October 22, 2020 at 2:00 PST

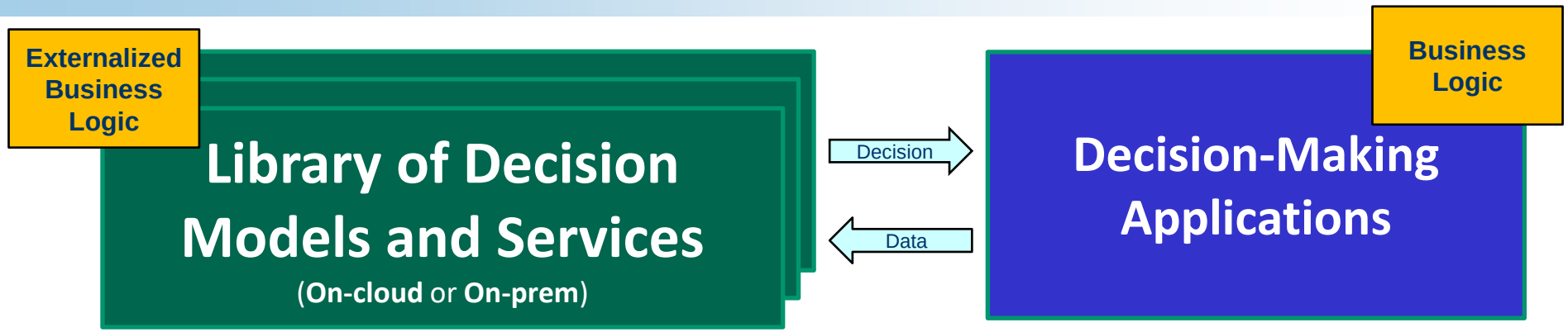
Building, Deploying, and Orchestrating **Complex Decision Services**

Jacob Feldman, PhD
OpenRules Inc., CTO
www.OpenRules.com

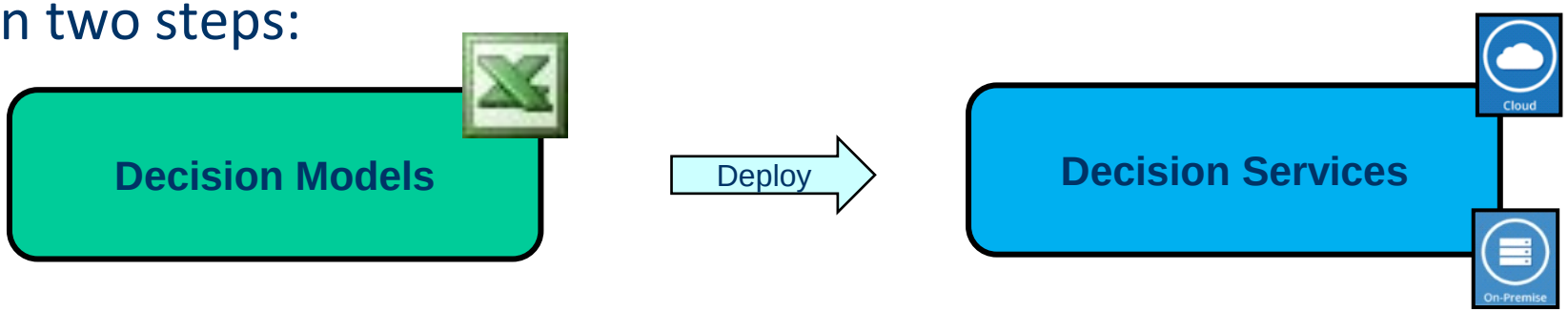
Building Complex Operational Decision Services

- This presentation is about building Operational Decision Services for different business domains
  - Not little pilots or demos but real Maintainable Decision Services
  - Created, tested, and deployed on cloud by Business Analysts
- We will use the main DMN Standard sample to build a library of Decision Models and Services for the Loan Origination domain
- Here is our implementation plan:
  1. Instead of developing a large (monolithic!) decision model, we will create a library of small reusable decision models
  2. We will test each model separately
  3. We will use these small models as “Lego Boxes” to create larger Decision Services and deploy them as cloud microservices
  4. We will create a top-level Loan Origination Service by orchestrating smaller decision services

How can Subject Mater Experts build Libraries of Operational Decision Models/Services?



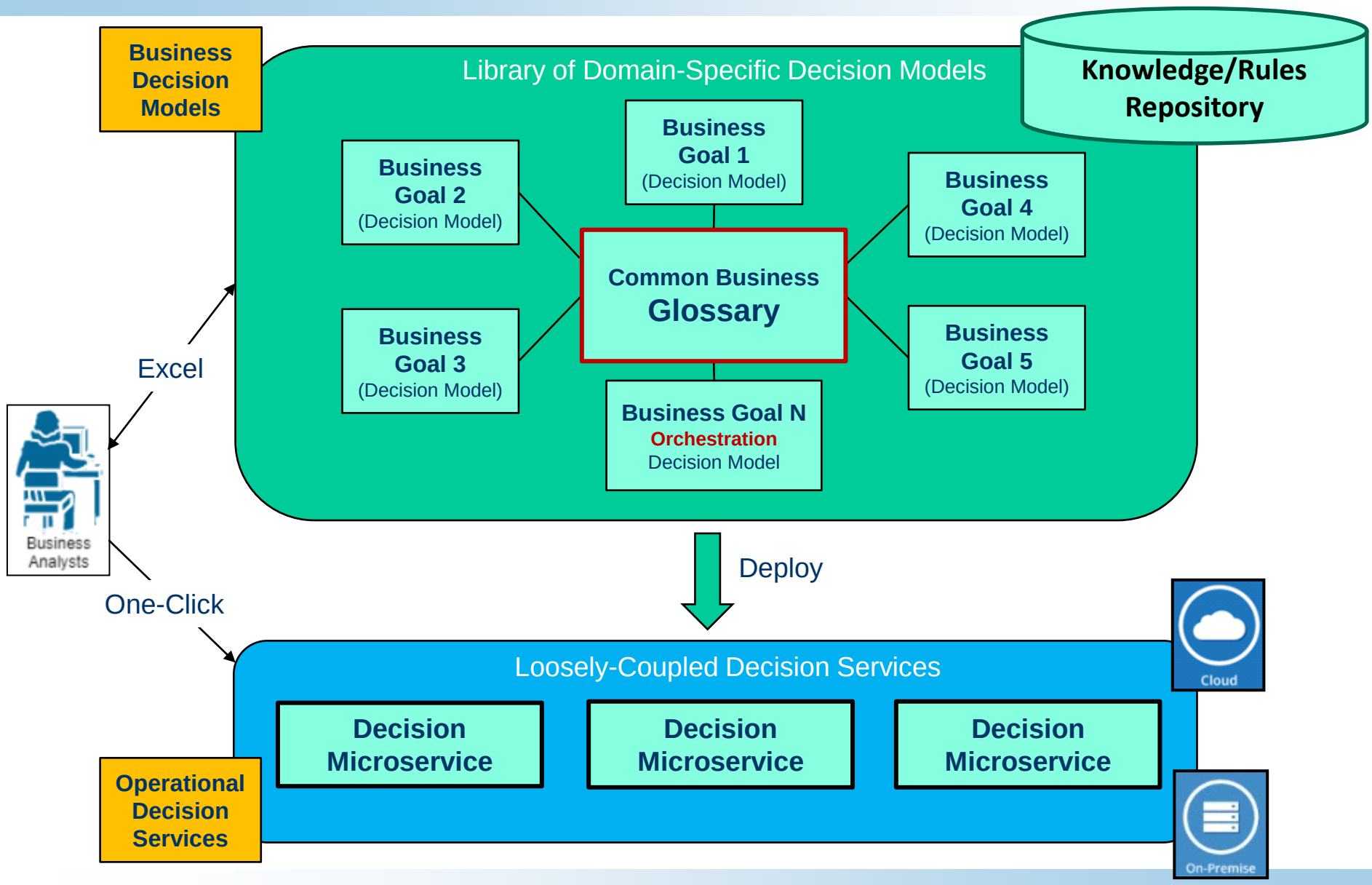
- In two steps:



STEP 1. Creating and Testing Business Decision Models

STEP 2. Deploying and Orchestrating Operational Decision Services

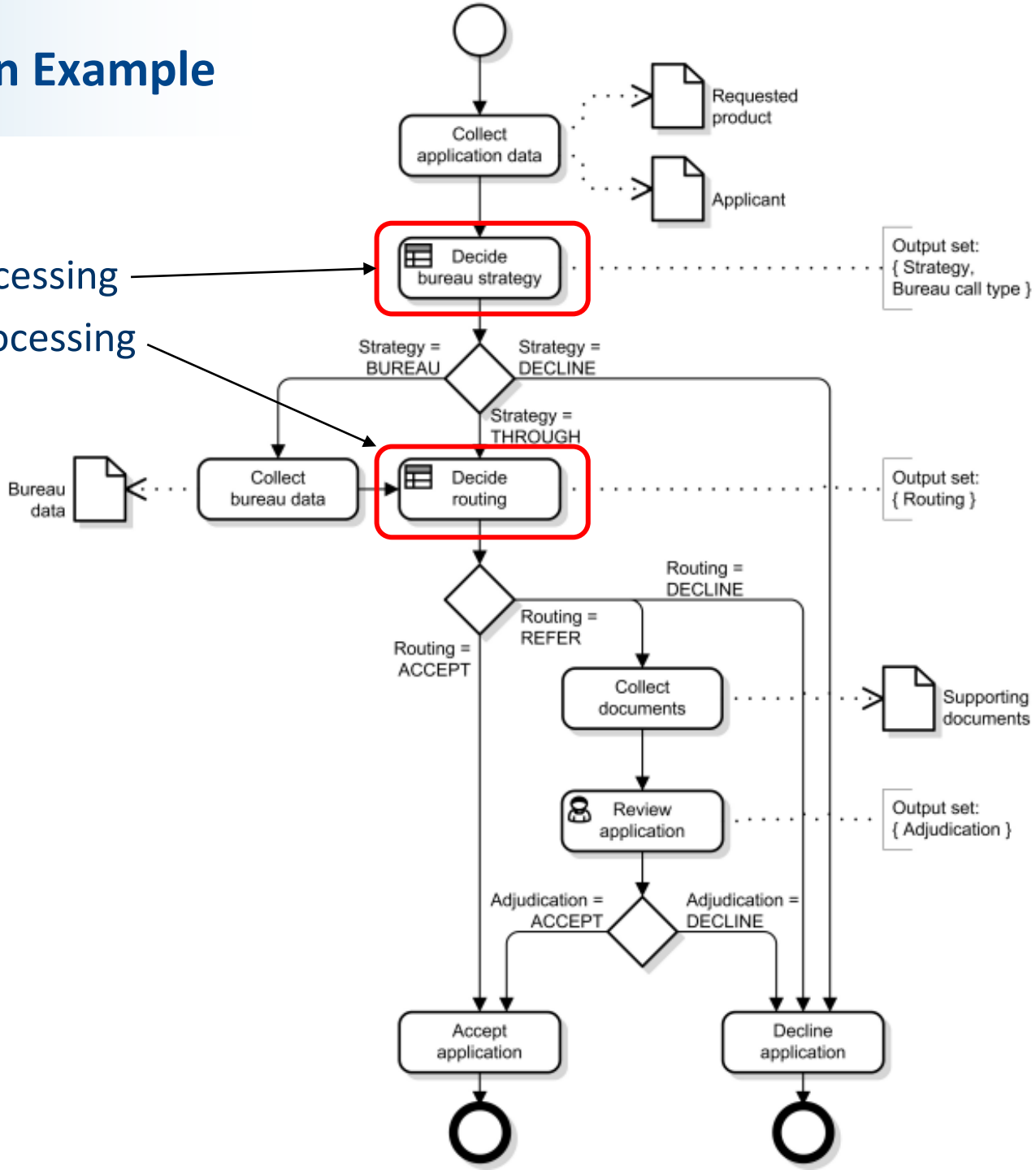
From Business Decision Models to Operational Decision Services



DMN Loan Origination Example

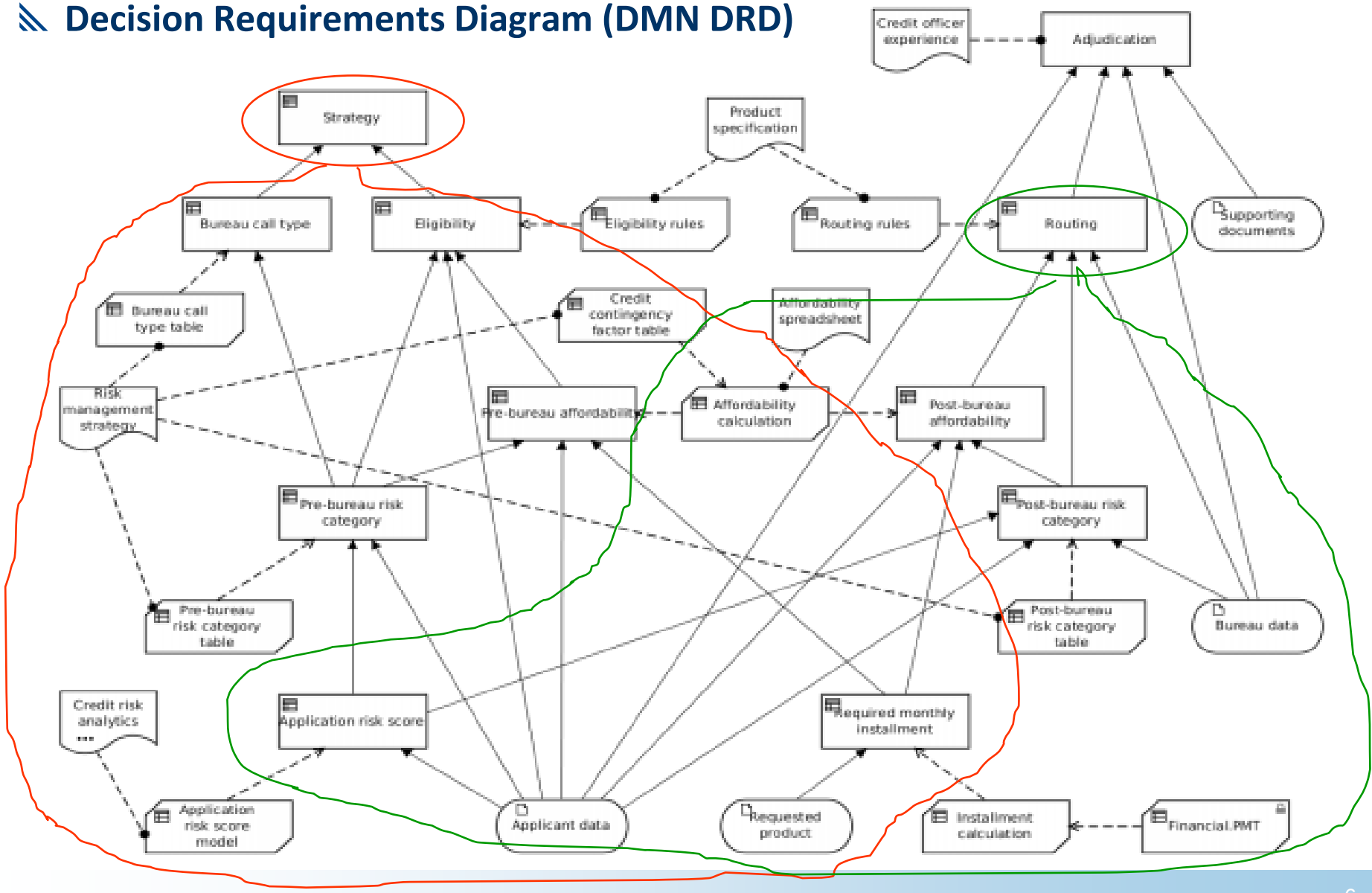
Business Process

- Pre-Bureau processing
- Post-Bureau processing



Example: DMN Loan Origination

Decision Requirements Diagram (DMN DRD)

