Loan Origination Rule Modeling Challenge (June 2017) Corticon Solution

Mike Parish

Introduction

The Loan Origination problem is described in Section 11 of the DMN specification

The example is quite simple. It has no hierarchical data structures or collections of repeating objects to deal with so translating the DMN model to Corticon (and other tools) is straightforward.

The main differences in Corticon are:

1. Decision tables read vertically instead of horizontally
2. Decision tables do not need the DMN hit policy codes (U, C+, C<, C>, C#, C-, P, A, R, F, O, N etc.) in order to clarify the meaning of the table.
3. Dependency diagrams can be created automatically by analyzing the inputs and outputs of each decision table.

The advantages of automatically creating the dependency diagram are:

1. You can see the true dependencies between the decision tables (even though a DRD shows a dependency it doesn't mean there necessarily is one)
2. The diagram can be automatically updated whenever a change is made to any rule.

The rule sheets are presented below, mostly in natural language form, since for the most part the conditions and actions are simple attributes. In a few cases the implementation is provided where more complex conditions or actions are involved.

Other than that, the Corticon solution looks pretty much like the DMN example.

The DMN example as documented does have a number of design flaws and some of these are discussed below.

Determine Strategy

Determine Strategy.ers

	Conditions	1	2	3
a	Eligibility	'INELIGIBLE'	'ELIGIBLE'	'ELIGIBLE'
b	Bureau Call Type	-	{'FULL', 'MINI'}	'NONE'
	Actions			
	Post Message(s)			
A	Strategy	'DECLINE'	'BUREAU'	'THROUGH'

Determine Bureau Call Type

Determine Bureau Call Type.ers

	Conditions	1	2	3
a	Pre-bureau risk category	{'HIGH', 'MEDIUM'}	'LOW'	{'VERY LOW', 'DECLINE'}
	Actions			
	Post Message(s)			
A	Bureau call type	'FULL'	'MINI'	'NONE'

Eligibility

Determine Eligibility.ers

	Conditions	1	2	3	4	5	6	7
a	Pre-bureau risk category	{'HIGH', 'MEDIUM', 'LOW', 'VERY LOW'}	'DECLINE'	-	-	other	-	-
b	Pre-bureau Affordability	T	-	F	-	-	other	-
c	Applicant age	>= 18	-	-	< 18	-	-	other
d								
e								
	Actions							
	Post Message(s)	✉	✉	✉	✉			
A	Eligibility	'ELIGIBLE'	'INELIGIBLE'	'INELIGIBLE'	'INELIGIBLE'			
B	Invalid Data					☑	☑	☑

Here we have chosen to make rule 1 very explicit about what constitute the eligibility conditions.
The original DMN example will classify someone as ELIGIBLE even if risk, affordability and age are all unknown. This is probably not what we want.
We've also added rules 5-7 to catch any possible invalid data (such as null or a misspelled risk category for example)
There may also be several reasons they are INELIGIBLE and we may want to know all of them, not just the first. Corticon will execute all applicable rules. Also in Corticon rule statements can be attached to

the rules to provide both the reasoning behind the rule and appropriate audit trail explanatory messages:

Ref	Post	Alias	Text	Rule Name
1	Info	Applicant	The applicant is ELIGIBLE because the pre bureau risk category is not DECLINE, the loan is affordable and the age is 18 or more	E1
2	Warning	Applicant	The applicant is INELIGIBLE because the pre bureau risk category is DECLINE	E2
3	Warning	Applicant	The applicant is INELIGIBLE because the loan is not affordable	E3
4	Warning	Applicant	The applicant is INELIGIBLE because they are under 18	E4

In the case of rule 3 for example we could go further and actually calculate and display the maximum amount that would be considered affordable. This would be much more helpful to the customer than simply saying the loan is not affordable.

Pre-Bureau Risk Category

These rules could have been represented as a single decision table but splitting them based on the different handling of new and existing customers produces more manageable tables and also allows for easier divergence in the future. Maybe in the future we might consider some different attributes for existing customers (like how long have they been a customer) which would not apply to a new customer.

