters use unstructured data for dynamic search. As you navigate through the attributes and create filters, the set of time series contained by the selection is dynamically updated along with the graph and the table. See Figure 8.

On the right side of the window is a graphical display of the data above a tabular display of the data. Both the diagram and table show the aggregation of historical data for all the time series.

By expanding the desired attribute variables and selecting one or more facets, you can display and explore the aggregation of the data, looking for patterns. The graph and table dynamically update when you select or deselect a specific facet value, reset a specific attribute to no selection (by clicking **Reset**), or reset all attributes to no selection (by clicking **Reset all**).

Above the graph, bread crumbs are displayed to show which facets have been selected. In Figure 8, the bread crumbs show that the **Jennifer** supplier and the **Deluxe** product type have been selected. From the plot, you can easily see that this combination of data has both a seasonal pattern and a decreasing trend.

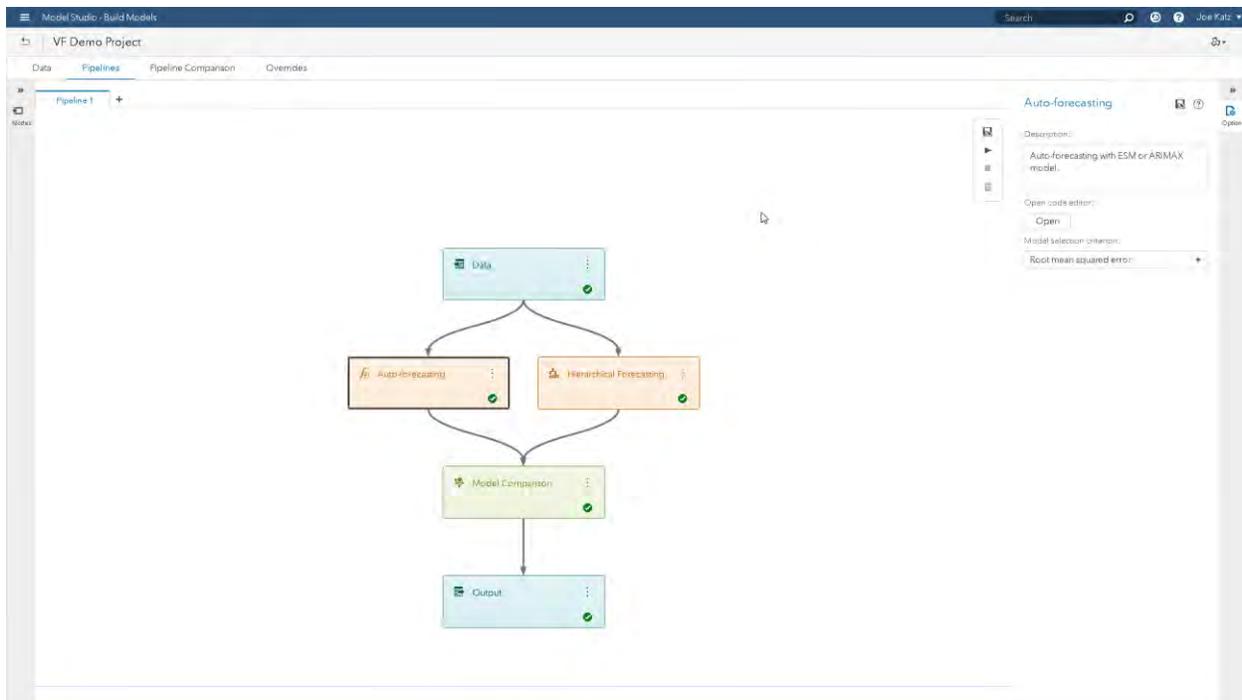
You can also explore the data in order to detect anomalies or inconsistencies within the data. If you are expecting data to be seasonal and they are not, you might have a problem with the data source or the data preparation process.