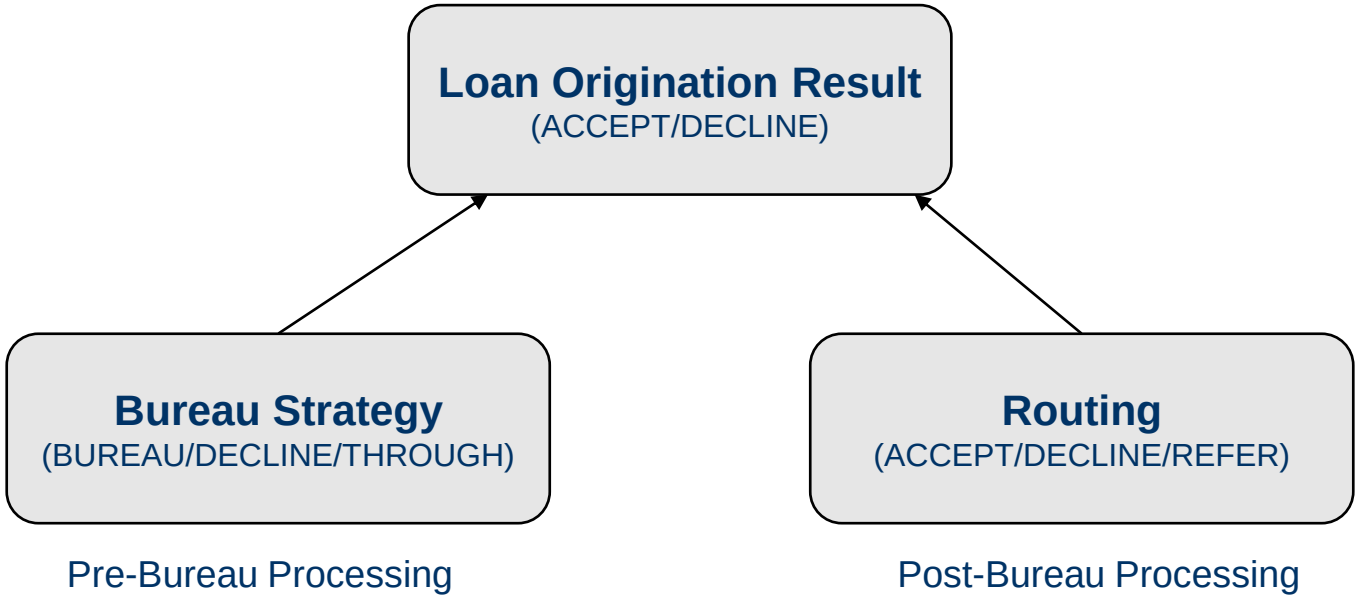


Example: DMN Loan Origination

- The above diagram with all decision variables and relationships between them is “messy” and difficult to comprehend
- We will apply the Goal-Oriented Decision Modeling Approach by selecting major goals and sub-goals, defining their internal logic using business rules, and letting a OpenRules Decision Manager to figure out all relationships and build the diagrams automatically

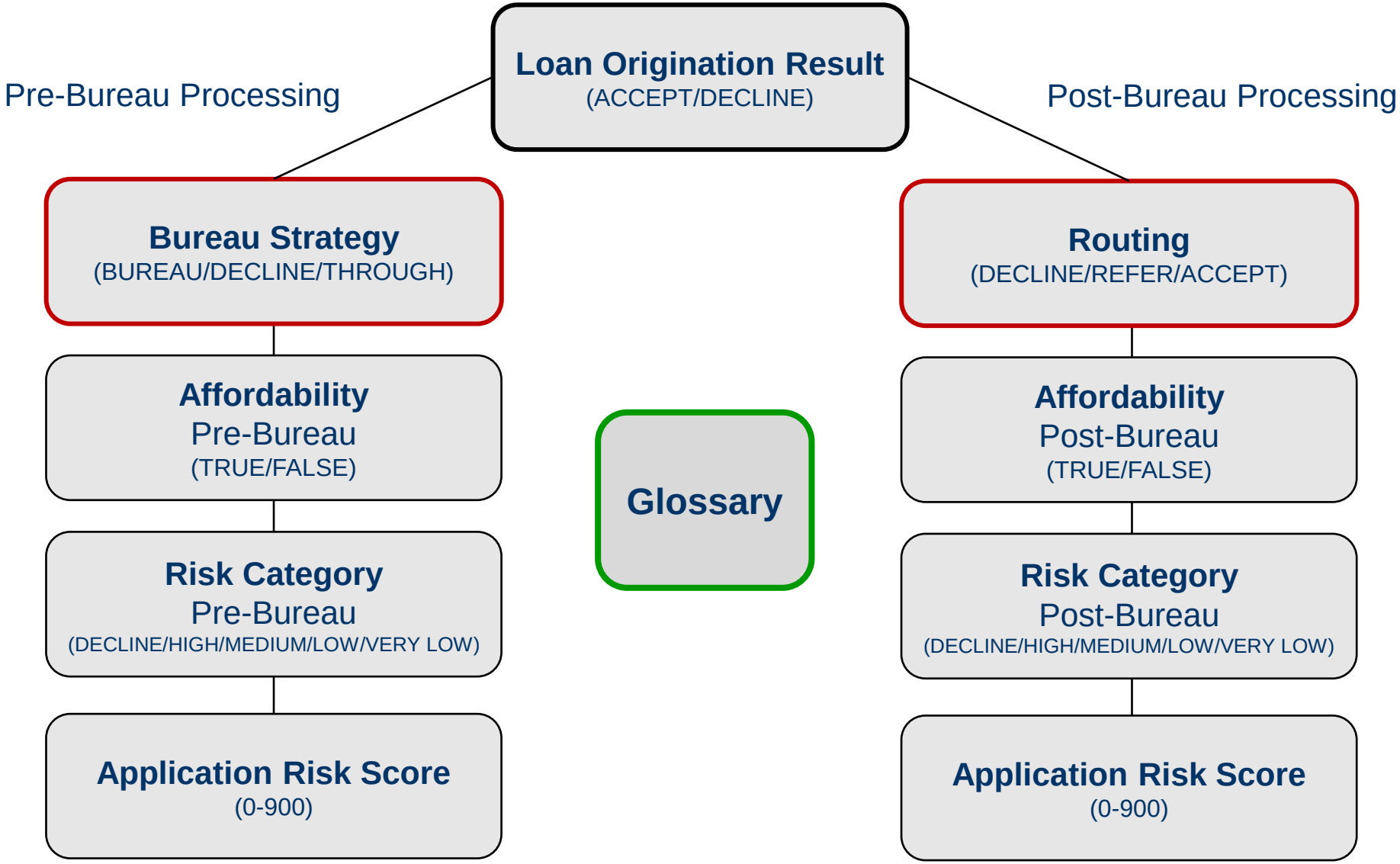
Example: DMN Loan Origination

- **Business analysts identify 3 Main Goals:**



- **These goals have common and specialized sub-goals like Affordability and Risk Category**

Decision Goals => Separate Decision Models



Let's start with an example of a complete executable decision model

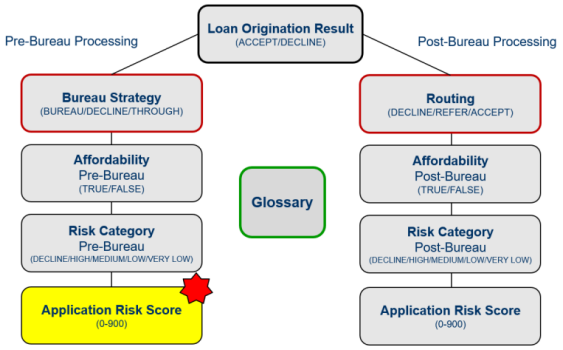
- A simple decision model "Application Risk Score"
- We will use only Excel and File Manager:

Rules.xls:

DecisionTableMultiHit ApplicationRiskScore				
If	If	If	Conclusion	
Age	Marital Status	Employment Status	Application Risk Score	
			=	0
[18..21]			+	32
[22..25]			+	35
[26..35]			+	40
[36..49]			+	43
>=50			+	48
	S		+	25
	M		+	45
		UNEMPLOYED	+	15
		STUDENT	+	18
		EMPLOYED	+	45
		SELF-EMPLOYED	+	36

Glossary.xls:

Glossary glossary			
Variable	Business Concept	Attribute	Type
Age	Applicant	age	Integer
Marital Status		maritalStatus	String
Employment Status		employmentStatus	String
Id	Application	id	String
Application Risk Score		applicationRiskScore	Integer



Decision Model "ApplicationRiskScore":

Test Cases

Data Applicant applicants			
fullName	age	maritalStatus	employmentStatus
Borrower Full Name	**Age**	**Marital Status**	**Employment Status**
Peter N. Johnson	51	M	EMPLOYED
Mary K. Brown	24	S	STUDENT
Robert Cooper Jr.	59	Other	UNEMPLOYED

Data Application applications	
id	applicationRiskScore
ID	**Application Risk Score**
1	0
2	0
3	0

DecisionTest testCases			
#	ActionUseObject	ActionUseObject	ActionExpect
Test ID	**Applicant**	**Application**	**Application Risk Score**
Test 1	applicants[0]	applications[0]	138
Test 2	applicants[1]	applications[1]	78
Test 3	applicants[2]	applications[2]	63

Demo

- Let's build and test this decision model from File Manager

```
Running 'Test 3' defined at testCases (B7:E7)
================================================
 OpenRules Decision Manager v.8.2.0
 Licensed to "Jacob Feldman" jacobfeldman@openrules.com
 Evaluation period expires on January 14, 2021
 Copyright (C) 2004-2020 OpenRules, Inc.
================================================
Execute 'ApplicationRiskScore'
   ApplicationRiskScore #1 (B5:F5)
     THEN 'Application Risk Score' = 0
     Variables:
         Application Risk Score: 0

   ApplicationRiskScore #6 (B10:F10)
     IF   'Age' >=50
     THEN 'Application Risk Score' + 48
     Variables:
         Age: 59
         Application Risk Score: 0 --> 48

   ApplicationRiskScore #9 (B13:F13)
     IF   'Employment Status'  Is UNEMPLOYED
     THEN 'Application Risk Score' + 15
     Variables:
         Application Risk Score: 48 --> 63
         Employment Status: UNEMPLOYED

Test 'Test 3' completed OK. Elapsed time 17.13 ms

------------------------------------------------
Test 1 ................................ SUCCESS
Test 2 ................................ SUCCESS
Test 3 ................................ SUCCESS
------------------------------------------------
Total time: 1831 ms.
```

Decision Model: Execution Results

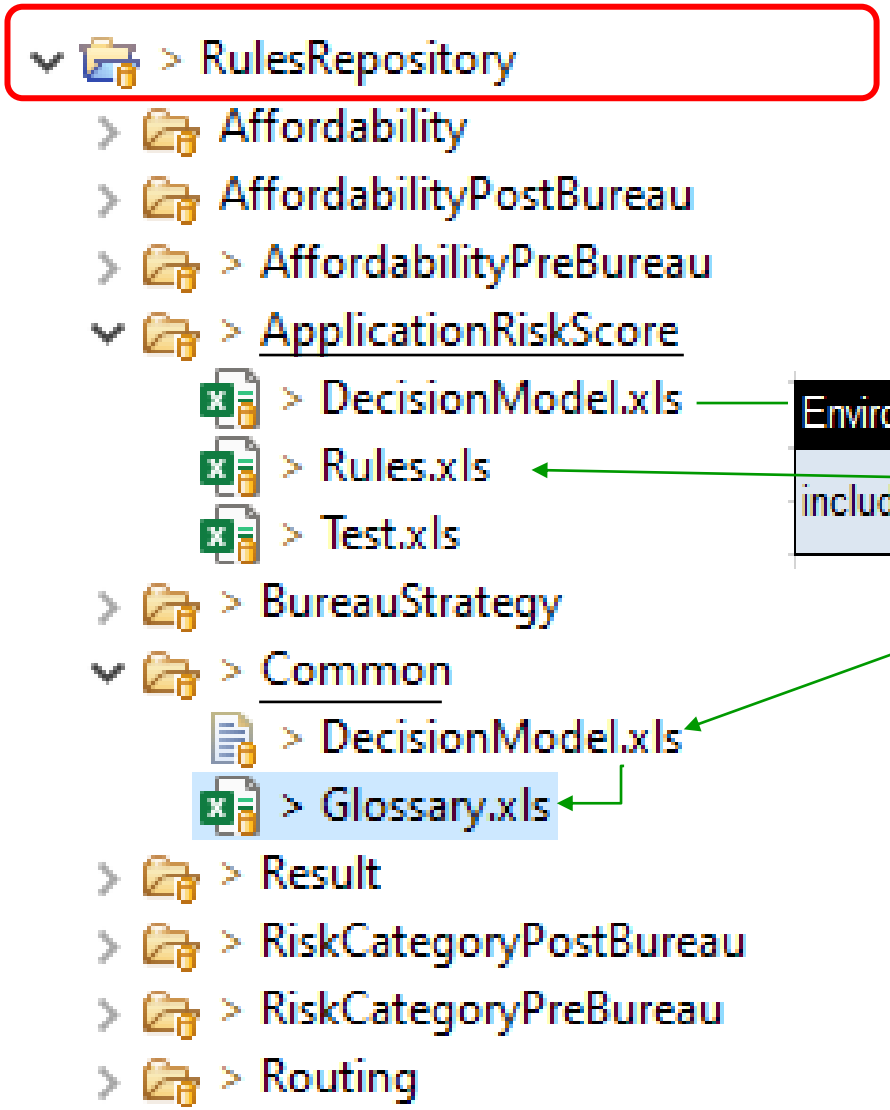
- **Build the Decision Model and Execute Test Cases** (click on *"test.bat"*)
- **Execution Results:**

Decision Table: Rule# (Cells)	Executed Rule	Variables and Values
ApplicationRiskScore: 1 (B5:F5)	**THEN** 'Application Risk Score' = 0	Application Risk Score=0
ApplicationRiskScore: 6 (B10:F10)	**IF** 'Age' >=50 **THEN** 'Application Risk Score' += 48	Age=51 Application Risk Score= {old:0, new:**48**}
ApplicationRiskScore: 8 (B12:F12)	**IF** 'Marital Status' Is M **THEN** 'Application Risk Score' += 45	Marital Status=M Application Risk Score= {old:48, new:**93**}
ApplicationRiskScore: 11 (B15:F15)	**IF** 'Employment Status' Is EMPLOYED **THEN** 'Application Risk Score' += 45	Employment Status=EMPLOYED Application Risk Score= {old:93, new:**138**}

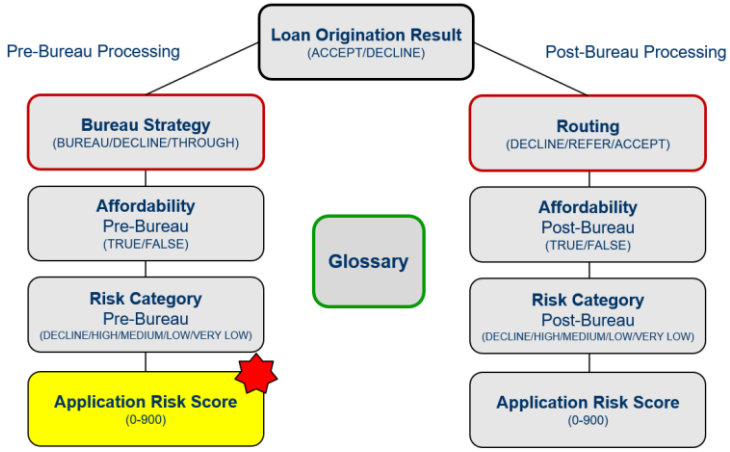
How is Rules Repository organized?

