

## Introduction to SAS® Decision Manager

Stephen Sparano, SAS; Charlotte Crain, SAS;  
David Duling, SAS Institute Inc., Cary, NC

### ABSTRACT

Organizations today make numerous decisions within their businesses that affect almost every aspect of their daily operations. Many of these decisions are now automatically generated by sophisticated enterprise decision management systems. These decisions include what offers to make to customers, sales transaction processing, payment processing, call center interactions, industrial maintenance, transportation scheduling, and thousands of other applications that all have a significant impact on the business bottom line. Concurrently, many of these same companies have developed or are now developing new or additional analytical models that provide valuable insight into their customers, their products, and their markets. Unfortunately, many of the decision systems cannot maximize the power of analytics in the business processes at the point where the decisions are made.

SAS® Decision Manager is a new product that integrates analytical models with business rules and provides ways to deploy models and rules together within operational systems where the decisions are made. Analytically driven decisions can be monitored, assessed, and improved over time. This paper describes the new product and its use and shows how models and business rules can be joined into a decision process and deployed to either batch processes or near real-time web processes that can be consumed by business applications.

### INTRODUCTION TO SAS DECISION MANAGER

#### WHAT ARE DECISIONS?

Organizations today are facing a growing need to make decisions faster and also make better decisions for their shareholders and their customers. They are being asked to do this in an environment that includes challenges such as rapidly changing markets, growing demands to process larger amounts of data, and dwindling IT resources.

These companies need more efficient ways to quickly develop these strategies that provide tools and approaches to design and implement these decisions so that they are an integral part of their operational processes. So, companies need not only to make these decisions faster but also to make better decisions that meet the demands of their customers, satisfy the needs of their investors, grow market share, save on operational expenses, and deliver revenue to the top line to fuel further growth.

Most companies start out with the idea that they want to implement a business strategy driven by a market need, such as reduce the number of delinquent loans, improve the quality of customers that we approve for credit cards, improve the manner in which we interact with our customers by making better and more relevant offers. In order to do that, companies need to bring together various elements that typically constitute a decision, and the elements that go into a decision typically include the following:

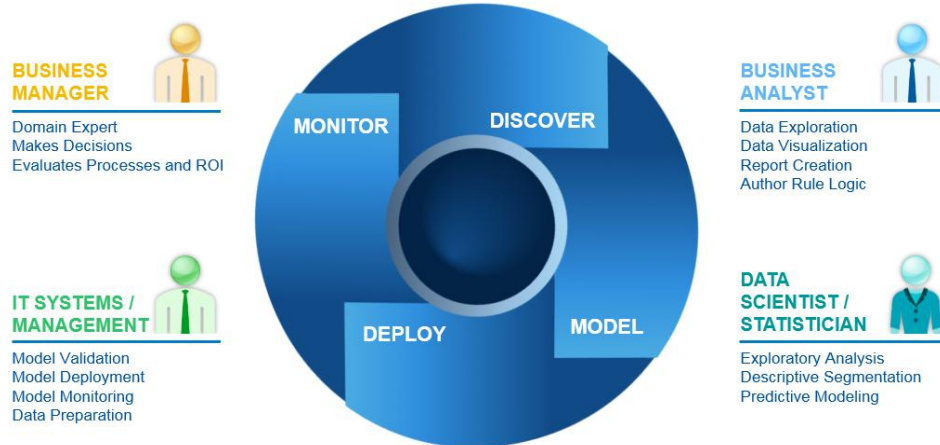
- Data Preparation: manage and prepare data to better understand what has happened in the past and use this information to formulate new strategies and implement better decision making
- Analytical Modeling: prepare and monitor analytical models to better understand why events and certain behaviors have occurred with customers or with business operations
- Business Rules: define and discover rules for what action to take based on future behaviors in order to react quickly to situations detected in the marketplace
- Optimization: leverage the information gained from past decisions to improve how decisions are made in the future
- Process Integration: deliver the decisions directly into the business process where the decisions are made to reduce any lag between when an event is detected and when action is taken

#### BENEFITS

##### Support for the Decision Lifecycle

There are many challenges to delivering a comprehensive and integrated decision making environment and often these challenges prevent a smooth flow of information across the different phases and roles within the Decision

Lifecycle, as shown in the figure below.



**Figure 1: Decision Lifecycle**

Within businesses today, the roles and activity phases are not as smoothly implemented as depicted in the graphic above. Numerous handoffs, overwhelmed and understaffed teams, and rapidly changing regulations and policies all contribute to a less than optimal decisioning process. Benefits of this approach can be summarized as follows:

- Reduce Siloes Separating IT and Business
  - o Allow Business Users to easily create or access models needed to support decision making strategies through model sharing and collaboration with the statisticians and data scientists
- Faster Decision Deployment Times
  - o Eliminates need for custom coding to generate business logic to simplify integration with analytical model output
  - o Simplify means for integration of models with business processes thereby reducing time to action
- Decision components treated and managed as corporate assets
  - o Provide governance and version control for rules and models
  - o Automate deployment steps for greater auditability and repeatability
- Analytics is closer to where decisions are made:
  - o Take action based on model output within business process
  - o Deployments are slow and not connected to process
- Business users, who know the business processes best, can design and change business rules
  - o React quickly to changing regulations and policies
  - o No programming required, all code is generated by system from rule definition

## DECISION MANAGEMENT: AN EXAMPLE

In order to best understand the benefits of SAS Decision Manager and how to obtain these benefits, this paper presents a scenario for usage of the technology that highlights its ease of use and important features. The scenario entails a large, national used vehicle reseller that has analyzed some of its past sales and returns data and has noticed an increase in the number of vehicles that have been returned to them by their customers for various reasons. After analyzing their data they have seen some relationships between the cars they've purchased at auctions around the country, various attributes of the vehicles, and the vehicles that have been returned by unhappy customers.

The returns by customers of the vehicles have resulted in some negative impacts on the company's business and they would like to understand how they can improve their decision making that would result in fewer cars being returned back to them and their lots. They decided to look into how they were making decisions to purchase vehicles at the start of their business process and felt that if they could improve how they made decisions about the vehicles

they purchased, the result would be fewer vehicles returned and the results would yield higher profits through reduced inventory costs, lower repair costs, lower reimbursement costs, and overall happier (and possibly repeat) customers.

So, the company decided to embark on the following efforts:

- review their data to understand the attributes of their vehicles including which vehicles were returned
- develop predictive models that could help determine the probability a vehicle will be returned
- augment the model outcomes with business rules driven by corporate policies and guidelines
- deploy these vehicle purchasing decisions to business processes to enable buyers to make better decisions at time of purchase

Once these decisions were defined, they could deploy them easily to their buyers for better results and improvements in their business profits and customer satisfaction.