Determine Pre Bureau Risk Category New Customer.ers

	Conditions	1	2	3	4
a	Existing Customer?	F	F	F	F
b	Applicant Risk Score	< 100	[100..120)	[120..130]	> 130
	Actions				
	Post Message(s)				
A	Pre-bureau Risk Category	'HIGH'	'MEDIUM'	'LOW'	'VERY LOW'

Determine Pre Bureau Risk Category Existing Customer.ers

	Conditions	1	2	3	4
a	Existing Customer?	T	T	T	T
b	Applicant Risk Score	< 80	[80..90)	[90..110]	> 110
	Actions				
	Post Message(s)				
A	Pre-bureau Risk Category	'DECLINE'	'HIGH'	'MEDIUM'	'LOW'

Observation

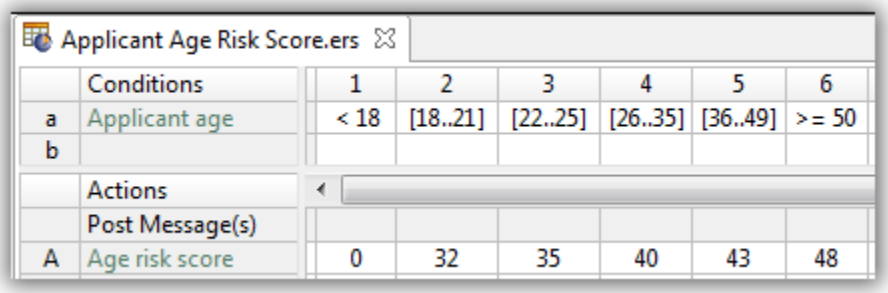
Interestingly, existing customers can be DECLINEd at this point but new customers, at worst, receive a HIGH risk. And existing customer can never receive a VERY LOW risk. Perhaps this reflects a desire to get more new customers rather than to grant loans to existing customers? Or perhaps this is not what we intended?

Also, it does seem a little counter-intuitive that a higher risk score actually represents a lower risk category! Maybe the name of the attribute or the scoring mechanism should be changed to avoid the cognitive dissonance. Perhaps instead of “risk” we should use “confidence” so when *confidence* score is high then the *risk* category is low.

Determine Applicant Risk Score

We have opted to score each factor (age, marital, employment) separately in order to get better visibility of each factors' contribution. It also makes it easier to add new factors as desired. Also in order to make each decision table complete, any input value not mentioned in the DMN spec will result in a risk score factor of zero.

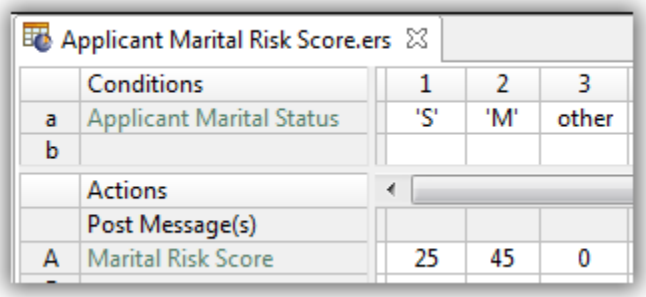
Applicant Age Risk



Applicant Age Risk Score.ers

	Conditions	1	2	3	4	5	6
a	Applicant age	< 18	[18..21]	[22..25]	[26..35]	[36..49]	>= 50
b							
	Actions						
	Post Message(s)						
A	Age risk score	0	32	35	40	43	48

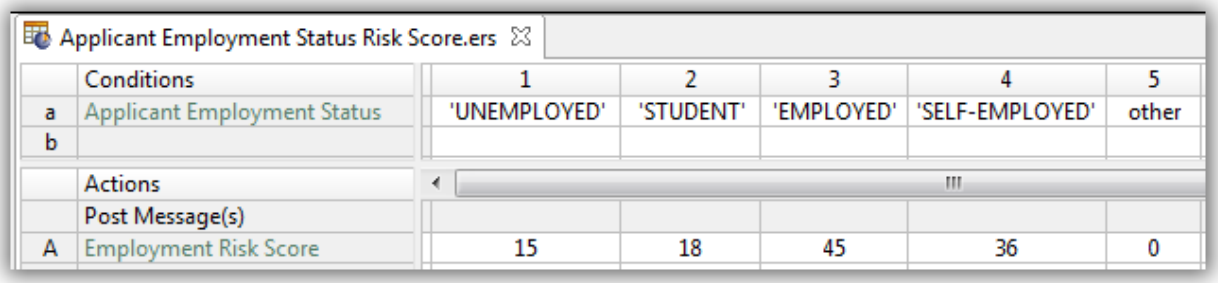
Applicant Marital Status Risk



Applicant Marital Risk Score.ers

	Conditions	1	2	3
a	Applicant Marital Status	'S'	'M'	other
b				
	Actions			
	Post Message(s)			
A	Marital Risk Score	25	45	0

Applicant Employment Risk



Applicant Employment Status Risk Score.ers

	Conditions	1	2	3	4	5
a	Applicant Employment Status	'UNEMPLOYED'	'STUDENT'	'EMPLOYED'	'SELF-EMPLOYED'	other
b						
	Actions					
	Post Message(s)					
A	Employment Risk Score	15	18	45	36	0

If it should turn out that no other values should occur in the input data then we can use rule 5 to issue an error message. In general it's a good practice to cover such situations even when the business users say "oh, that will never occur". Sooner or later it will and a suitable error message can avoid hours of tedious debugging. As a rule modeling best practice, every rule should have an appropriate rule statement that explains what happened.