- One main folder “RulesRepository”
- Every decision model has a separate sub-folder
- The sub-folder “Common” keeps the Common Glossary in the file “*Common/Glossary.xls*”
- Each decision model has the file “*DecisionModel.xls*” with the table “*Environment*” that describes its structure
- Lets’ look at the Repository from a File Manager

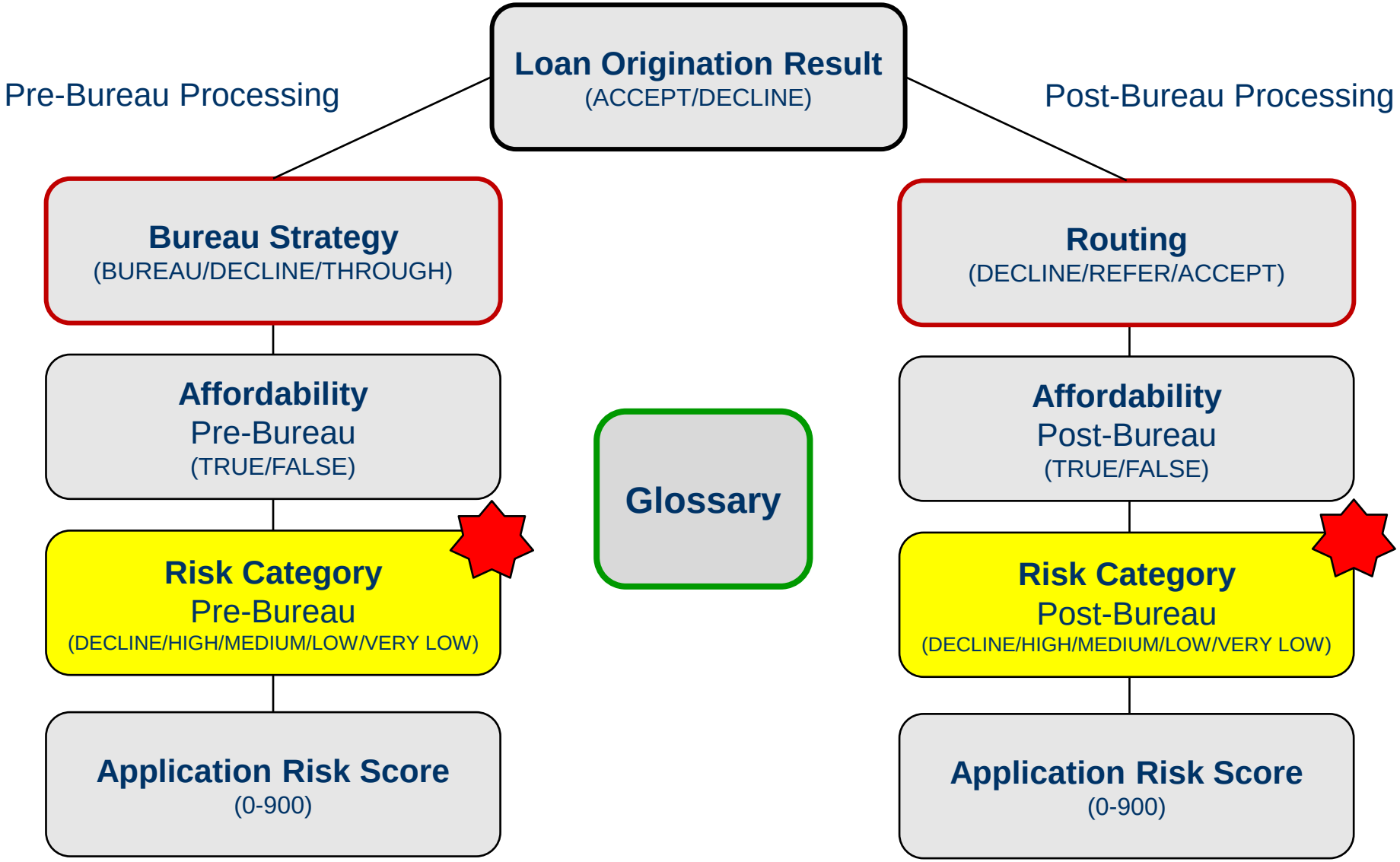
Rules Repository with multiple inter-connected decision models



Environment	
include	Rules.xls
	../Common/DecisionModel.xls



Decision Models “RiskCategoryPreBureau” / “RiskCategoryPostBureau”



Risk Category

Risk Category (Pre-Bureau)

DecisionTable PreBureauRiskCategory			
Condition		If	Then
Existing Customer		Application Risk Score	Risk Category
Is	TRUE	<100	HIGH
Is		[100..120)	MEDIUM
Is		[120..130]	LOW
Is		>130	VERY LOW
Is	FALSE	<80	DECLINE
Is		[80..90)	HIGH
Is		[90..110]	MEDIUM
Is		>110	LOW

Risk Category (Post-Bureau)

DecisionTable PostBureauRiskCategory				
Condition		If	If	Then
Existing Customer		Application Risk Score	Credit Score	Risk Category
Is	TRUE	< 120	<590	HIGH
Is			[590..610]	MEDIUM
Is			>610	LOW
Is		[120..130]	<600	HIGH
Is			[600..625]	MEDIUM
Is			>625	LOW
Is		> 130		VERY LOW
Is	FALSE	<=100	<580	HIGH
Is			[580..600]	MEDIUM
Is			>600	LOW
Is		>100	<590	HIGH
Is			[590..615]	MEDIUM
Is			>615	LOW

The Variable Names are the same, the logic is different!

We'll create two decision models

Decision Model “RiskCategoryPreBureau”

- Determines the goal “Risk Category” for pre-bureau processing
- Rules.xls:

DecisionTable PreBureauRiskCategory			
Condition		If	Then
Existing Customer		**Application Risk Score**	**Risk Category**
Is	TRUE	<100	HIGH
Is		[100..120)	MEDIUM
Is		[120..130]	LOW
Is		>130	VERY LOW
Is	FALSE	<80	DECLINE
Is		[80..90)	HIGH
Is		[90..110]	MEDIUM
Is		>110	LOW

Glossary is extended by:

- Existing Customer (Applicant)
- Risk Category (Application)

Environment	
include	Rules.xls
	../ApplicationRiskScore/DecisionModel.xls

Decision Model “RiskCategoryPostBureau”

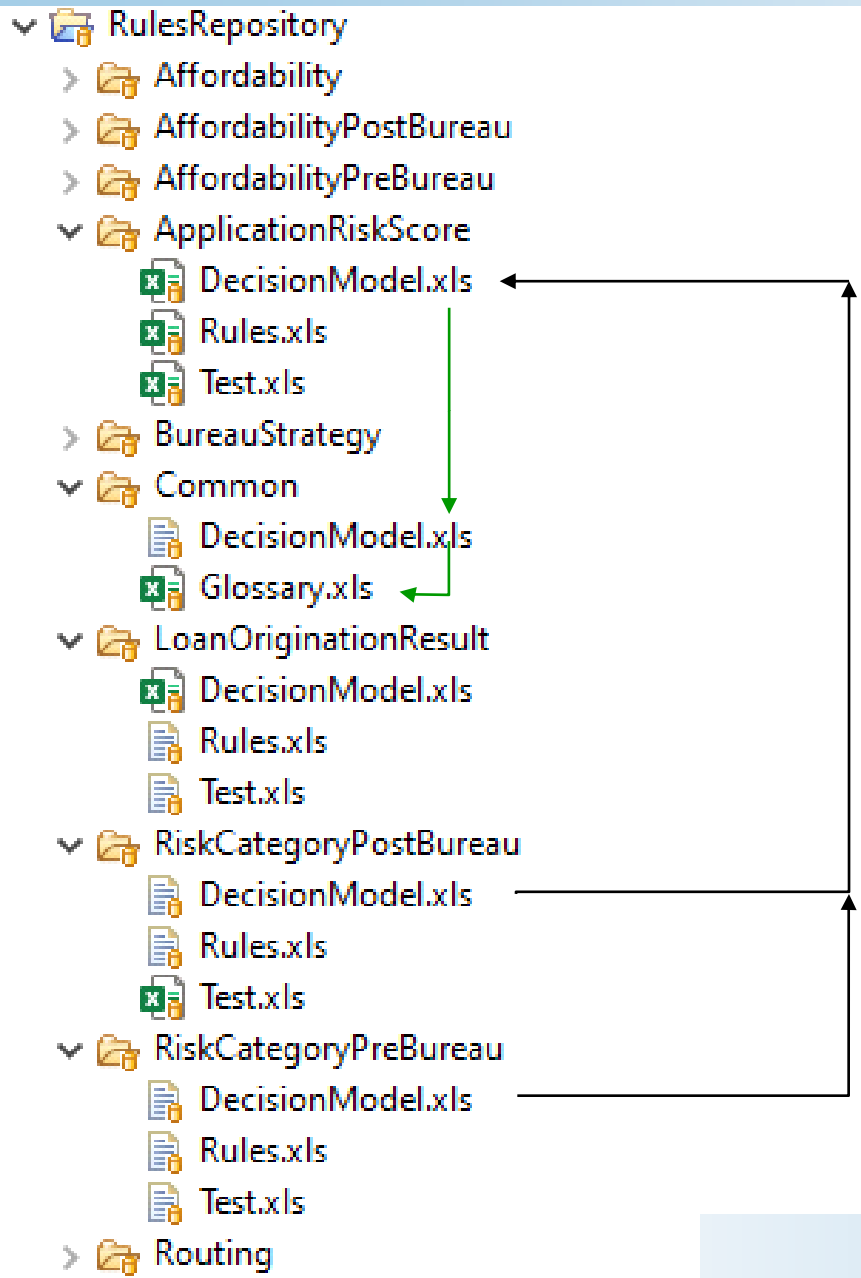
- **Determines the goal “Risk Category” for post-bureau processing**
- **Rules.xls:**