## SAS DECISION MANAGER: FEATURE OVERVIEW

SAS Decision Manager supports collaboration between business analysts, statisticians, IT staff, and business managers by unifying the important aspects of decision management. These components support the decision lifecycle and are supported in the following categories within the user interface:

- **Data:** Provides support for adding new data tables, define data tables in SAS libraries, and define libraries in SAS Metadata
- **Business Rules:** Provides support for authoring, testing, and publishing business rules
- **My Tasks:** Provides support for automated workflow within both SAS business rules and SAS model projects to automate the lifecycle tasks, deliver notifications, and provide tracking and auditability for increase governance
- **Models:** Provides support for managing analytical models including importing new models using shared SAS metadata and local files, automated retraining of models, monitoring model performance, and publishing models for integration with business processes
- **Workflows:** Workflows that can be initiated in support of activities such as reviews and approvals to support repeatability and auditability.

The categories are surfaced within the user interface on the left-hand side and are shown in the figure below.

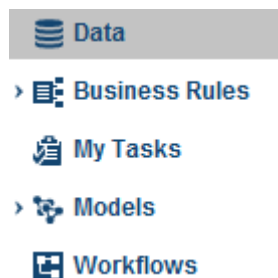


Figure 2: SAS Decision Manager Categories

## DATA PREPARATION

As with most business processes and decisions, the process starts with gaining a strong understanding of your data. In order to determine what decisions they needed to improve, the company started by understanding what attributes were available for their past sales and for the vehicles that were returned to them. The data preparation capabilities in SAS Decision Manager provide for the following:

- Simple Data Access: Users can select tables available from SAS libraries in SAS metadata to minimize reliance on IT resources and allow the users to review their auto data sales.
- Data Summarization: Auto sales data can be summarized to show counts for vehicle type, color, size, transmission types, and sales price.
- Base Table Creation: Using query building, column selection, and integration with SAS® Visual Analytics,

users have ability to create tables that can combine other vehicle sales data along with geographic data about where these vehicles were purchased.

- **Data Table and Library Management:** The simple interface allows business users to locate tables, and create libraries for their project

The data being used contains information about the vehicles and their prices, as well as some statistical data. A summary of the table metadata is shown in Figure 3: Vehicle Table Metadata, below:

Data source name:	AUTO_RULE_TEST	Table name:	AUTO_RULE_TEST
Description:		Type:	Table
		Date created:	Feb 2, 2014 01:43 PM
Library:	Auto	Columns:	38
Engine:	BASE	Rows:	
Location:	/Shared Data/HLS/AUTO_RULE_TEST		
<b>Metadata Columns</b>			
Name	Type	Format	Length
nationality	CHAR		14
size	CHAR		11
vehbcost	NUM		8
vehicleage	NUM		8
vehodo	NUM		8
wheeltype	CHAR		7
refid	NUM		8
purchase	NUM	MMDDYY10.	8
auclon	CHAR		7
vehyear	NUM		8
make	CHAR		12
model	CHAR		20
trim	CHAR		3
submodel	CHAR		32
color	CHAR		9
transmission	CHAR		6
wheeltypeid	CHAR		4
topthreeamericanname	CHAR		8
mmracquisitionauclonaverageprice	NUM		8
mmracquisitionaucloncleanprice	NUM		8
mmracquisitionretailaverageprice	NUM		8
mmracquisitionretailcleanprice	NUM		8
mmrcurrentauclonaverageprice	NUM		8
mmrcurrentaucloncleanprice	NUM		8
mmrcurrentretailaverageprice	NUM		8
mmrcurrentretailcleanprice	NUM		8
amount	CHAR		1

**Figure 3: Vehicle Table Metadata**

## Data Summarization and Viewing

Data summarizations, as seen in Figure 4: Data Summarization Example below, provide additional statistical summarizations of data that can be useful to the decision author by providing greater insights into the data. Numeric and character summarizations provide information about maximum, minimum, mean, average, percentiles, skewness, kurtosis, and missing values within the data.

Because the data is always changing, business users can create data summarizations as their data changes over time thereby allowing them to visualize these changes and identify patterns in the data that can be useful in their use of models or during business rule authoring.

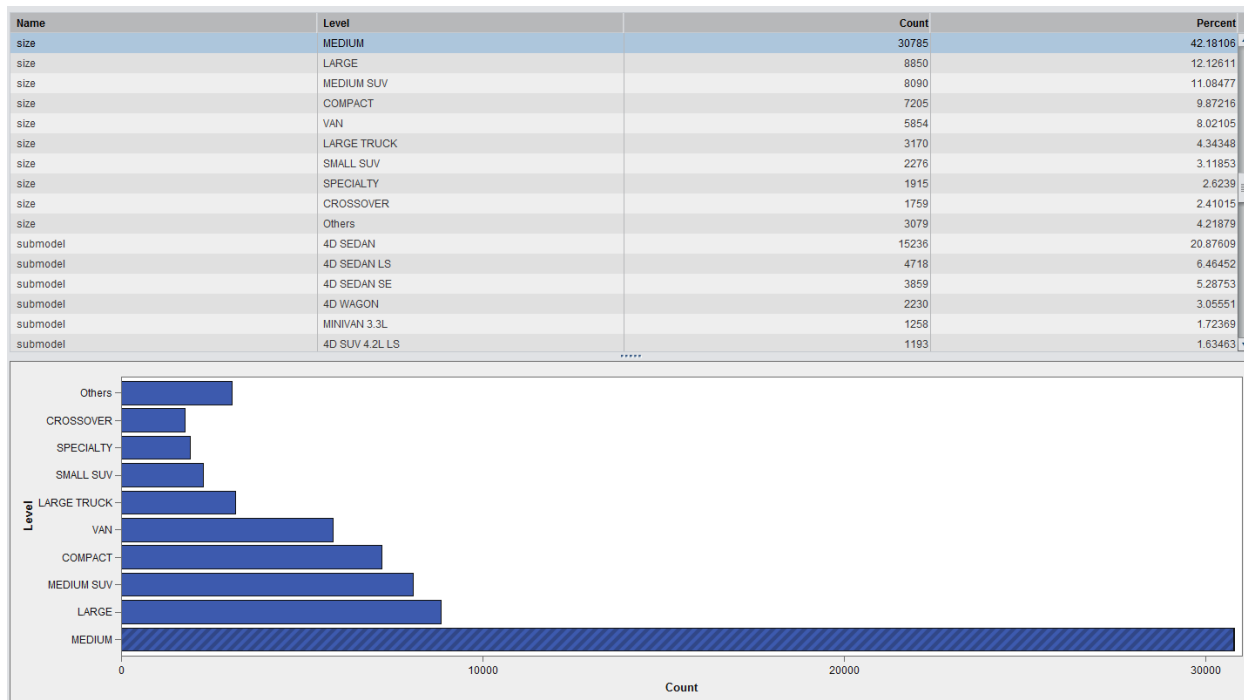


Figure 4: Data Summarization Example

After analyzing the vehicle data and preparing their base tables, the business users would like to understand how they can predict which vehicles will have a high probability of being returned to them. The business users felt that by collaborating with the statisticians, together they can use this data to build analytical models that can improve their ability to predict what vehicles are not good bets--that is, which vehicles have a high probability of being returned, or are bad cars.