Figure 8. Explore Time Series

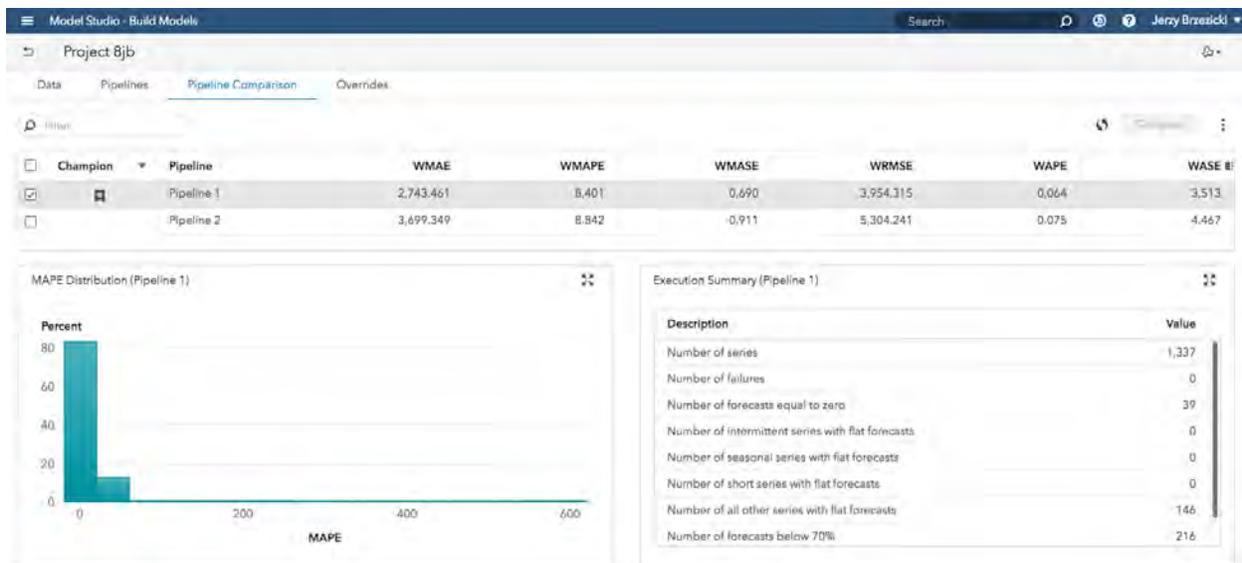
## BUILD PIPELINES

A pipeline in a SAS Visual Forecasting project represents an autonomous forecasting process. The scope of a SAS Forecast Studio project is analogous to the scope of one linear pipeline in SAS Visual Forecasting. A pipeline starts with a data input node and ends with a data output node. A forecasting pipeline has three stages: data-related activities (such as data accumulation and handling missing values), time series modeling, and postprocessing, as illustrated in Figure 9.



**Figure 9. Pipeline with Two Modeling Node Branches**

A set of prebuilt forecasting pipeline templates is available for you to select from. You can decide which template to use by default in the Toolbox. See the section “The Toolbox Collaborative Environment.” For a SAS Visual Forecasting project that contains multiple pipelines, you can go to the **Pipeline Comparison** tab (shown in Figure 10) to compare pipeline results and to select a champion pipeline. Multiple pipelines are a convenient way to try different settings or modeling approaches and compare results. To achieve this comparison in SAS Forecast Server, you often needed to create multiple projects and write custom code to perform the comparison. If a SAS Visual Forecasting project with multiple pipelines is executed, each pipeline runs in parallel. As a final step in pipeline processing, forecasting results can be exported to any available library on the CAS server.



**Figure 10. Pipeline Comparison Tab**

## Modeling in a Pipeline

You can modify the default pipeline by replacing the default modeling node with any forecasting modeling node that is available in the Toolbox. Each modeling node has list of settings and properties that are associated with it. You can modify these settings and properties by using the properties panel on the right side. After pipeline development is complete, you can store a reusable version of it in the Toolbox as a pipeline template. Pipeline templates can be used across multiple projects.

More than one modeling node can be attached to the data input node, resulting in a branch of modeling nodes. Figure 9 shows a pipeline that contains two modeling node branches. You can compare two or more modeling nodes within one pipeline by using the Model Comparison node. Each modeling node within a pipeline executes in parallel.

## Pluggable and Portable Custom Modeling Strategy Nodes

In addition to prebuilt automatic modeling strategy nodes, you can create your own modeling strategy and plug it into the system. This very powerful feature of SAS Visual Forecasting enables you to create custom, user-defined modeling approaches.

You can develop code for a pluggable modeling node, as shown in Figure 11. When you select a node within a pipeline, click **Open** in the properties panel to open a dialog box that displays the code. To run your node, you define a contract between the system and your custom code that involves input variables (which define the data available to a modeling node) and a set of output variables (which need to be assigned so that the results of the modeling node are passed down the pipeline). Except for using input and output variables, you have full control over the pluggable modeling code. You can edit custom code within a pipeline. You can export a custom modeling node from one installation as a portable version and import it to another installation.