DecisionTable PostBureauRiskCategory				
Condition		If	If	Then
Existing Customer		**Application Risk Score**	**Credit Score**	**Risk Category**
Is	TRUE	< 120	<590	HIGH
Is			[590..610]	MEDIUM
Is			>610	LOW
Is		[120..130]	<600	HIGH
Is			[600..625]	MEDIUM
Is			>625	LOW
Is		> 130		VERY LOW
Is	FALSE	<=100	<580	HIGH
Is			[580..600]	MEDIUM
Is			>600	LOW
Is		>100	<590	HIGH
Is			[590..615]	MEDIUM
Is			>615	LOW

- Glossary is extended by:
  - Existing Credit Score
  - New Business Concept “BureauData”
- Decision tables “RiskCategory**Pre**Bureau” and “RiskCategory**Post**Bureau” share the same “Risk Category” calculated differently
- So, they cannot be used inside the same decision model

Environment	
include	Rules.xls
	../ApplicationRiskScore/DecisionModel.xls

Assembling Decision Models like LEGO-boxes



Decision Model “ApplicationRiskScore”

Environment	
include	Rules.xls
	../Common/DecisionModel.xls

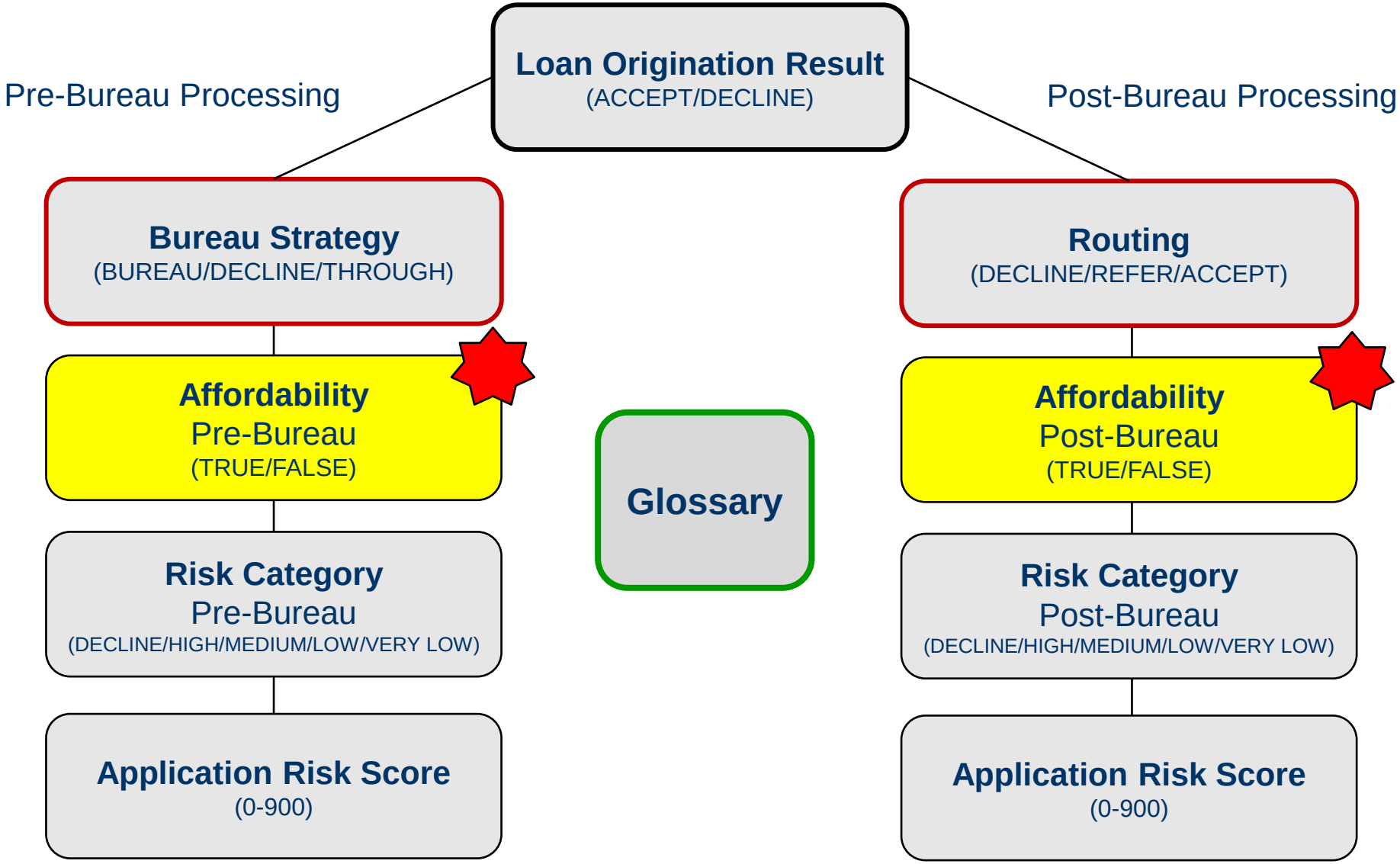
Decision Model “RiskCategoryPostBureau”

Environment	
include	Rules.xls
	../ApplicationRiskScore/DecisionModel.xls

Decision Model “RiskCategoryPreBureau”

Environment	
include	Rules.xls
	../ApplicationRiskScore/DecisionModel.xls

Decision Models “AffordabilityPreBureau” / “AffordabilityPostBureau”



Common Decision Rules for “Affordability”

Rules.xls:

DecisionTable Affordability

If	Then
Required Monthly Installment	**Affordability**
< Disposable Income * Credit Contingency Factor	TRUE
	FALSE

DecisionTable DisposableIncome

Action
Disposable Income
Monthly Income - (Monthly Repayments + Monthly Expenses)

DecisionTable RequiredMonthlyInstallment

Condition		Action
Product Type		**Required Monthly Installment**
Is	SPECIAL LOAN	PMT + 25.00
		PMT + 20.00

DecisionTable CreditContingencyFactor

Condition		Action
Risk Category		**Credit Contingency Factor**
Is One Of	HIGH,DECLINE	0.6
Is	MEDIUM	0.7
Is One Of	LOW, VERY LOW	0.8

DecisionTable PMT

Action
PMT
(Amount * Rate/12) / (1 - pow(1 +Rate/12,-Term))