Applicant Total Risk Score

Applicant Total Risk Score.ers

	Conditions	0
a		
	Actions	
	Post Message(s)	
A	Applicant total risk = age risk + marital risk + employment risk	☑

New risk factors can be added in here, as required.
We also have the ability here to add weighting factors if that becomes necessary.

Routing

*Determine Routing.ers

	Conditions	1	2	3	4	5
a	Post-bureau affordability	F	-	T	T	T
b	Is applicant bankrupt?	-	T	F	F	not T
c	Post-bureau risk category	-	-	'HIGH'	-	not 'HIGH'
d	Applicant credit score	-	-	-	< 580	{>= 580, null}
	Actions					
	Post Message(s)	✉	✉	✉	✉	✉
A	Routing	'DECLINE'	'DECLINE'	'REFER'	'REFER'	'ACCEPT'

Again we'll opt to have Corticon report all reasons for DECLINE or REFER, not just the first and we'll also make the ACCEPT rule explicit. Note that the original DMN example allows nulls for bankruptcy and credit score and this could be another source of ambiguity.

Ref	Post	Alias	Text
1	Warning	Loan	DECLINE the loan because it is not affordable
2	Warning	Loan	DECLINE the loan because the applicant in bankrupt
3	Info	Loan	REFER the loan because the post bureau risk is HIGH
4	Info	Loan	REFER the loan because the credit score is under 580
5	Info	Loan	ACCEPT the loan (affordable, not bankrupt, not HIGH risk, credit score >=580 or null)

Post Bureau Risk Category

Again these are split to explicitly recognize the different handling of new and existing customers. Once again only new customers can possibly get a VERY LOW risk category.

New Customers

Determine Post Bureau Risk Category New Customer.ers

	Conditions	1	2	3	4	5	6	7
a	Existing Customer?	F	F	F	F	F	F	F
b	Applicant risk score	< 120	< 120	< 120	[120..130]	[120..130]	[120..130]	> 130
c	Applicant credit score	< 590	[590..610]	> 610	< 600	[600..625]	> 625	-
	Actions							
	Post Message(s)							
A	Post-bureau risk category	'HIGH'	'MEDIUM'	'LOW'	'HIGH'	'MEDIUM'	'LOW'	'VERY LOW'

Existing Customers

Determine Post Bureau Risk Category Existing Customer.ers

	Conditions	1	2	3	4	5	6
a	Existing Customer?	T	T	T	T	T	T
b	Applicant risk score	<= 100	<= 100	<= 100	> 100	> 100	> 100
c	Applicant credit score	< 580	[580..600]	> 600	< 590	[590..615]	> 615
	Actions						
	Post Message(s)						
A	Post-bureau risk category	'HIGH'	'MEDIUM'	'LOW'	'HIGH'	'MEDIUM'	'LOW'

Determine Affordability

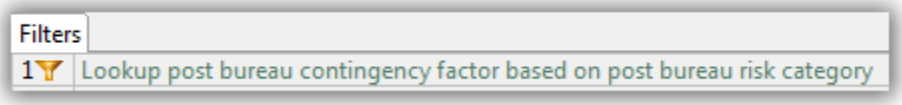
*Create In Memory Contingency Factors Lookup Table.ers

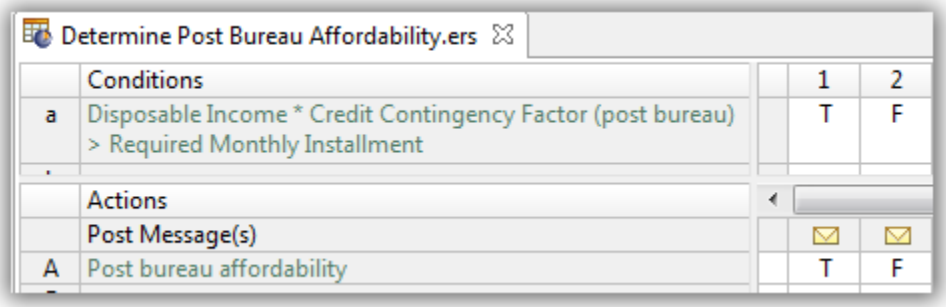
	Conditions	0
a		
	Actions	
	Post Message(s)	
A	HIGH category has a contingency of	0.6
B	DECLINE category has a contingency of	0.6
C	MEDIUM category has a contingency of	0.7
D	LOW category has a contingency of	0.8
E	VERY LOW category has a contingency of	0.8