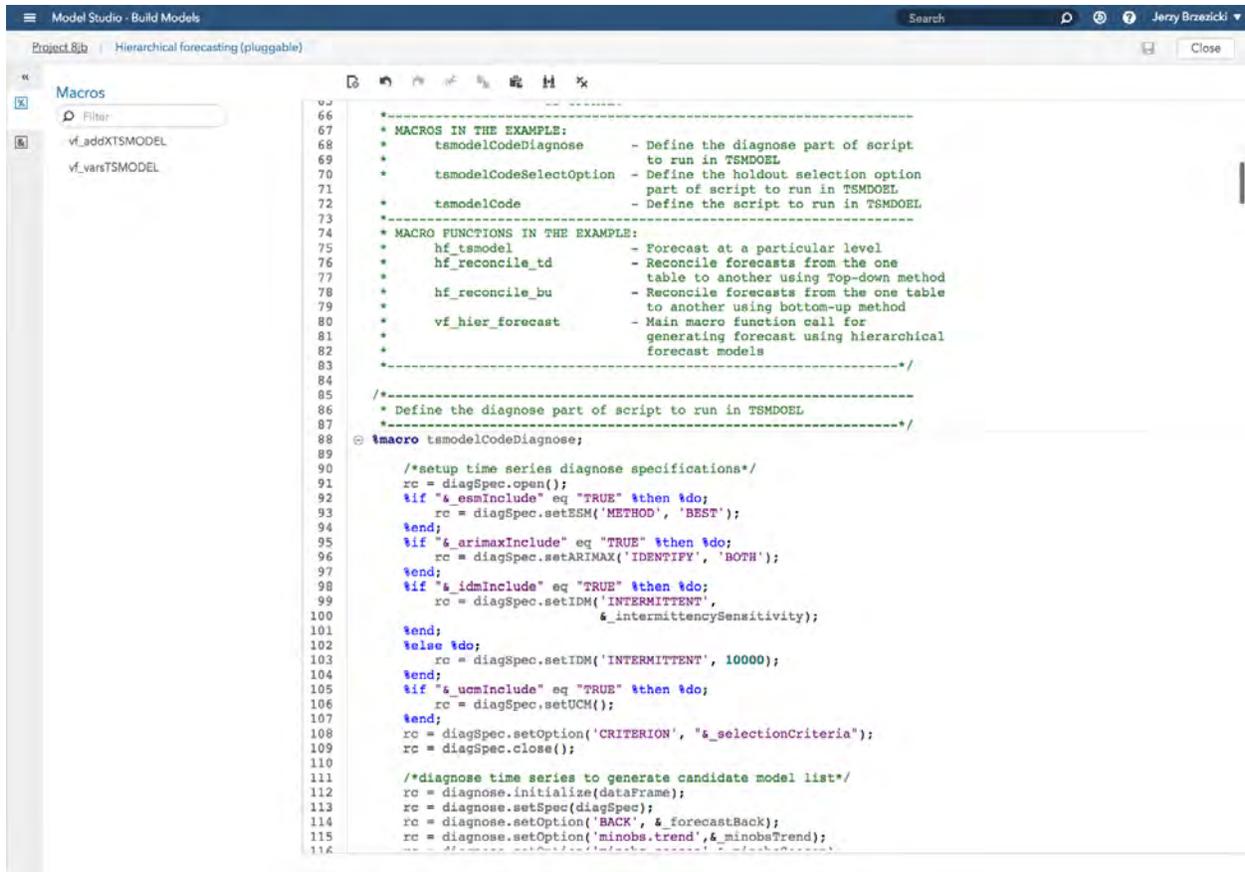


Figure 11. Code Editing Dialog Box for Pluggable Modeling Node

## Overrides

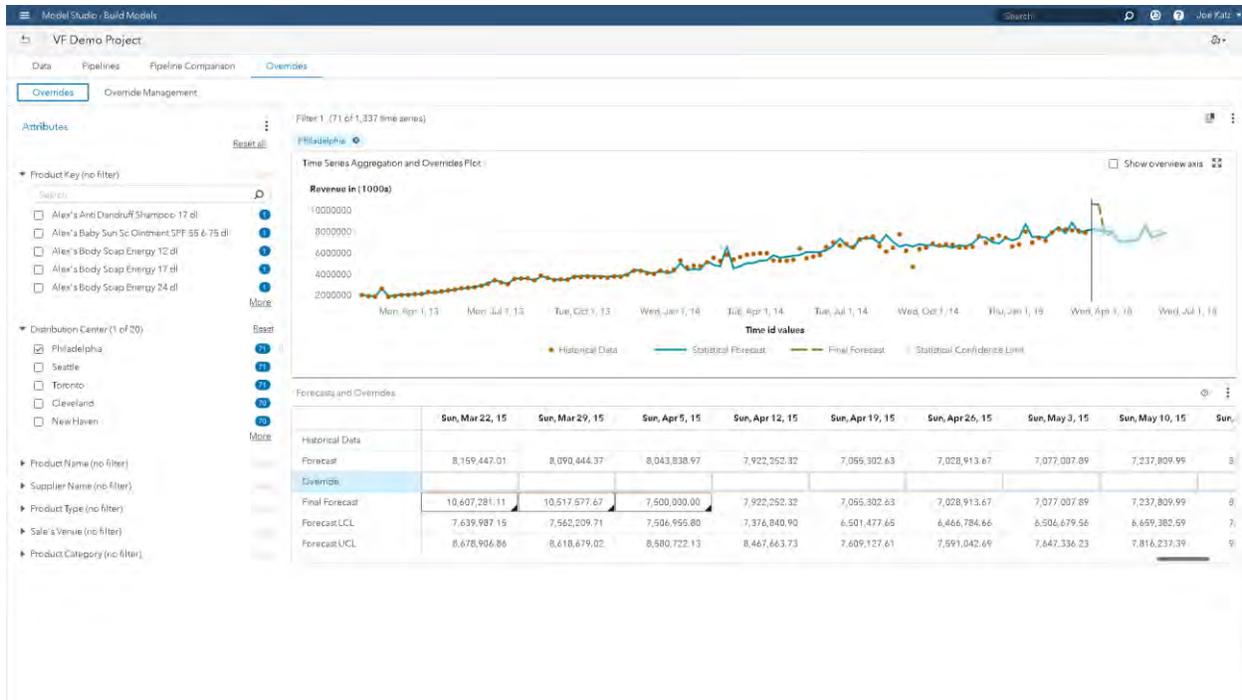
Overrides enable you to influence or modify the baseline statistical or final forecasts with enriched forecasts. This capability enables you to incorporate into the forecasting process judgmental forecasts or additional information that might not have been available at the time the statistical models and forecasts were generated. Examples include competitor or market actions, upcoming promotions, and restrictions such as production constraints. In addition, overrides are often part of a formal consensus forecasting process, a Sales and Operations Planning (S&OP) process, or an Integrated Business Planning (IBP) business process. The ultimate purpose of using overrides is to improve overall forecast quality.

The **Overrides** tab has two views:

- The **Overrides** view enables you to create or edit overrides, or to delete pending overrides. When you have finished defining an override instruction, you must submit it in order to update the forecasts. After you have submitted an override, it is moved from a pending state to an applied state, and the final forecast for that time period reflects the resolved value.
- The **Override Management** view enables you to view all overrides that have been either submitted (applied) or defined but not submitted (pending), to edit overrides that have been submitted, or to resolve conflicts between overrides. You can also delete overrides that have been applied or resubmit overrides that have been affected by other activity.

The statistical forecasts that are shown for each time series in the **Overrides** view are based on the champion pipeline that is selected on the **Pipeline Comparison** tab. All the pipelines should have completed execution and the pipeline champion should be selected before you start working in the overrides module. If a new champion pipeline is selected after you start working with overrides, you are prompted to refresh the override data and resubmit any overrides that have been applied.

The **Overrides** view in **Error! Reference source not found.** is structured similarly to the Explore Time Series window that can be selected from the Data node in the pipeline. On the left side of the view is a list of the attributes. You can expand the attributes to show a list of attribute values, which are called facets. For example, you can expand the **Distribution Center** attribute to show five different facets, and you can click **More** to show the full list.