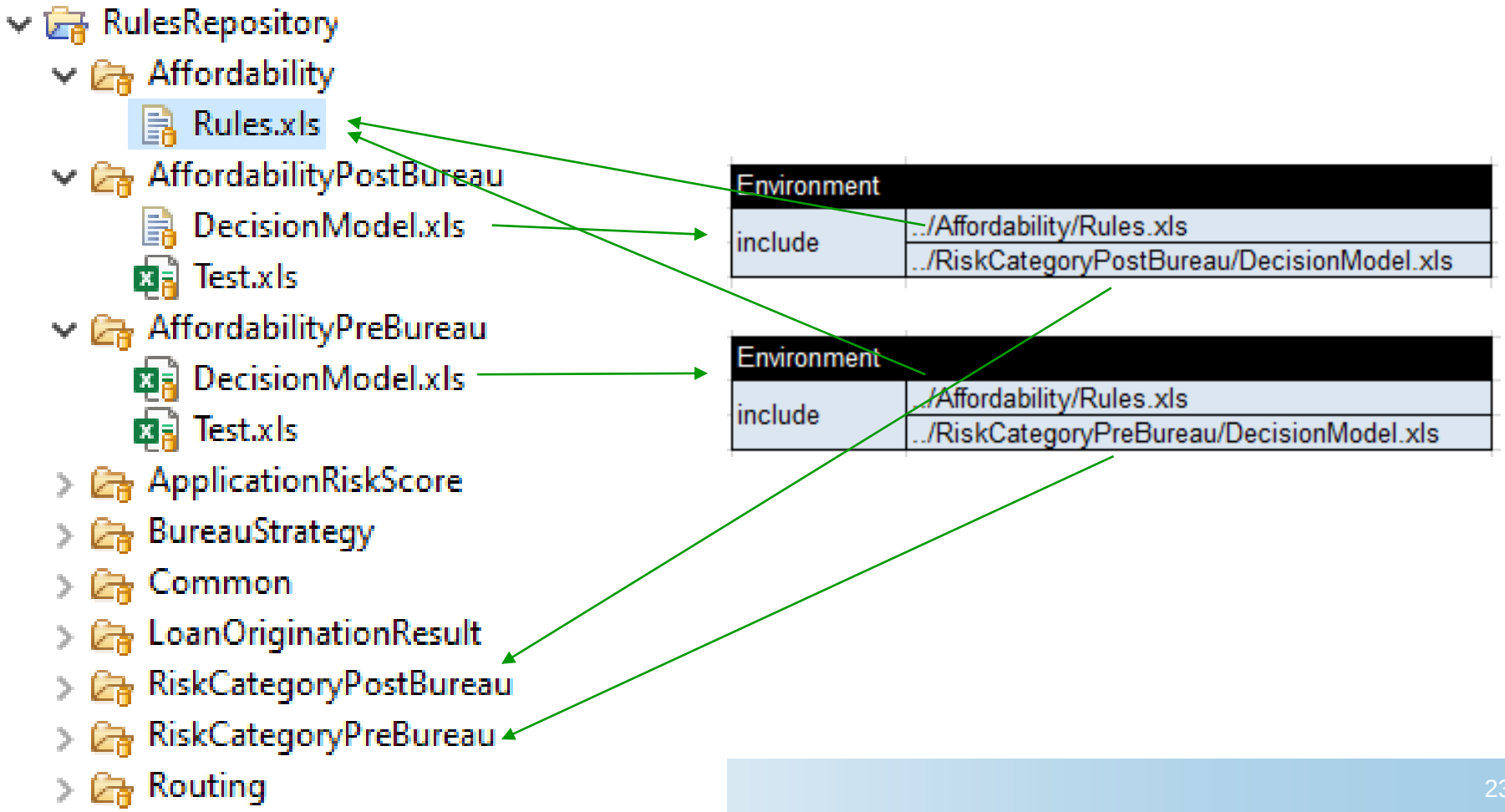
We need two Decision Models:

- Affordability**Pre**Bureau
- Affordability**Post**Bureau
- Same rules but different Environment tables

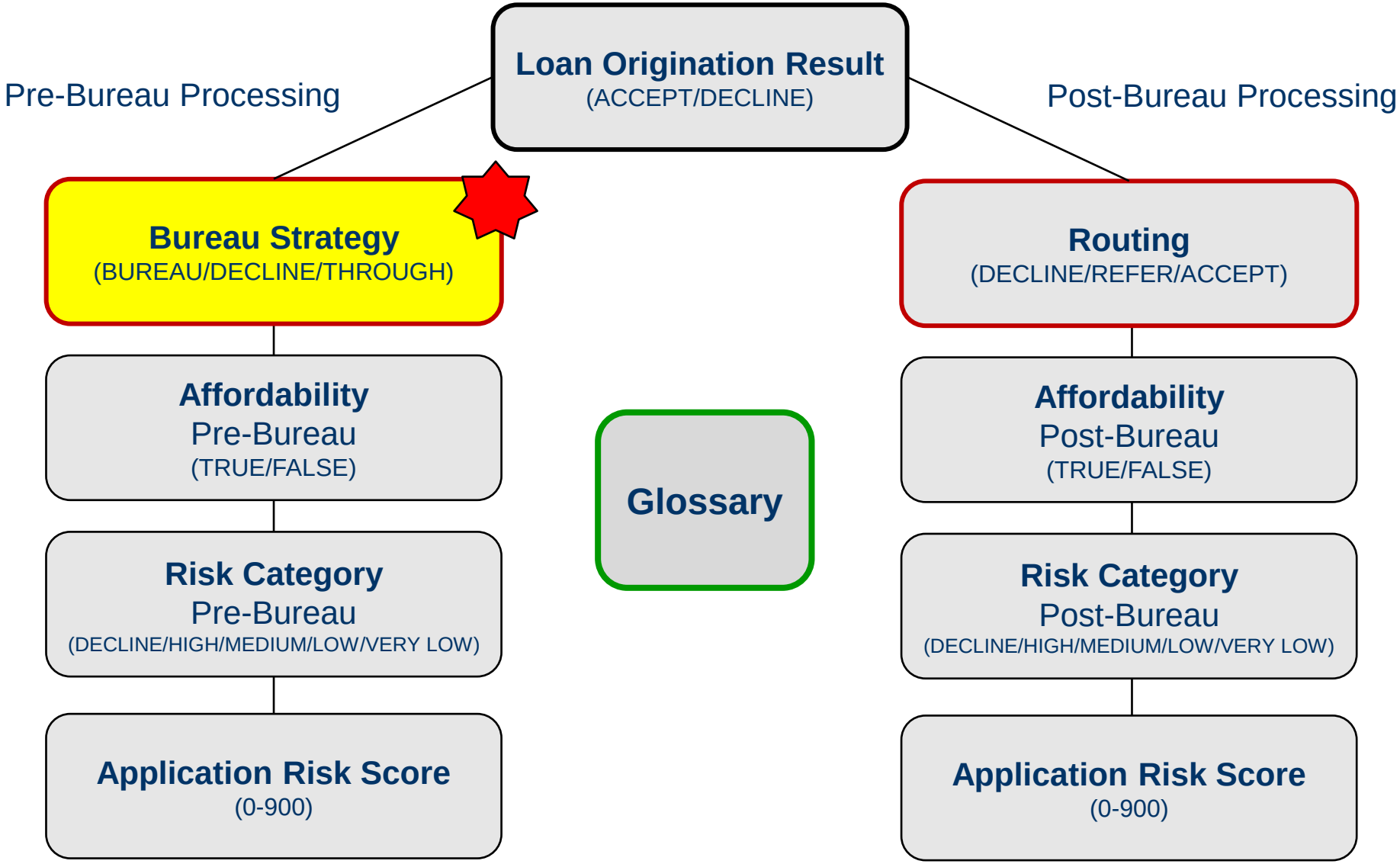
We will be in trouble if we try to manually define all inter-table relationships even using friendly diagrams. Rule engine should figure them out automatically!