## MODEL MANAGER

### PREDICTIVE MODELS IN DECISION MAKING

Predictive modeling is the most essential analytical step of the automated decision making process. Predictive models are often more accurate than expert systems and can detect patterns that human recognition will miss. Models are trained on historical data that contains both input variables and target variables. Inputs are conditions that can be used to make predictions and targets are measurements that need to be predicted.

In this case, the input variables represent the information that was available to the purchasing dealership at the time of the used car auction. The data has been labeled with a new variable named *isBadBuy* that indicates that the car was later returned by the customer to the dealership as a lemon – a bad car. In the summary plots, we can see that *isBadBuy* is *TRUE* about 12% of the time. That is a lot of lemons that the auto dealerships want to avoid. Therefore it is important to build a model that can predict which cars will be returned to the dealership as bad buys. We will use that prediction along with other information about strategy and policies in our decision making process.

There are two actions needed for deploying a predictive model into a decision making process.

- 1) **Model Training.** This is the act of running a statistical or data mining procedure that will build a model to predict *isBadBuy* based on a set of input variables. Most users employ either SAS/STAT® or SAS® Enterprise Miner™ for this purpose. The final result is a model scoring expression such as *model\_score\_code.sas*, which is typically represented by tens to thousands of lines of SAS code. For each row of data, the model will output a probability that *isBadBuy* is true.
- 2) **Model Scoring.** This is the act of running the model scoring code with new data to produce new scores. We have trained the model using historical data that has already been labeled with *isBadBuy*. The model scoring process will add a prediction of *isBadBuy* to each record of data that is scored. For SAS programmers, we may conceptually represent this as a simple DATA step.

```
data newscores; set newdata; %include "model_score_code.sas"; run ;
```

## BUILDING PREDICTIVE MODELS

SAS provides several tools for building good statistical models. SAS/STAT and SAS Enterprise Miner can be used to construct accurate and robust statistical and machine learning models. The word 'accurate' indicates models that successfully detect patterns in the training data to produce the minimum error rate in their predictions. The word 'robust' is an indicator that the model's predictions will be reliable on data that is similar to the training data. A developer may create several models with different attributes as candidates for a new decision making project. In this case we have created two candidate models using SAS Enterprise Miner and have saved each model as a Model Package File. This file includes all possible details about training and scoring this model. SAS Decision Manager can import these models for use in a Decision Making process.

## MANAGING MODELS

As predictive models are an important asset to the business, they often need to be carefully managed. Models should be used only in situations that match the period in which the training data was collected. There are frequently several candidate models that have different performance attributes. Some models perform better or work for different customer segments, in different time periods, or in different economic conditions. Selecting the champion and challenger models from the set of candidate models is often a process governed by internal and external regulatory concerns, and can have a large impact on business profit.

SAS *Decision Manager* includes all the functionality of SAS® *Model Manager*, and is the preferred software for managing collections of statistical and data mining models. It can import models from SAS/STAT, SAS *Enterprise Miner*, the PMML (Predictive Model Markup Language) standard, and even some models created in the R language. SAS *Model Manager* provides functions for testing the model scoring functions, for computing performance metrics that can be used to select models, and for monitoring model performance in production systems within a framework that provides automation, workflow planning and tracking, and model governance using versions, comments, and attachments.

We have imported both *isBadBuy* models created with SAS *Enterprise Miner* into our Model Project named *AuctionPurchaseBadCar*. We can now compare the models side by side, where we immediately recognize that the Decision Tree based model uses much fewer inputs than the Logistic based model. Since our project will be deployed in a real-time system, we will prefer the model with fewer input variables as that will make our real-time data payload smaller and simplify the data needs for our deployment. Therefore, we have marked the Tree model as the champion model and will use that in our decisioning process.

AuctionPurchaseBadCar: Tree_ShallKicked					
Model Properties Attachments Comments					
Variables					
Input					
Name	Type	Measure	Length	Description	
auction	C	NOMINAL	7	Auction	
color	C	NOMINAL	9	Color	
vehicleage	N	INTERVAL	8	VehicleAge	
wheeltypeid	C	NOMINAL	4	WheelTypeID	
transmission	C	NOMINAL	6	Transmission	
make	C	NOMINAL	12	Make	



  

AuctionPurchaseBadCar: Logistic_AutoKick					
Model Properties Attachments Comments					
Variables					
Input					
Name	Type	Measure	Length	Description	
auction	C	NOMINAL	7	Auction	
aucguart	C	NOMINAL	5	AUCGUART	
warrantycost	N	INTERVAL	8	WarrantyCost	
mmrcurren...	N	INTERVAL	8	MMRCurrentRetailA...	
isonlinesale	N	BINARY	8	IsOnlineSale	
mmracqui...	N	INTERVAL	8	MMRAcquisitionAuct...	
mmracqui...	N	INTERVAL	8	MMRAcquisitionReta...	
transmissi...	C	NOMINAL	6	Transmission	
mmrcurren...	N	INTERVAL	8	MMRCurrentAuction...	
mmracqui...	N	INTERVAL	8	MMRAcquisitionRet...	
vehodo	N	INTERVAL	8	VehOdo	
size	C	NOMINAL	11	Size	
mmracqui...	N	INTERVAL	8	MMRAcquisitionAuct...	

Figure 5. The Model Project Showing Both the Tree and Logistic Models

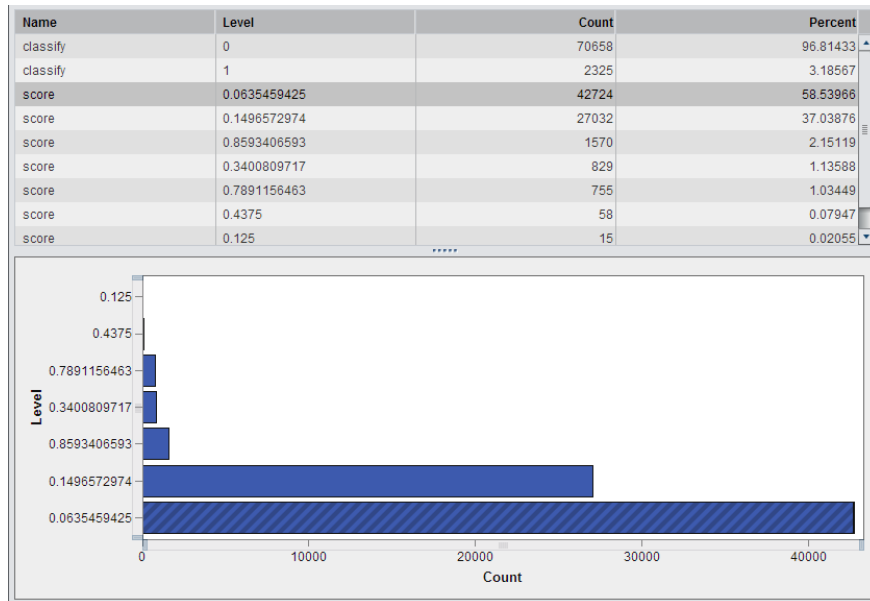
## MODEL TESTING

Once we have selected the model, we should test the scoring function using new data to validate how the model performs outside the statistical developer environment. The model project provides score testing functions. We can set up multiple tests with different data sets representing conditions that may be found in the production environment.