Implementation detail

This rule sheet creates an in-memory lookup table of the risk categories and their corresponding contingency factors. This approach makes it very easy to switch to an external database that contains these values if necessary

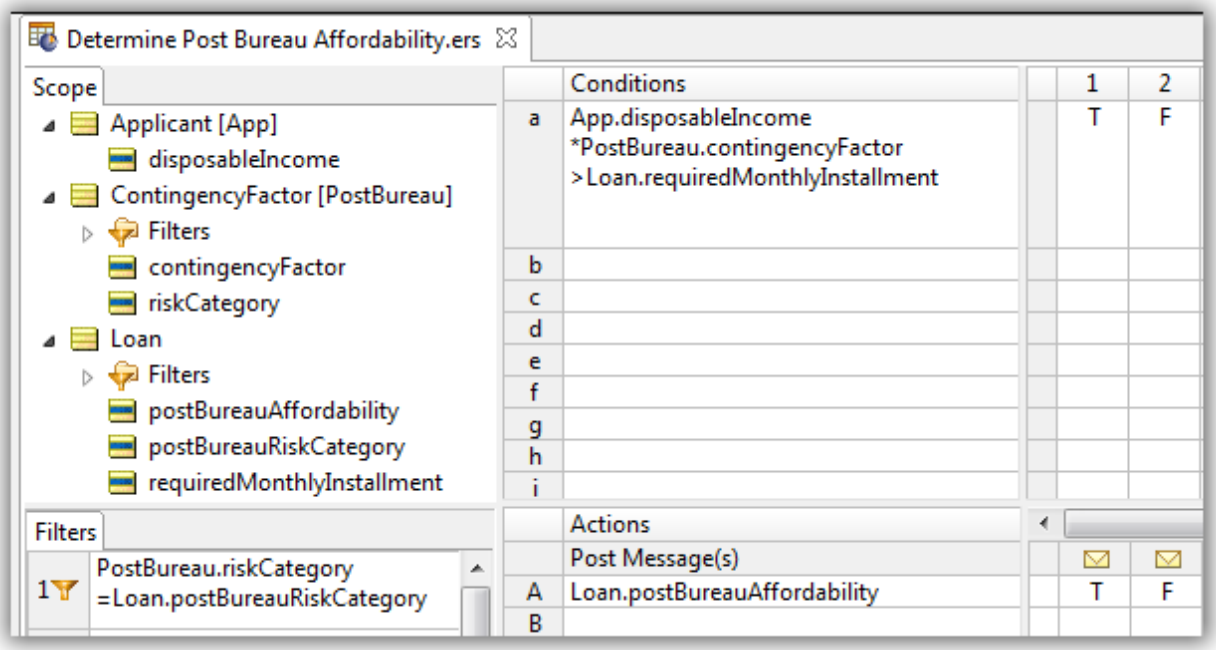
ContingencyFactor.new[riskCategory='HIGH',contingencyFactor=cellValue]	0.6
ContingencyFactor.new[riskCategory='DECLINE',contingencyFactor=cellValue]	0.6
ContingencyFactor.new[riskCategory='MEDIUM',contingencyFactor=cellValue]	0.7
ContingencyFactor.new[riskCategory='LOW',contingencyFactor=cellValue]	0.8
ContingencyFactor.new[riskCategory='VERY LOW',contingencyFactor=cellValue]	0.8



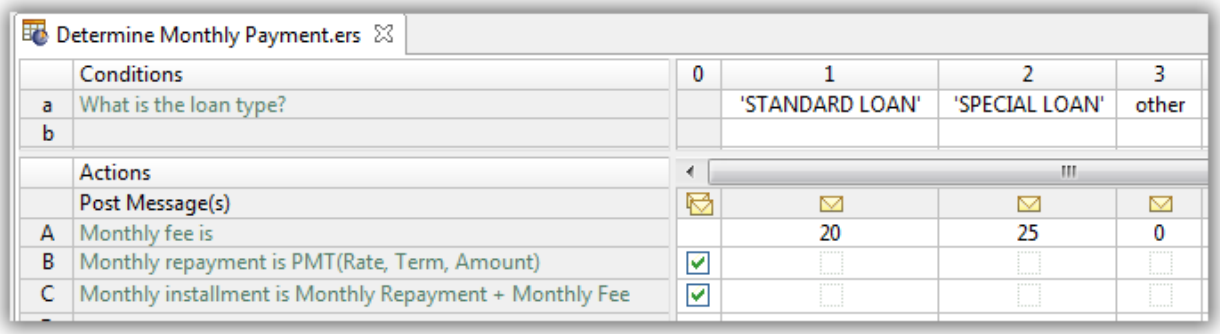


The pre-bureau affordability is handled in the same way. Both make use of the same common lookup table to get the contingency factors appropriate to the pre or post bureau risk category.
The filter is what locates the correct factor by matching the lookup table row to the risk category.