**Figure 12. Overrides Applied to Forecasts for Philadelphia Distribution Center**

As you navigate through the attributes and select facets, you are creating filters. The filter selection contains a set of time series. On the top right side of the view is a graphical display of the historical data, the model fit to history, the statistical forecast, the upper and lower confidence limits, and the final forecast. The graph and table dynamically update when you select or deselect a specific facet value, reset a specific attribute to no selection (by clicking **Reset**), or reset all attributes to no selection (by clicking **Reset all**). Above the graph, breadcrumbs are displayed to show which facets have been selected. In **Error! Reference source not found.**, the breadcrumbs show that the **Philadelphia** distribution center is selected. Both the diagram and table show the aggregation of historical data for all the time series when no facets are selected.

Below the graph is a tabular display of the same data elements along with a row named “Override” for specifying override instructions. The following section, “Overrides Instructions,” explains how to create the override instructions and the options available to you.

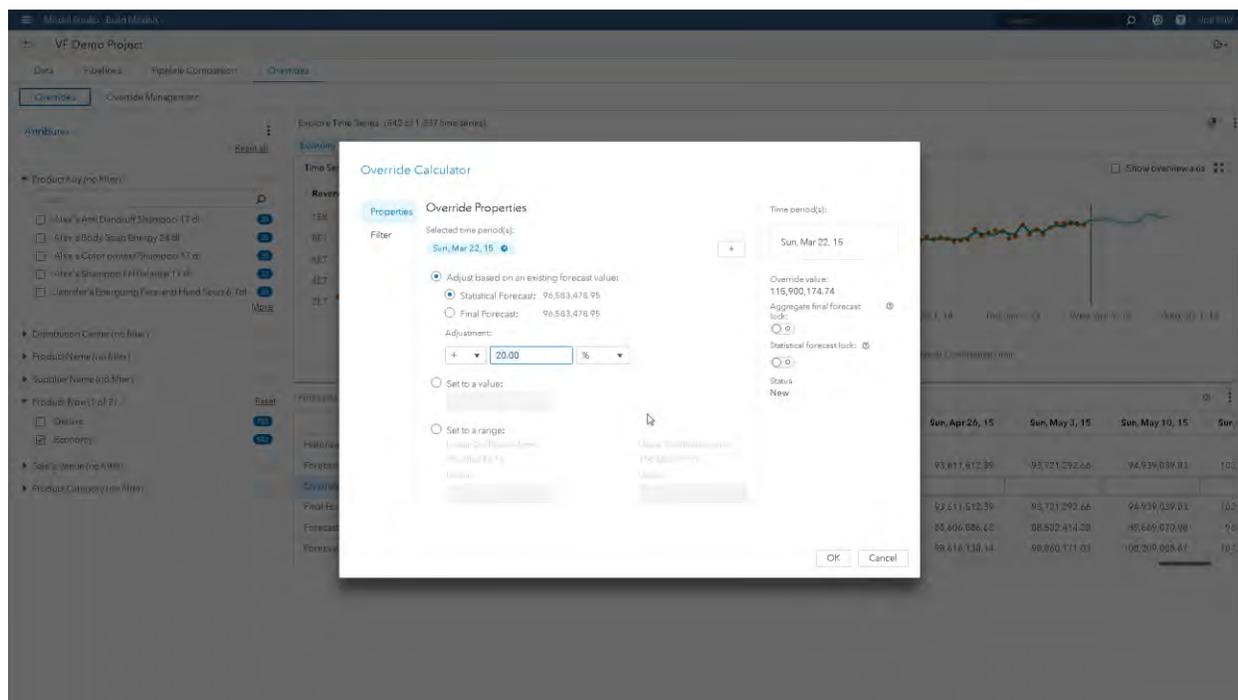
In SAS Visual Forecasting, you are no longer restricted to making overrides only according to the variables that are used in the forecasting modeling hierarchy. You can apply overrides to groups of time series based on a combination of time series attributes or BY variables (or both).

Consider a simple example. The forecasting modeling hierarchy consists of the variables region, store, and SKU. If a particular brand is going on promotion and you expect the sales forecast to increase by 15%, you need to be able to increase the forecast for each product (time series) within that brand. Entering overrides in the hierarchical context means that you would have to manually adjust the forecast for each individual product of that brand. There could be hundreds or thousands of such products!

With SAS Visual Forecasting, you can select the desired filter based on brand, specify the time periods that are affected, and create and apply the override instruction. You do not need to search for each product within that brand and individually make override adjustments.

## Override Instructions

After you use facted search to select the attribute values and you select the time periods to be affected, you can specify override instructions. You can enter either a number or a range in a specific cell. Alternatively, you can open the Override Calculator, which provides options for creating and editing specifications, as shown in Figure 13.



**Figure 13. Override Calculator**

In addition to the time period already selected, you can add multiple time periods. You can modify the forecast based on either the statistical forecast or the final forecast.

Adjustments to the forecast are computed either as a percentage or as a number of units from the statistical or final forecast value. You can increase or decrease the amount of the adjustment.

You can set override instructions to a fixed value. In this case, the specific value entered is to be considered during optimization for the final forecast value. Or you can set override instructions to a range. In this case, you specify lower and upper values to be considered during optimization for the final forecast value.

### Lock and Unlock Overrides

You can choose to lock an override value or the statistical forecast for a particular filter and one or more particular time periods. A locked override requires that the instruction for the override value for a forecast be honored when all the override instructions are evaluated and optimized. When you lock an override, SAS Visual Forecasting changes the final forecast to the override value and ensures that the reconciled forecasts add up to the locked override value. If the locked override value cannot be honored, an override conflict exists and must be resolved, as described in the section “Automatic Conflict Detection and Resolution.”