Tests	Results	Schedule	Job History			
Status	Name	Model Name	Date Created	Date Completed	Created By	
	TreeTest_prod1	<Tree_ShakKicked>	Feb 20, 2014 01:13 PM	Feb 20, 2014 01:14 PM	edmddev	
	StevePMAccTestMM	<Logistic_AutoKick>	Jan 22, 2014 04:46 PM		Steve	

**Figure 6. Multiple Test Cases Can Be Run Interactively or Scheduled**

Once the tests have run, we can examine the scores as seen in the view below, using the data summary function found in the data tables category. For this project we have tested both the logistic and the tree models and found them both acceptable, and so we can continue to use the Tree model as the champion.



**Figure 7. Distributions of Model Scores**

## MODEL PUBLISHING

Once we are satisfied with the choice of the Tree model, we will publish the model to the SAS enterprise so that it may be used in a new decision making process. Once it is published, there are several tools that will consume the model in a business analysis or decision making process. Refer to the matrix of possibilities in the deployment section. In our example, we will use import this model into SAS® Real-Time Decision Manager for deploying the final decision making process.

## MODEL MONITORING

Once the model is employed in a production decision process, the business will begin to accumulate results. These results will need to be monitored to make sure the model is still producing robust predictions. There is typically a time lag between model deployment and model monitoring. In our case, the auto decision process may be deployed in January and the cars purchased and resold in January, February, and March. During that time, we may monitor the scoring processes for "drift", otherwise known as change in the distribution of either the source data attributes or the scores that have been generated. For example, we may find that new cars have been introduced that have different characteristics than the cars we originally modeled. One line of cars may have been found to be less reliable than previously measured, as compared to other models. This may lead to change in the distribution of scores that will affect the decision process. For example, the original data may have contained 12% bad buys, while the current data is predicting that 50% of cars will be bad buys. Such a change will lead us to consider rebuilding the models to adapt to the new conditions. SAS Decision Manager includes processes that will monitor the health of models during their production processes and alert users when unfavorable conditions exist. This paper is focused on building and deploying a model decisioning process; for more information about monitoring models and decisions refer to SAS documentation and training.

## MODELS AND DECISIONS

We are now ready to move to the next phase of our project. We have a model that will predict the probability that a car will be returned. However, the statistical model is only one part of the decision making process. The model was built on historical data representing conditions that were present when the data was collected. Conditions may have already changed since that time. In addition, we have changes in business policy that tell us to prefer one type of car over another such as evolving customer preferences and available inventory. We need a way to build *Business Rules* to produce a valid decision process.

## BUSINESS RULES MANAGER

Once the business users have the output of the analytical models developed by the statisticians, they can apply business policies to the model output to further fine tune the predictive nature of the models. The business users see that business rules are a way to extend the decision making power of the models without having to retrain the models or adjust the models that would require the support of the statisticians. The business rules can be modified and adjusted easily and adapt to changing business requirements.

## DEFINING AND TESTING THE BUSINESS RULES

Business rules represent the business logic that drives decisions in companies. These business rules are often found in government policies, regulations, tax laws, financial guidelines, and federal and local laws. These rules can change frequently (for example, in marketing and customer relationship applications), and rules need to be modified, tested, and deployed quickly. Each of these updates needs to be tracked for reporting and auditing purposes.

Our vehicle dealer application requires that their business rules override the output of the models described earlier and that they determine the action to take based on the defined business rules that evaluate to true. Based on the model output, the business would like to evaluate the probability of the vehicle being a bad purchase in conjunction with other business policies that upper management would like to implement related to a few of the vehicle's attributes, including the following:

- Vehicle cost: avoid purchasing vehicles that are above a price point.
- Color: avoid vehicles of a specific color.
- Make: some makes are not desirable due to a change in lack of parts availability.
- Wheel Type: avoid vehicles that have a specific type of wheel.
- Vehicle VIN State: consider the state where the vehicle was purchased.
- Probability: probability of vehicle being a bad purchase decision. This is available as output from the model.

These attributes are being considered as a way to override the probability that a vehicle is a bad vehicle. So, even if the model says that the vehicle is a bad purchase, management would like to override the model and reset the bad car classification to a good car classification.

## THE BUSINESS VOCABULARY

Business vocabularies allow business users to define the terms or variables used in the creation of business rule logic. The vocabularies consist of two primary components, entities and terms. Entities are the higher order component and each entity will have one or more terms associated with it and, collectively, a set of entities and terms represents a vocabulary.

Vocabularies can be large and contain many terms and therefore there need to be efficient ways to create and manage them. Users have the following approaches that support the creation of a vocabulary:

- User entered: manual method to define vocabularies, terms, and entities.
- Import and Export Macros: macros that support importing and exporting vocabularies, terms, and entities using a predefined data format based on a .CSV (comma-separated values) format.
- Data-driven Creation: automated method to define vocabularies, terms, and entities based on existing SAS data sets.

Based on the vehicle input data, the business users have developed a vocabulary of automobile terms that will be used in the business rule conditions and actions. But rather than manually enter these terms and entities, an automated approach has been taken to speed up the process. This approach allows the business user to take advantage of the data elements contained within the same data set that will be used for testing their rules, to generate the vocabulary used to author the rules. Figure 8: Importing Vocabularies from Data Sets, is an example of



The screenshot displays the 'AUTO\_RULE\_TEST' application interface. The top section shows a table of test results with columns like nationality, size, vehcost, vehicleage, wheeltype, refid, purchdate, auction, vehyear, make, model, and i. Below this is a 'Business Rules: Vocabularies' section with a search bar and a list of rules. The bottom section shows a 'Column Information' panel on the left and a 'Value' and 'Frequency' table on the right. A green arrow points from the 'Value' and 'Frequency' table to the 'Column Information' panel.

## RULES, RULE SETS, AND RULE FLOWS

Using rule sets as a way to logically group associated rules together, the business user can create multiple rule sets, each with its own objective. The rules are defined using the decision table approach, which allows business users to easily develop business rules using a table-like approach to define both conditions and actions.

- General vehicle qualification rules
  - o Rules included that evaluate cost, probability, vehicle age, and odometer readings
- Evaluate the vehicle's VIN state
- Evaluate the vehicle's manufacturer

A business policy has defined the cut off for any bad car to be a 70% probability by the model that predicts this value. This probability value represents the predicted likelihood that a vehicle will be returned to us by a customer. In other words, if the model predicts the probability of a vehicle being returned is greater than 70%, then the vehicle is considered a bad buy and it should be avoided at auction in the first place. Conversely, if the vehicle has a calculated probability of less than 70% then the vehicle is a good car and represents a risk that management has deemed acceptable.