Implementation Details



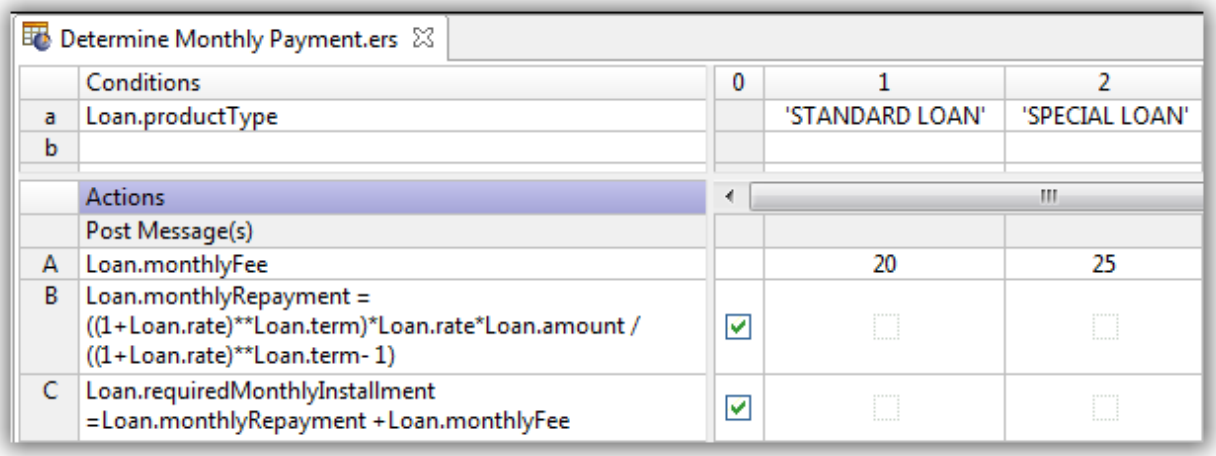
Determine Required Monthly Payment



Determine Monthly Payment.ers

	Conditions	0	1	2	3
a	What is the loan type?		'STANDARD LOAN'	'SPECIAL LOAN'	other
b					
	Actions				
	Post Message(s)				
A	Monthly fee is		20	25	0
B	Monthly repayment is PMT(Rate, Term, Amount)	☑	☐	☐	☐
C	Monthly installment is Monthly Repayment + Monthly Fee	☑	☐	☐	☐