SAS Visual Forecasting provides the following types of override, which are preserved for the specified time periods:

- The *statistical forecast lock* preserves the statistical forecast for each individual child time series that is captured by a filter, ignoring any other override instructions that might affect the values for that time. For example, if you lock the statistical forecast for all products in one store during the month of June, any other overrides that might apply to the products sold in that store for June do not affect the final forecasts. However, the overrides for those products might be applied to any other stores where those products are sold for the month of June.
- An *aggregate final forecast lock* preserves the desired value for the final forecast aggregation of each individual time series that is captured by a filter. The final forecast values for each individual child time series are not locked, and their values are allowed to float. This means that the values for each individual time series can be changed by other override instructions that overlap any subset of these time series.

You might need to remove the lock on the override instruction if SAS Visual Forecasting cannot automatically resolve any override conflicts. You can remove a lock from the Override Conflicts dialog box or by editing the override from the **Override Management** view.

### Impact Analysis

Figure 14 shows an impact analysis report, which enables you to view information about overrides from other filters and view the impact of those filters on your current filter and time slice selection either before or after submitting an override. You can review the impact of previous overrides in history (for example, to inform a decision on an upcoming set of overrides). The report shows the total number of series in each filter and the number of time series that intersect with the current filter and time slice. The Delta column shows the difference between the statistical forecast and the final forecast. You can click any filter link to view the override specification in the **Overrides** view or click a transaction group to view the override in context in the **Override Management** view. If a lock has been placed on any of the filters, the lock type is displayed in the report, as shown in Figure 14.

Transaction	Filter	Status	Number of Series	Intersecting Series	Statistical Forecast	Previous Final Forecast	Override	Final Forecast	Delta	Lock Type
Group 2	Jennifer Economic	Applied	303	91	28,869,331.25	46,190,930.00	11,547,732.50	45,665,845.62	-25,084.38	Aggregated final forecast
Group 1	Jennifer	Applied	369	111	29,279,553.42	46,847,285.46	46,847,285.46	46,847,285.46	-	Aggregated final forecast

**Figure 14. Impact Analysis Report**

### Automatic Conflict Detection and Resolution

Whenever you apply an override instruction, SAS Visual Forecasting runs an optimization model to ensure feasibility of the solution given the set of applied overrides. When you submit an override or you resubmit previously applied overrides, SAS Visual Forecasting attempts to satisfy the override instructions with the other binding override instructions. If a conflict is found, you are notified and the conflict must be resolved immediately. No other action can be taken in the project until all conflicts are resolved and the optimization problem is feasible.

Conflicts can be caused by the following conditions:

- Two or more override instructions cannot be feasibly satisfied for the same time period.
- An override instruction causes the final forecast value to exceed the “Maximum percentage change allowed” as specified in the **Project Settings**.

- Any combination of constraints that are caused by locked statistical or aggregate values, maximum allowed changes, or overrides from filters that have overlapping filter specifications.

When a conflict is found, you are prompted to make a choice between **Resolve Automatically** (letting SAS Visual Forecasting resolve the conflict) or **Resolve Manually** (resolving it yourself), as shown in Figure 15.

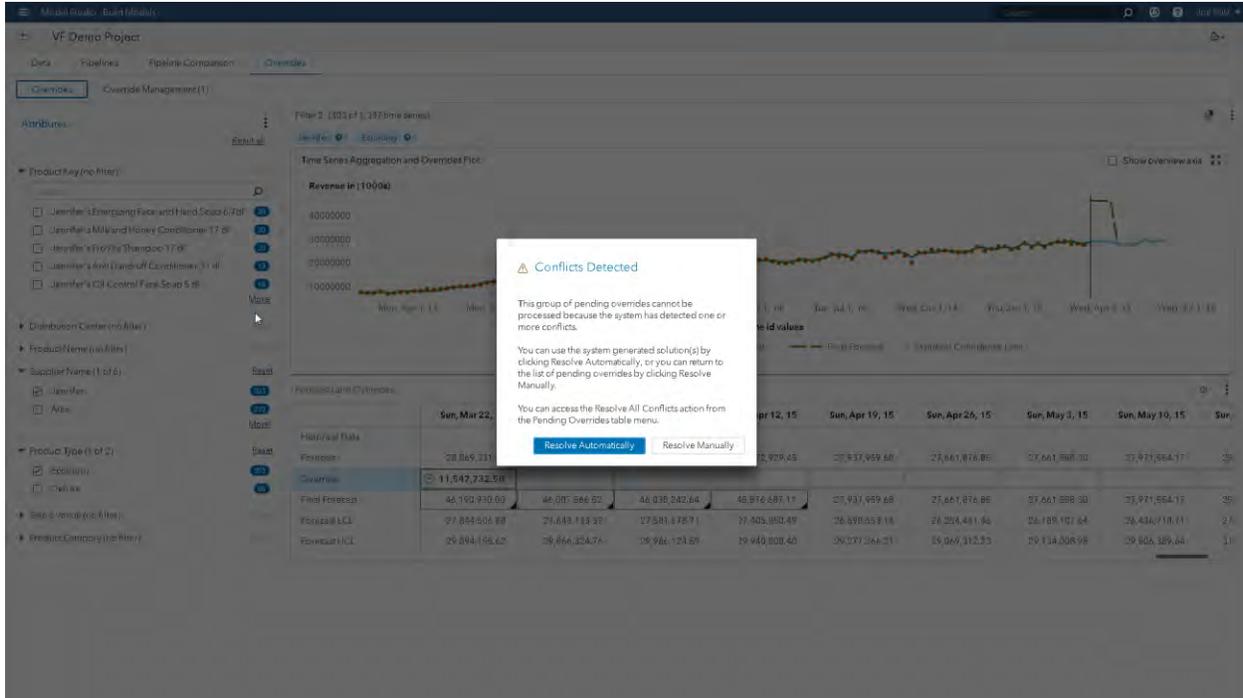


Figure 15. Override Conflict Detected

- If you click **Resolve Automatically**, SAS Visual Forecasting resolves the conflicts and updates the table in the **Override Management** view.