9

## BUILDING BUSINESS RULE SETS

The following rules are created for the general vehicle qualification rules that will override a probability of greater than 70%:

- **IF** Make is Chevrolet AND Odometer is Less Than 65,000 miles AND Cost is Less Than \$4,000 AND Vehicle is Less Than or Equal to 6 years old **THEN** Good Car.
- **IF** Make is Nissan or Toyota AND Vehicle is Less Than or Equal to 8 years old **THEN** Good Car.

These rules are entered into the decision table as shown below in Figure 9: Vehicle Qualification Rules:

Condition Term						Action Term
#	eventProbability	make	vehicleage	vehbcost	vehodo	BadCarDecision
1	If	> .7				1
2	Else	<= .7				0
3	If	> .7	'CHEVROLET'	<= 6	< 4000	0
4	If	> .7	'NISSAN' OR make = 'TOYOTA'	<= 8	< 65000	0
Click here to add a new rule.						

**Rule Details**

Rule name: NissanException  
Record rule-fired data: ☒  
Name: AutoQualification  
Rule folder name: Auto  
Rule set identifier: 10913  
Rule order: 4  
Description: Override the model probability if NISSAN or TOYOTA, not too old, cost, and mileage arent concerns. These are good risks for our inventory.  
Rule expression: Condition: (eventProbability > .7) AND (make = 'NISSAN' or make = 'TOYOTA') AND (vehicleage <= 8)  
Action: BadCarDecision = 0

**Figure 9: Vehicle Qualification Rules**

In addition to the rule set for general qualification, the business also wants to implement policies using business rules based on the state the vehicle VIN is registered to. These rules include the following:

- **IF** VIN State is one of the states that management has deemed as high risk **THEN** Bad Car.

These rules use the LIKE function that allows input values to be verified against a list of valid values and return a true or false result. In the example here, Figure 10: Invalid States in LIKE Function, a list of invalid states that management does not want to purchase vehicles from is used as the reference list, and if the state matches the value in the input variable, then it will be noted as a bad car.

Condition Term		Action Term
#	vnst	BadCarDecision
1	If	= 1
	LIKE ('VA','CT','NJ','MD','GA','FL')	
Click here to add a new rule.		

**Rule Details**

Rule name: BadVINStates  
Record rule-fired data: ☒  
Name: VehicleStateRules  
Rule folder name: Auto  
Rule set identifier: 10944  
Rule order: 1  
Description: Rule to identify vehicles from states we want to avoid for policy reasons.  
Rule expression: Condition: (vnst LIKE ( 'VA' , 'CT' , 'NJ' , 'MD' , 'GA' , 'FL' ))  
Action: BadCarDecision = 1

**Figure 10: Invalid States in LIKE Function**

Lastly, the management team has determined that costs associated with parts for vehicle makes that are no longer manufactured or simply too expensive should be avoided as well. Therefore, additional rules are designed to address:

- If Make is from list of Makes no longer manufactured,  
THEN Bad Car.

These rules, as shown in Figure 11: Manufacturer Rules, enforce a policy that vehicles that are from a certain manufacturer will be deemed bad buys.

The screenshot displays a rule configuration interface. The top section shows a list of rules under the heading 'Condition Term' and 'Action Term'. The 'Condition Term' column lists various car makes, and the 'Action Term' column shows the action 'BadCarDecision' with a value of '1'. Below this, the 'Rule Details' section provides information for a specific rule named 'Saturn'.

#	Condition Term	Action Term
1	If "SATURN"	1
2	If "HUMMER"	1
3	If "OLDSMOBILE"	1
4	If "PONTIAC"	1
5	If "PLYMOUTH"	1
6	If "TESLA"	1
7	If "BENTLEY"	1
8	If "ROLLS ROYCE"	1
9	If "PEUGOT"	1
10	If "RENAULT"	1

**Rule Details**

Rule name: Saturn

Record rule-fired data: ☐

Name: BadBuyAutoMakes

Rule folder name: Auto

Rule set identifier: 10967

Rule order: 1

Description: No Longer Manufactured

Rule expression: Condition: (make = "SATURN")  
Action: BadCarDecision = 1

**Figure 11: Manufacturer Rules**

## CREATING BUSINESS RULE FLOWS

The business rule sets have been created based on their objectives and can be organized into a sequential flow. The execution of the rule sets is defined within a rule flow that determines the order in which the rule sets will be performed. The rules need to be executed in a particular order and the rule flow below shows them ordered based on the following needs:

- Eliminate Vehicles from Bad States
- Eliminate Invalid Vehicle Makes
- Apply Rules based on Model predictive values

#	<input checked="" type="checkbox"/> Run	Name	Version	Description	Date Modified
1	<input checked="" type="checkbox"/>	VehicleStateRules	Latest	Rules to validate the VIN state of the v...	Feb 21, 2014 11:24 AM
2	<input checked="" type="checkbox"/>	BadBuyAutoMakes	Latest	Do Not Buy Auto Make List	Feb 21, 2014 11:02 AM
3	<input checked="" type="checkbox"/>	AutoQualification	Latest		Feb 21, 2014 11:02 AM

AutoQualification : Version 1.3 rules logic (4 rules)			
#	Rule Name	Condition	Action
1	BadCar	If (eventProbability > .7)	then BadCarDecision = 1
2	ElseGood	Else if (eventProbability <= .7)	then BadCarDecision = 0
3	ChevyException	If (eventProbability > .7) AND (m...	then BadCarDecision = 0
4	NissanException	If (eventProbability > .7) AND (ma...	then BadCarDecision = 0

**Figure 12: Ordering Rule Sets in Rule Flows**

## DATA-DRIVEN RULE FLOW DISCOVERY

Data Driven Rule Discovery uses Decision Trees to derive business rule flows in an alternative way. This technique allows a user to automatically generate business rules from a representative data set of historical data as an alternative to the business user manually entering these rules, thereby saving time and effort. The technique is based on a proven analytical approach to define one or more business rule sets and business rule flows. Once derived using the simple, easy to use, wizard-based approach, the business user can inspect the rule sets and rule flows.

An important feature of the data-driven business rule discovery approach is the automated clustering techniques used to cluster similar rules together into rule sets. This automated rule clustering supports clustering similar rules together into rule sets that can be managed, ordered, and deployed as rule flows to meet different business needs.

The decision table allows users to view the rules in a simple and easy to understand format for editing of the business rule sets and the rule logic. This approach allows business users, who are not programmers, to easily read, understand, and modify the derived business rules to meet their business process needs.

## TESTING RULES FOR VALIDITY

Testing to ensure the rules produces the desired output before publishing the business rule flows to be consumed within a business process. Testing allows the business user to evaluate the business logic in the rule flow using input data and run those tests to evaluate the output. Once users have run the rule flow tests, the results can be verified by viewing the output tables.

### Rule Fired Analysis

Assessing the rules that were fired helps understand why results were produced and to understand if the correct rules are being applied. The business user can open the rules fired view, select an output record, and display the rules that lead to the result as shown below in Figure 13: Rule Fired Analysis.