Implementation Detail



Determine Monthly Payment.ers

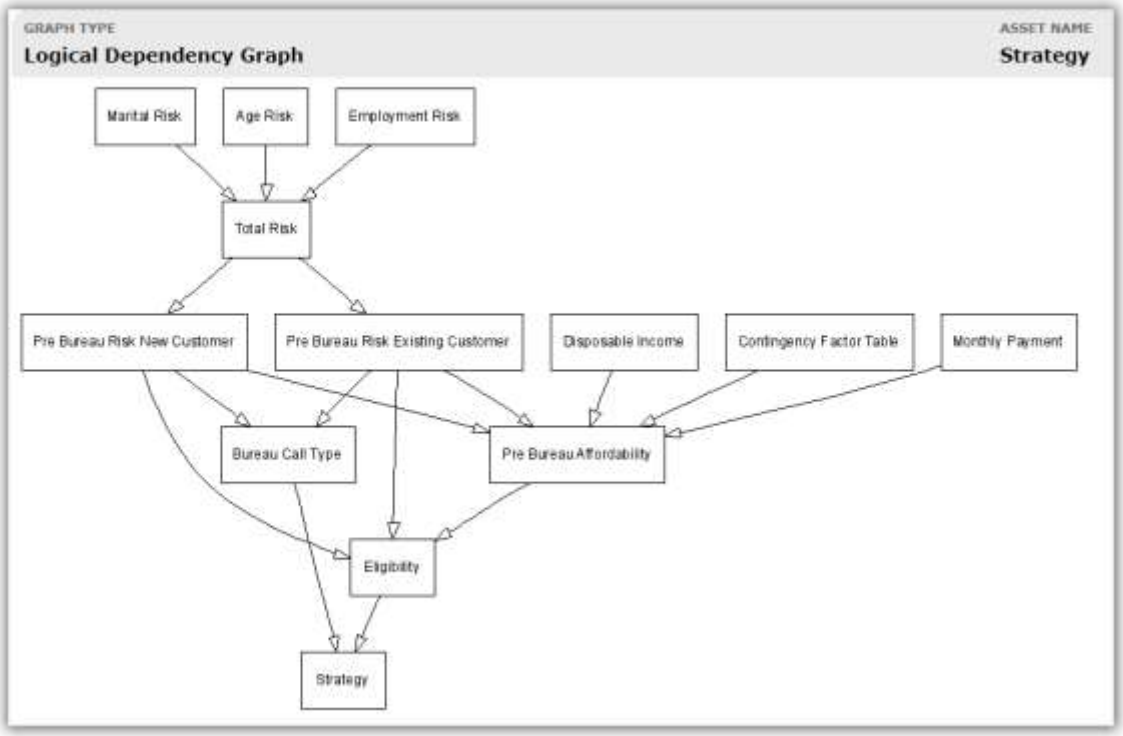
	Conditions	0	1	2
a	Loan.productType		'STANDARD LOAN'	'SPECIAL LOAN'
b				
	Actions			
	Post Message(s)			
A	Loan.monthlyFee		20	25
B	Loan.monthlyRepayment = ((1+Loan.rate)**Loan.term)*Loan.rate*Loan.amount / ((1+Loan.rate)**Loan.term- 1)	☑	☐	☐
C	Loan.requiredMonthlyInstallment =Loan.monthlyRepayment +Loan.monthlyFee	☑	☐	☐

The calculation of the payment could be accomplished by a call to an external function. In this case we have chosen to provide the detail of how the calculation is done in the rule sheet itself.
Note that Corticon will automatically determine the correct order of execution of these statements based on the dependencies between them. The rule author doesn't need to worry about that.

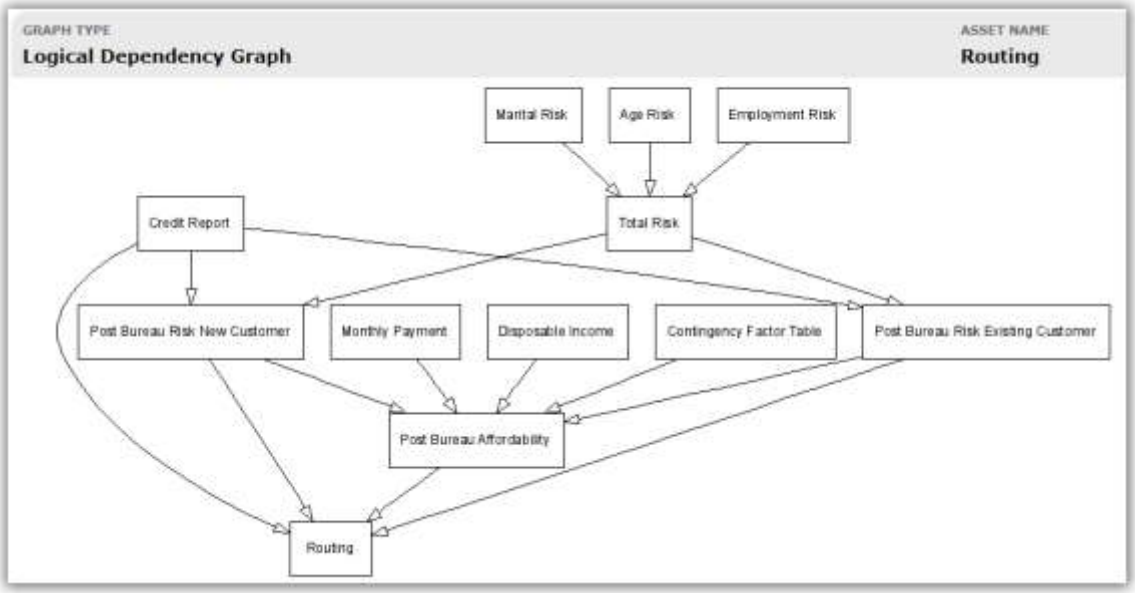
Rule Sheet Dependency Diagram

In Corticon, diagrams can be generated automatically to show the dependencies between the various rule sheets. Each box in the diagrams below is an individual rule sheet (decision table).