**Note:** Even if you do not choose to resolve conflicts automatically at this prompt, you can still choose later to resolve conflicts automatically from the **Override Management** view by clicking the three vertical dots icon above the upper right corner of the table and selecting **Automatically resolve all conflicts**.

- If you click **Resolve Manually**, the **Override Management** view appears and shows the override that has the conflict in the Status column. Each conflict in the table is labeled beginning with the word “Conflict” followed by a space and a number (such as “Conflict 1”), as shown in Figure 16.

Filter	Time Period	Status
<a href="#">Filter 2</a>	22Mar2015	<a href="#">Conflict 1</a>
<a href="#">Group B</a>	22Mar2015	Applied
<a href="#">Filter 1</a>	22Mar2015	Applied

Figure 16. Resolving Override Conflicts Manually

Click the link in the Status column for the conflict you want to resolve. The Manual Conflict Resolution dialog box appears and provides several options for resolving the conflict. Figure 17 shows a conflict between two filters that have already been applied with an aggregate final forecast lock.

Filter	Time Period	Override Status	Statistical Forecast	Override	Solution 1		Solution 2		Delete
					Optimized Final Forecast	Optimized Final Forecast	Optimized Final Forecast	Aggregated Final Forecast Lock	
Group B	Sun, Mar 2...	Applied	4,297,5...	2,477,0...	1	2,477,0...	3,517,5...		
Filter 2	Sun, Mar 2...	Pending	5,884,0...	10,414,...		7,861,5...	10,414,...		
Filter 1	Sun, Mar 2...	Applied	4,343,7...	6,515,6...	1	6,515,6...	8,178,0...		

**Figure 17. Manual Conflict Resolution Dialog Box**

The dialog box also provides one or two possible solutions, as shown in Figure 17.

- Solution 1 gives priority to locked overrides that have previously been applied. Choosing this solution replaces your pending override value with the best alternative that can be applied as determined by the optimization model.
- Solution 2 is available when you try to apply a locked override that cannot be resolved with a previously locked value. This solution gives priority to preserving the new pending locked override instruction and replaces one or more previously applied locked overrides as determined by the optimization model.

Another way you can choose to resolve conflicts is to remove a lock on an override instruction or delete an override altogether. In some cases, the Optimized Final Forecast value is the same for both solutions, so only Solution 1 is offered. After you make your selection and resolve the conflict, go to the **Override Management** view and resubmit the overrides.

## EXPORT FINAL FORECASTS AND OVERRIDES

You can export the forecasted data from two areas within the application. From the **Pipeline Comparison** tab, you can export any of the pipeline-forecasted data without overrides for further exploration, analysis, and reporting. From the **Overrides** tab, you can export data that contain both the baseline statistical forecasts and the calculated overrides values for further exploration, analysis, and reporting.

## DATA REFRESH ON FORECASTING CADENCE

All forecasting projects operate according to a specified cadence (forecasting interval). This cadence can be daily, weekly, monthly, and so on. The cadence is usually determined by the downstream application that consumes the forecast. These downstream applications can be for inventory management, scheduling, budgeting, and so on. In some cases, the forecast interval is driven by limitations within an organization that force the forecast interval to be at a lower frequency than the business would prefer (such as monthly instead of weekly).

Typically, new historical data is added for each time interval as time moves forward and those time periods pass from future to past. The addition of new data requires new forecasts to be created. The new forecasts can be based on the following:

- The existing forecast models with existing parameter values
- The existing forecast models with updated parameter values
- The full process of model generation, forecast model generation, automatic variable and event selection, automatic parameter optimization, automatic champion model selection, and forecast generation

When input data potentially change as time series are added or removed (or both), the filters are dynamically resolved. A certain filter might have contained 500 time series originally, but after the latest refresh of the data might now contain 510 time series (or 483 time series). Because the filters are dynamically resolved with data changes, the resulting forecast numbers for a particular filter are also dynamically resolved and updated. Regardless of the method used to update the baseline statistical forecasts, the override instructions are applied to create the updated final forecasts.

## SUPPORT FOR PRODUCTION

### BATCH CODE FOR RECURRENT EXECUTION

You can execute a SAS Visual Forecasting project by using batch code. An example of batch code is shown in Figure 18. Batch code enables you to easily run projects repeatedly.

You can download either the SAS or the Python version of the batch code for a project to a client machine. You can execute batch code from a SAS client, such as SAS Studio, or from a Python client. You can also schedule the batch code for execution according to project data update cadence. The batch code will execute in a “headless” way (that is, without any need to use the graphical user interface).

Another important aspect of support for production is portability of projects. You can download a project’s archive version to a client machine in a form of a ZIP file. Because the ZIP file does not contain data, it has a small footprint. You can easily upload it to another installation (for example, to a production system). Prior to migrating an archived version of a project to a new system, you need to load the data tables that are used by the project. Project data can be placed in any available library in the new environment. The new location of project data can be updated through project settings.