Filter: None							
_recordSequenc...	BadCarDecision	eventProbability	make	vehbcost	vehicleage	vehodo	vnst
42	0	0.7891156463	NISSAN	5200	5	86889	FL
43	0	0.1496572974	CHRYSLER	8500	6	68990	FL
44	0	0.0635459425	CHEVROLET	7900	2	80949	FL
45	0	0.1496572974	DODGE	11335	6	52774	FL
46	0	0.8593406593	NISSAN	7100	6	72191	FL
47	0	0.1496572974	FORD	4600	7	59858	FL
48	0	0.7891156463	TOYOTA	3900	8	79576	FL
49	0	0.1496572974	FORD	4100	6	73291	FL
50	1	0.8593406593	PONTIAC	7500	4	50227	FL
51	0	0.1496572974	DODGE	9400	7	82146	FL
52	0	0.0635459425	SATURN	5600	4	58024	FL
53	0	0.0635459425	CHEVROLET	7700	4	40919	FL
54	0	0.1496572974	DODGE	4600	5	87643	FL
55	0	0.1496572974	FORD	4600	5	80968	FL
56	0	0.0635459425	DODGE	9635	4	50308	FL
57	0	0.0635459425	FORD	7700	3	80795	FL
58	0	0.0635459425	DODGE	9135	4	62239	FL
59	0	0.1496572974	CHRYSLER	6900	6	87008	FL
60	0	0.0635459425	HYUNDAI	7535	4	64060	FL

Rows: 72983 Filtered rows: 72983 Columns: 8 Selected row: 46

Rules Fired

#	Rule Set	Version	Rule Name	Rule Expression
1	AutoQualification	Latest	BadCar	If (eventProbability > .7), then BadCarDecision...
2	AutoQualification	Latest	NissanException	If (eventProbability > .7) AND (make = NISSA...
3	VehicleStateRules	Latest	BadVINStates	If (vnst LIKE ('VA', 'CT', 'NJ', 'MD', 'GA', 'FL...

Figure 13: Rule Fired Analysis

## PUBLISHING RULE FLOWS

Once the results have been verified and they are satisfactory, the business analyst can publish a version of the rule flow for consumption by various deployment engines. To publish, the versions of rule sets used in the rule flow should be locked to prevent any changes to the logic once deployed. By locking the rule set version, the business can ensure that the published versions are immutable until a new version of the rule flow is published. Figure 14: Rule Flow with Locked Rule Set Versions, shows the selected versions of rule sets that have been locked and are now included in the version of the rule flow to be sent for publishing.

#	<input checked="" type="checkbox"/> Run	Name	Version	Description	Date Modified
1	<input checked="" type="checkbox"/>	VehicleStateRules	1.1	Rules to validate the VIN state of the v...	Feb 21, 2014 11:24 AM
2	<input checked="" type="checkbox"/>	BadBuyAutoMakes	1.1	Do Not Buy Auto Make List	Feb 21, 2014 11:02 AM
3	<input checked="" type="checkbox"/>	AutoQualification	1.3		Feb 21, 2014 11:02 AM

Figure 14: Rule Flow with Locked Rule Set Versions

The next step in the process is to put these deployed rules and models to work in a business process that is close to where these decisions are made. These decisions can be used by the vehicle purchasing agents using applications that utilize web services to call the decision engine to render decisions or as part of a nightly batch process that renders decisions on incoming transaction files. These are just two examples of the ways these decisions can be employed in the business operations.

## INTEGRATING THE DECISION COMPONENTS

With SAS Decision Manager, the analytical models and business rules can be deployed individually or in combination as part of a decision flow. A matrix here summarizes deployment targets that are supported out of the box. Additional targets may be configured manually. Check the product documentation or call SAS Technical Support for more information.

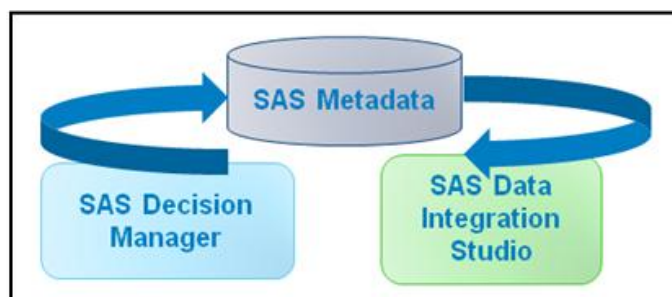
<i>Deployment Target</i>	<i>Decision Manager Objects</i>		
	<b>Rule Flow</b>	<b>Model</b>	<b>Decision Flow (Rules and Models)</b>
<i>Web Services</i>			✓
<i>In Database</i>		✓	
<i>SAS Stored Process</i>	✓		
<i>SAS Batch Processing</i>	✓	✓	✓

This paper will describe two common approaches that can be used to deploy models and rules together to be integrated into business processes easily and quickly. Each approach highlights how this can be done without custom coding and how integration is supported with SAS Decision Manager and other SAS tools.

## TWO COMMON APPROACHES FOR INTEGRATING MODELS AND RULES

There are two common approaches that can be used to formulate a logical flow that brings together essential decision components such as data, models, and business rules. These two approaches utilize either 1) a process flow within SAS® Data Integration Studio or 2) the Flow Builder within SAS® Decision Management.

SAS metadata ties the logic together (see Figure 1) for straightforward integration into production environments.



**Figure 1: SAS Decision Manager Uses Metadata for Straightforward Integration**

## Business Example: Rules and Models Combined for Better Decisions

The business problem addressed in screenshots for this section relates to the context of making a decision whether to buy a particular used car at an automobile auction sale. If the car is determined to be a good buy, then what is an expected purchase price? The models and business rules together provide answers whether a used car is a good buy and, if it is a good buy, then what is an expected purchase price? The business rules enhance the decisioning process because there can be overriding reasons why a car may be a good buy or not that are outside of what could be appropriately modeled or that new information is known about a particular car market. For example, a particular state, year, and make of the car combination could be handled by business rules.

## SAS® DATA INTEGRATION STUDIO – COMBINE MODELS AND RULES

### Bringing data, models, and rules together using SAS® Data Integration Studio

Create a process flow to integrate data, models, and business rules using SAS Data Integration Studio as seen in Table 1 below. In this example, start with data on left hand side, add a model to determine whether the car is worth buying or not. Run qualification rules that use additional logic to enhance the decision and then add a price model. Final output is a table of decisions of whether it is desirable to buy a particular used car and the expected purchase price. There are other combinations of logic that could also be utilized in this same scenario.