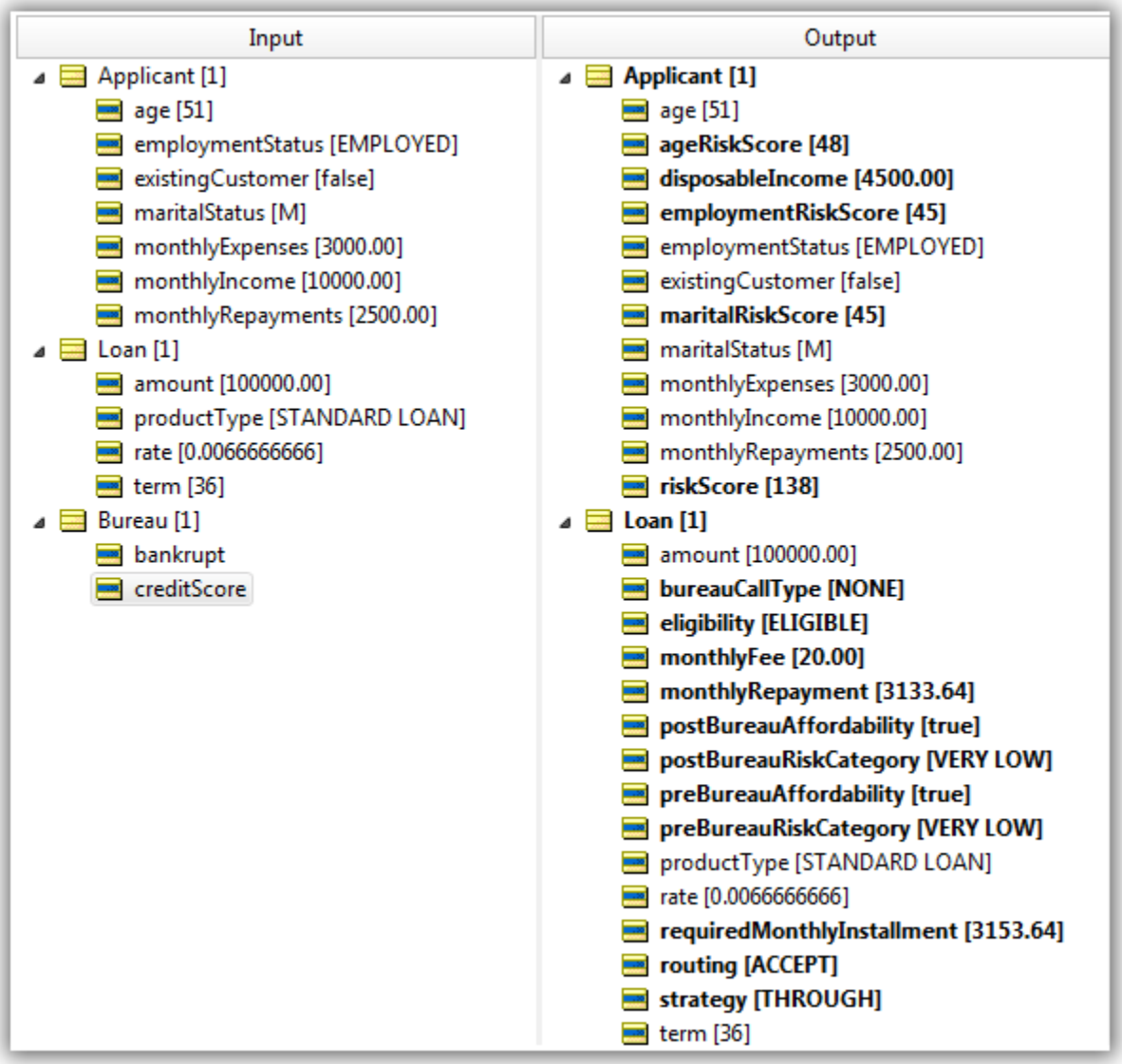
Strategy



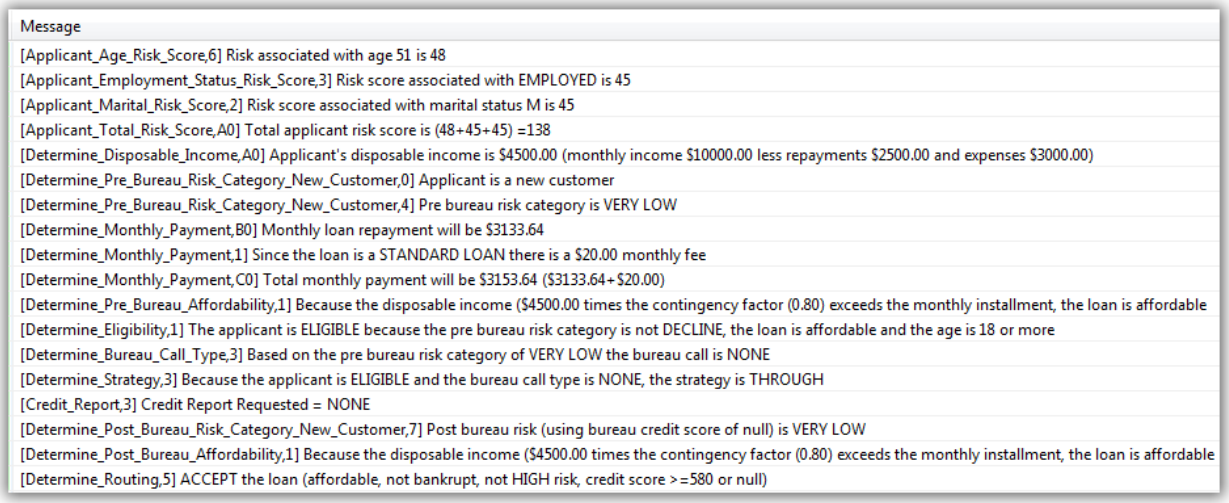
Routing



Test Case


Input
Applicant [1]
age [51]
employmentStatus [EMPLOYED]
existingCustomer [false]
maritalStatus [M]
monthlyExpenses [3000.00]
monthlyIncome [10000.00]
monthlyRepayments [2500.00]
Loan [1]
amount [100000.00]
productType [STANDARD LOAN]
rate [0.0066666666]
term [36]
Bureau [1]
bankrupt
creditScore
Output
Applicant [1]
age [51]
ageRiskScore [48]
disposableIncome [4500.00]
employmentRiskScore [45]
employmentStatus [EMPLOYED]
existingCustomer [false]
maritalRiskScore [45]
maritalStatus [M]
monthlyExpenses [3000.00]
monthlyIncome [10000.00]
monthlyRepayments [2500.00]
riskScore [138]
Loan [1]
amount [100000.00]
bureauCallType [NONE]
eligibility [ELIGIBLE]
monthlyFee [20.00]
monthlyRepayment [3133.64]
postBureauAffordability [true]
postBureauRiskCategory [VERY LOW]
preBureauAffordability [true]
preBureauRiskCategory [VERY LOW]
productType [STANDARD LOAN]
rate [0.0066666666]
requiredMonthlyInstallment [3153.64]
routing [ACCEPT]
strategy [THROUGH]
term [36]

The associated audit trail that explains how the result was determined is as follows:



Message
[Applicant_Age_Risk_Score,6] Risk associated with age 51 is 48
[Applicant_Employment_Status_Risk_Score,3] Risk score associated with EMPLOYED is 45
[Applicant_Marital_Risk_Score,2] Risk score associated with marital status M is 45
[Applicant_Total_Risk_Score,A0] Total applicant risk score is (48+45+45) =138