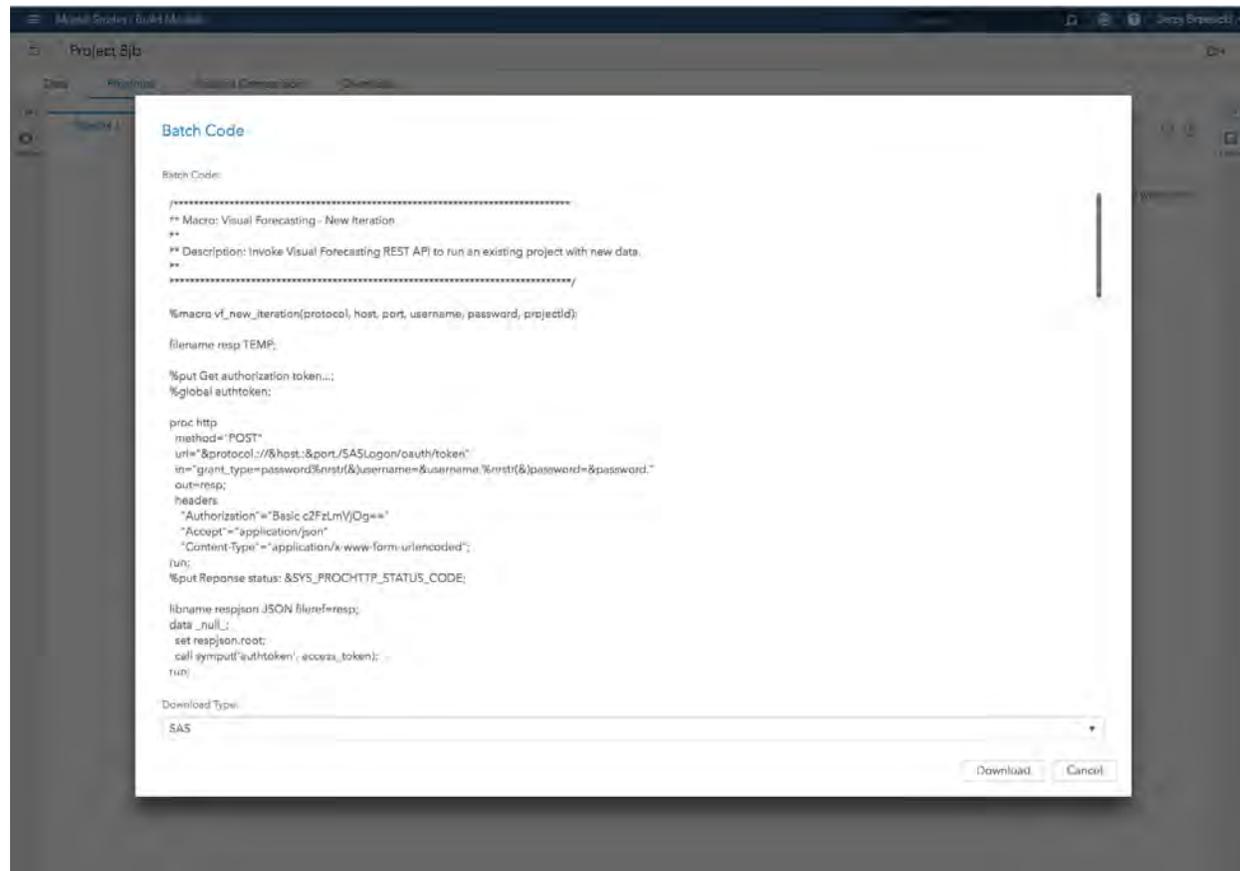


Figure 18. Batch Code for a SAS Visual Forecasting Project

## THE TOOLBOX COLLABORATIVE ENVIRONMENT

Collaboration is one of the cornerstones of the functionality and architecture of Model Studio. There is a dedicated area (as shown in Figure 19) called the Toolbox, where reusable components of Model Studio projects can be placed. Available in the Toolbox are reusable versions of pipeline templates and modeling strategy nodes. Both pipeline templates and nodes can be shared and used by different projects. You can specify which pipeline template is a default template for new projects. Because both pipeline templates and nodes in the Toolbox are portable, they can be downloaded to a client machine in the form of ZIP files and moved to another system.

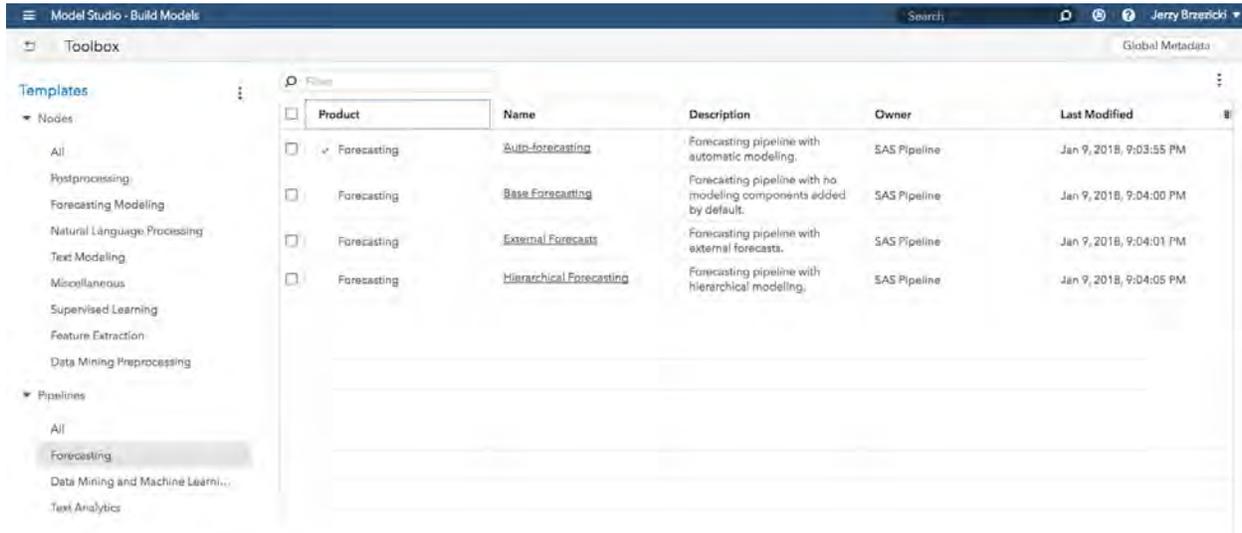


Figure 19. Model Studio Toolbox

## REST API

Model Studio architecture uses a REST API as a layer through which the HTML5 user interface communicates with the server by using series of HTTP requests such as GET, PUT, POST, and DELETE. All functionality of the Model Studio HTML5 client can be reproduced through series of REST API calls. The documentation for SAS Visual Forecasting REST APIs will be published and available in the future. Following is an example of a REST API call:

Request URL:

<http://visualforecasting.sas.com/analyticsGateway/projects>

Response (json):

```
items:[{creationTimeStamp: "2018-01-26T02:04:26.954Z", modifiedTimeStamp: "2018-01-26T02:43:51.323Z", ...}, ...]
```

## PROGRAMMING ENVIRONMENT FOR TIME SERIES ANALYSIS AND FORECASTING

SAS Visual Forecasting 8.2 includes full access to coding capabilities that are supported by the TSMODEL, TSRECONCILE, and TSINFO procedures.

The TSMODEL procedure is the central procedure; it provides a framework for processing time series data in a single pass through the data. It supports time series analysis and processing, model diagnostics and selection, and forecasting. PROC TSMODEL also supports execution of custom code that you provide. The syntax of the TSMODEL procedure is as follows:

```

PROC TSMODEL options ;
  BY variables ;
  ID variable INTERVAL=interval < options > ;
  OUTARRAYS array-name-list ;
  OUTSCALARS scalar-name-list ;
  INSCALARS scalar-name-list ;
  VAR variable-list / options ;
  REQUIRE package-list ;
  PRINT print-options ;
  SUBMIT < FILE= SAS-file-ref | 'File-path' > < submit-options > ;
  Programming statements ;
  ENDSUBMIT ;

```

All aspects of time series analysis and the forecasting modeling process are organized in a series of time series packages that are used with the TSMODEL procedure:

- ATSM (automatic time series analysis and forecasting)
- MTF (time series motif analysis)
- SFS (simple forecast service)
- SSA (singular spectrum analysis)
- TIMFIL (time filters)
- TFA (time frequency analysis)
- TSA (time series analysis)
- TSM (time series model)
- UTL (utility)

The TSINFO procedure evaluates a variable in an input data table as a time ID candidate; it is the next generation of the TIMEID procedure. The TSRECONCILE procedure reconciles forecasts at two different levels of hierarchy in a top-down fashion and supports multithreaded reconciliation for input data that are contained in CAS tables; it is the next generation of the HPFRECONCILE procedure. In contrast to earlier generations of these procedures, the TSMODEL, TSRECONCILE, and TSINFO procedures do not require pre-sorting of data by BY variables and ID variable.

For more information about SAS Visual Forecasting, see Trovero and Leonard (2018) and Quirino, Leonard, and Blair (2018).

## CONCLUSION

SAS Visual Forecasting, which runs on SAS Viya, is the latest forecasting offering from SAS. The user interface emphasizes a process workflow approach to forecast modeling and analysis that uses pipelines. SAS Visual Forecasting supports automatic large-scale and hierarchical forecasting but also enables you to use non-hierarchical modeling options. SAS Visual Forecasting is a collaborative tool that enables you to create custom modeling strategy nodes and share reusable components such as modeling strategy node templates, pipeline templates, and projects. You can also compare forecasting modeling strategies both within a pipeline and across multiple pipelines.

SAS Visual Forecasting contains a flexible override capability that is based on time series attributes, and it enables you to use faceted search filters to influence or modify the statistical forecasts. Because of this capability, overrides instructions are no longer limited to the variables that define the modeling hierarchy. This extremely powerful feature enables you to easily override forecasts by using a combination of attributes instead of having to override one series at a time.

The SAS Visual Forecasting ecosystem has the following properties:

- Scalable: takes advantage of the speed and elasticity of the SAS in-memory environment
- Consistent: provides a common HTML5 web-based user experience
- Open: uses REST API microservices that can be called from SAS, Python, R, Java, and Lua
- “Pluggable”: supports modeling nodes that are based on user-defined custom code
- Portable: enables all projects, pipelines, and nodes to be shared and reused

SAS Visual Forecasting projects are configured as a dynamic forecasting process. As new data are added to the project over time, the entire project, from pulling in the new data to executing the override instructions, can be run either from the user interface or via batch code by REST API calls. The override instructions for future periods will dynamically adjust the final forecasts in line with the forecasting cadence.

## REFERENCES

Quirino, Thiago, and Leonard, Michael, and Blair, Ed. 2018. “Scalable Cloud-Based Time Series Analysis and Forecasting.” *Proceedings of the SAS Global Forum 2018 Conference*. Cary, NC: SAS Institute Inc.

Trovero, Michele, and Leonard, Michael. 2018. “Time Series Feature Engineering.” *Proceedings of the SAS Global Forum 2018 Conference*. Cary, NC: SAS Institute Inc.

## ACKNOWLEDGMENTS

Special thanks to Michele Trovero for reviewing and providing suggestions for improvement and to Anne Baxter for her editing.

## CONTACT INFORMATION

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

Jerzy M. Brzezicki  
SAS Institute Inc.  
[jerzy.brzezicki@sas.com](mailto:jerzy.brzezicki@sas.com)

Joseph H. Katz  
SAS Institute Inc.  
919-531-3923  
[joe.katz@sas.com](mailto:joe.katz@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.