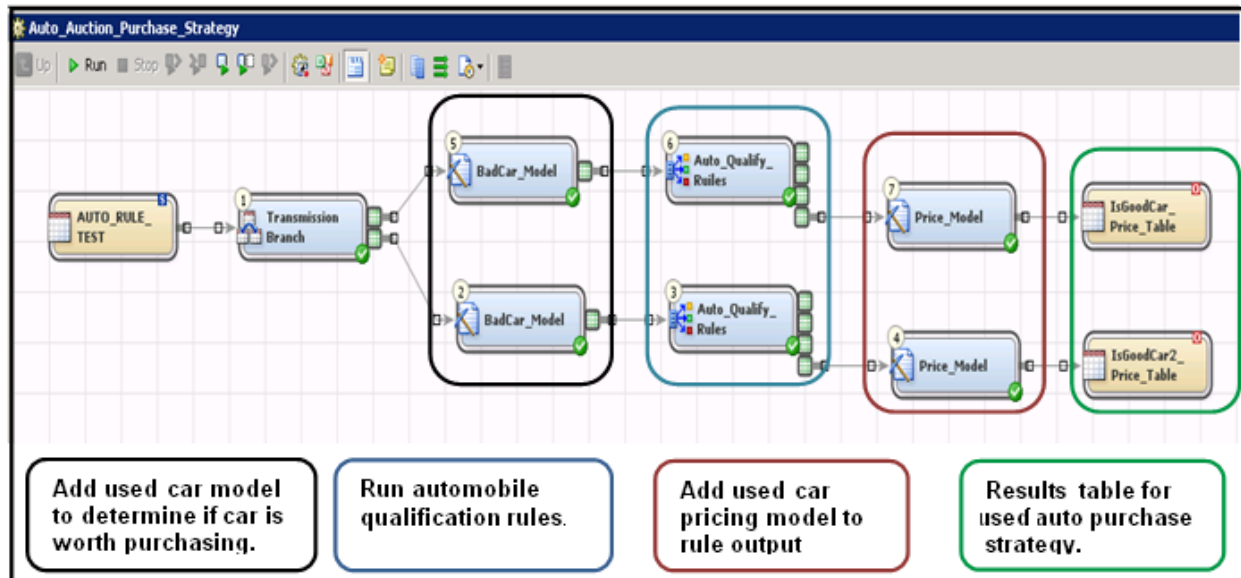


Table 1: Data, Models, and Business Rules Process Flow within SAS Data Integration Studio

Mapping data columns to model parameters (Table 2) and mapping business rules to data columns (Table 3) within SAS Data Integration Studio seen below is easily accomplished through the user interface.

The screenshot shows the 'BadCar\_Model Properties' dialog box with the 'Mappings' tab selected. It displays the mapping between the 'Source table: Splitter 0 (W1L534PM)' and the 'Target table: InputTable'. The source table columns are: #, hgh, Informat, Format, Is Nullable, and Column. The target table columns are: #, Column, Column Description, Expression, Type, Length, and Informa. Arrows indicate the mapping from source columns to target columns: hgh to auction, Informat to color, Format to make, Is Nullable to transmission, and Column to vehicleage and wheeltpeid.

Source Table	Column	Target Table	Column
Splitter 0 (W1L534PM)	hgh	InputTable	1 auction
Splitter 0 (W1L534PM)	Informat	InputTable	2 color
Splitter 0 (W1L534PM)	Format	InputTable	3 make
Splitter 0 (W1L534PM)	Is Nullable	InputTable	4 transmission
Splitter 0 (W1L534PM)	Column	InputTable	5 vehicleage
Splitter 0 (W1L534PM)	Column	InputTable	6 wheeltpeid

Table 2: Mapping Data Columns to Model Parameters



#	Column	Column Description	Type	Length
1	CORRELATION_ID	CORRELATION_ID	Character	32 \$
2	REFID	REFID	Numeric	8 (3)
3	SRACBUY	SRACBUY	Numeric	8 (3)
4	PURCHDATE	PURCHDATE	Numeric	8 (3)
5	AUCTION	AUCTION	Character	7 \$
6	VEHYEAR	VEHYEAR	Numeric	8 (3)
7	VEHCLEAGE	VEHCLEAGE	Numeric	8 (3)
8	MAKE	MAKE	Character	12 \$
9	MODEL	MODEL	Character	20 \$
10	TRIM	TRIM	Character	3 \$

#	Term	Term Description	Type
1	make	Make of vehicle	String
2	eventProbab...	Probability that veh...	Decimal
3	wheeltype	Type of wheel	String
4	vehbcost	Vehicle Base Cost	Integer
5	vint	State of Vehicle VIN	String
6	vehicleage	Age of vehicle in ye...	Integer

Port	Table	Complete
Business Rule Flow Mapping Port	Business Rule Flow Mapping Table	Complete
DOM_DEPLOYMENT_ID_ELECTION	DOM_DEPLOYMENT_ID_ELECTION	Complete
DOM_RULE_ACTION_FIRE	DOM_RULE_ACTION_FIRE	Complete
DOM_RULE_ACTION_FIRE	DOM_RULE_ACTION_FIRE	Complete

Table 3: Mapping Business Rule Terms to Data Columns

## DECISION FLOW BUILDER - COMBINE MODELS AND RULES

### Bringing data, models, and rules together using SAS Real-Time Decision Manager

The approach to formulate a logical flow using the Flow Builder is shown in Table 4 below. Data, models, and business rules are related to each other by simply dropping objects onto the pallet and setting their properties. Additional objects such as branching and other data, SAS, or web processes are available for creating or enhancing decision logic. The flow can be run and tested from the same user interface. All objects seen in Table 5 below are created using definitions that are accessible within a Decision Flow.