[Determine_Disposable_Income,A0] Applicant's disposable income is $4500.00 (monthly income $10000.00 less repayments $2500.00 and expenses $3000.00)
[Determine_Pre_Bureau_Risk_Category_New_Customer,0] Applicant is a new customer
[Determine_Pre_Bureau_Risk_Category_New_Customer,4] Pre bureau risk category is VERY LOW
[Determine_Monthly_Payment,B0] Monthly loan repayment will be $3133.64
[Determine_Monthly_Payment,1] Since the loan is a STANDARD LOAN there is a $20.00 monthly fee
[Determine_Monthly_Payment,C0] Total monthly payment will be $3153.64 ($3133.64+$20.00)
[Determine_Pre_Bureau_Affordability,1] Because the disposable income ($4500.00 times the contingency factor (0.80) exceeds the monthly installment, the loan is affordable
[Determine_Eligibility,1] The applicant is ELIGIBLE because the pre bureau risk category is not DECLINE, the loan is affordable and the age is 18 or more
[Determine_Bureau_Call_Type,3] Based on the pre bureau risk category of VERY LOW the bureau call is NONE
[Determine_Strategy,3] Because the applicant is ELIGIBLE and the bureau call type is NONE, the strategy is THROUGH
[Credit_Report,3] Credit Report Requested = NONE
[Determine_Post_Bureau_Risk_Category_New_Customer,7] Post bureau risk (using bureau credit score of null) is VERY LOW
[Determine_Post_Bureau_Affordability,1] Because the disposable income ($4500.00 times the contingency factor (0.80) exceeds the monthly installment, the loan is affordable
[Determine_Routing,5] ACCEPT the loan (affordable, not bankrupt, not HIGH risk, credit score >=580 or null)

Each line starts with [decision table name, rule number]