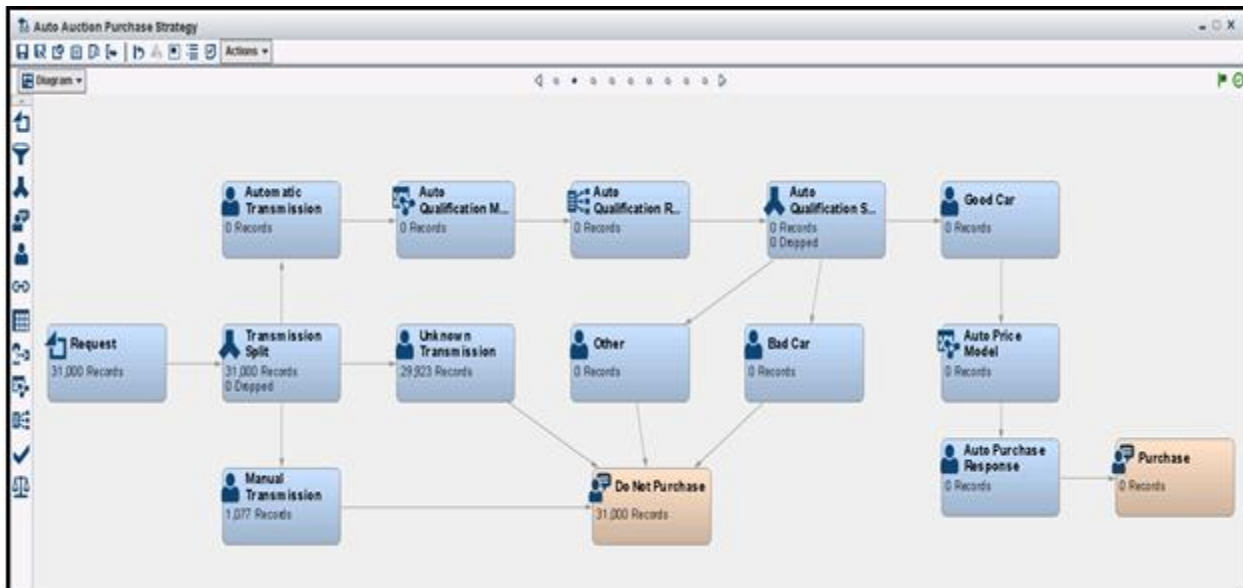


Table 4 – Data, Models, and Business Rules within Flow Builder



Name	Rule Flow	Vocabulary	Rule Sets	Date Modified	Modified By
Auto					
AutoDecideToBuyRules	AutoPurchaseRuleFlow	Auto	3	Sep 13, 2013 08:34 PM	SAS Demo User
AutoQualificationRuleFlow	AutoQualificationRuleFlow	Auto	3	Oct 3, 2013 09:22 AM	SAS Demo User
DecideToBuyFlow	DecideToBuyFlow	AutoLoan	1	Sep 13, 2013 08:35 PM	SAS Demo User
Loans					
Sales Ops Strategy					

**Table 5 – Left Side Shows Categories of Object Definitions That Are Accessible within the Flow Builder**

## Deployment of Decision Components

Both batch and near real-time deployments are supported. For batch deployments, use SAS Data Integration Studio where code is saved and then scheduled with any scheduler. Any process flow built within SAS Data Integration Studio can become a candidate for scheduling.

The components for SAS Decision Manager provide a scalable tiered architecture for design time and run time. There are two ways to deploy logical flows within a production environment or deploy as web services. Models and business rules deployed as SOAP or REST web services provide a great amount of flexibility for application integration for the downstream consumer.

If stand-alone Business rules deployment is desired, this can be accomplished via web services. Web services offer true flexibility for application integration. Below is an example of the auto qualification rules used above in our batch job that could also be deployed via a web service. In Table 6 below, the business rule request/reply is seen below from the SOAP UI.

Request	Response
<pre> &lt;?xml version='1.0' encoding='UTF-8'&gt; &lt;soap:Envelope xmlns:soap='http://schemas.xmlsoap.org/soap/envelope/'&gt;   &lt;soap:Header/&gt;   &lt;soap:Body&gt;     &lt;auto:Auto_BRM_MinimumCode&gt;       &lt;parameters&gt;         &lt;vehcost&gt;7100&lt;/vehcost&gt;         &lt;vtype&gt;FL&lt;/vtype&gt;         &lt;vehcleage&gt;3&lt;/vehcleage&gt;         &lt;make&gt;MAZDA&lt;/make&gt;         &lt;vehcleage&gt;3&lt;/vehcleage&gt;         &lt;eventProbability&gt;0.06345&lt;/eventProbability&gt;       &lt;/parameters&gt;     &lt;/auto:Auto_BRM_MinimumCode&gt;   &lt;/soap:Body&gt; &lt;/soap:Envelope&gt; </pre>	<pre> &lt;?xml version='1.0' encoding='UTF-8'&gt; &lt;soap:Envelope xmlns:soap='http://schemas.xmlsoap.org/soap/envelope/'&gt;   &lt;soap:Header/&gt;   &lt;soap:Body&gt;     &lt;auto:Auto_BRM_MinimumCodeResponse xmlns:n='http://testuri.org/Auto_BRM_MinimumCodeResult'&gt;       &lt;n:Parameters&gt;         &lt;n:isadBuy&gt;0&lt;/n:isadBuy&gt;         &lt;n:vtype&gt;FL&lt;/n:vtype&gt;         &lt;n:vehcleage&gt;3&lt;/n:vehcleage&gt;         &lt;n:make&gt;MAZDA&lt;/n:make&gt;         &lt;n:eventProbability&gt;0.06345&lt;/n:eventProbability&gt;         &lt;n:refid&gt;&lt;/n:refid&gt;         &lt;n:vehcost&gt;7100&lt;/n:vehcost&gt;       &lt;/n:Parameters&gt;       &lt;n:Stream&gt;         &lt;n:_VBOUT contentType='text/html; charset=utf-8'&gt;           &lt;n:Value&gt;             &lt;sc:Include href='cid:1.urn:uuid:1f930b1f93f01090e213'&gt;               &lt;/sc:Include&gt;             &lt;/n:Value&gt;           &lt;/n:_VBOUT&gt;         &lt;/n:Stream&gt;       &lt;/n:Auto_BRM_MinimumCodeResult&gt;     &lt;/auto:Auto_BRM_MinimumCodeResponse&gt;   &lt;/soap:Body&gt; &lt;/soap:Envelope&gt; </pre>

**Table 6: Business Rules Tested as a Web Service before Deployment, Seen in the SOAP UI**

## CONCLUSION

SAS Decision Manager provides the features that businesses need as they look for easier and faster ways to deploy their powerful analytical models to their business processes. These same businesses also know that the business analysts and modelers need to collaborate easily to implement their decisions easily. SAS Decision Manager offers

the capabilities needed to bring together powerful analytical models and the flexibility of business rules that can adapt to changing business policies and guidelines.

By offering an integrated suite for modelers, business analysts, and IT, SAS Decision Manager streamlines the analytical process by supporting these roles working together to deliver decisions to the operational side of the business faster and more efficiently, thereby reducing costs and reducing the lag between the time a business can react to their changing environments and customer demands.

## REFERENCES

Taylor, James. 2012. *Decision Management Systems: A Practical Guide to Using Business Rules and Predictive Analytics*. Upper Saddle River, NJ: IBM Press.

Wexler, Jonathan; Wayne Thompson, and Kristen Aponte. 2013. "Time Is Precious, So Are Your Models: SAS® Provides Solutions to Streamline Deployment." *Proceedings of the SAS Global 2013 Conference*. Cary, NC: SAS Institute Inc. Available at <http://support.sas.com/resources/papers/proceedings13/086-2013.pdf>.

SAS Institute Inc. 2012. "SAS® Data Management Advanced Fact Sheet." Cary, NC: SAS Institute Inc. Available at <http://www.sas.com/resources/factsheet/sas-enterprise-data-integration-server-factsheet.pdf>.

## ACKNOWLEDGMENTS

The authors of this paper would like to thank the members of the various teams in R&D, Product Management, Marketing, and of course, our customers.

## RECOMMENDED READING

- *SAS Business Rules Manager 2.1: User's Guide*
- *SAS Model Manager 12.3: User's Guide*
- *SAS Data Integration Studio 4.8: User's Guide*

## CONTACT INFORMATION

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

Stephen Sparano  
100 SAS Campus Drive  
Cary, NC 27513  
SAS Institute Inc.  
[Steve.Sparano@sas.com](mailto:Steve.Sparano@sas.com)  
<http://www.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.