Decision Models for “Affordability”

- We will place the above rules in the file “Affordability/Rules.xls”
- We will create two folders “AffordabilityPreBureau” and “AffordabilityPostBureau” with different Environment tables:



Decision Model “BureauStrategy”



Decision Model “BureauStrategy”

- Determines the goal “Bureau Strategy” for pre-bureau processing
- Rules.xls:

DecisionTable BureauStrategy				
Condition		Condition		Action
Eligibility		Bureau Call Type		Bureau Strategy
Is	INELIGIBLE			DECLINE
Is	ELIGIBLE	Is One Of	FULL, MINI	BUREAU
Is		Is	NONE	THROUGH

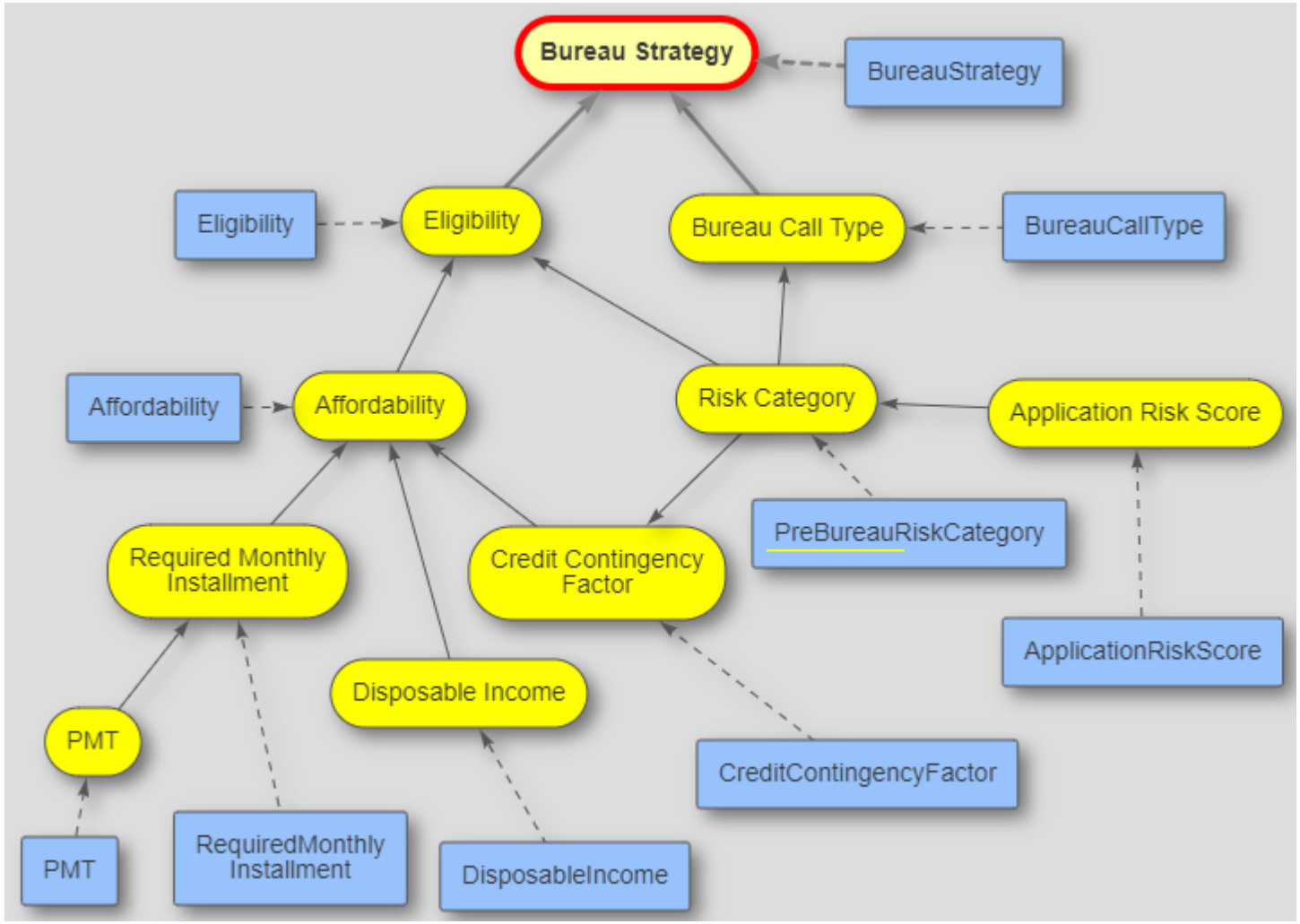
DecisionTable Eligibility			
If	If	If	Then
Risk Category	Affordability	Age	Eligibility
DECLINE	-	-	INELIGIBLE
-	FALSE		INELIGIBLE
-	-	<18	INELIGIBLE
-	-	-	ELIGIBLE

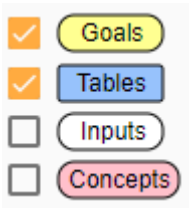
DecisionTable BureauCallType		
Condition		Action
Risk Category		Bureau Call Type
Is One Of	HIGH,MEDIUM	FULL
Is	LOW	MINI
Is One Of	VERY LOW, DECLINE	NONE

Environment	
include	Rules.xls
	../Affordability/PreBureau/DecisionModel.xls

Decision Model “BureauStrategy” – Decision Requirement Diagram

Here is an automatically generated DRD and Execution Path:





Execution path:

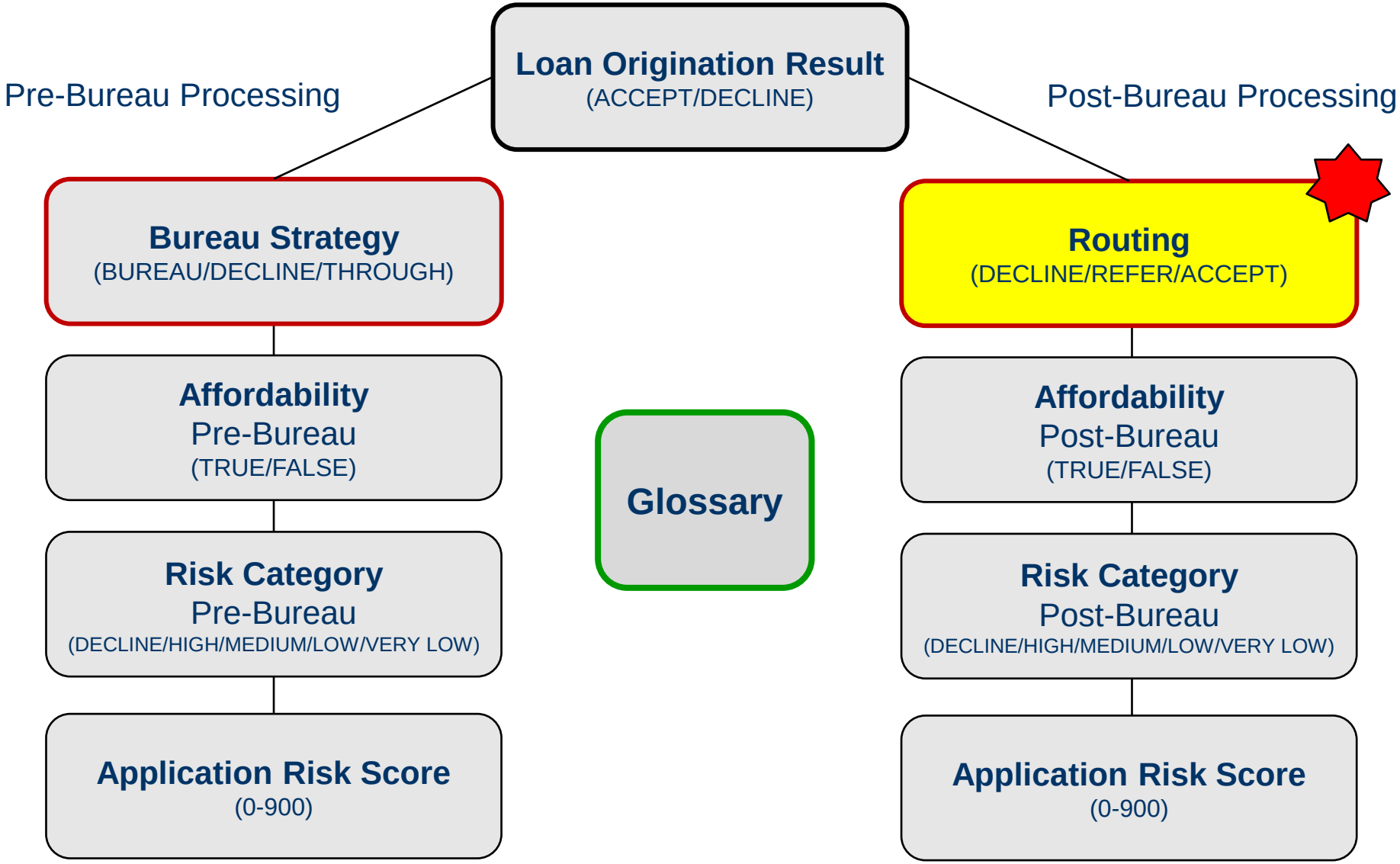
- ApplicationRiskScore
- PreBureauRiskCategory
- PMT
- RequiredMonthlyInstallment
- DisposableIncome
- CreditContingencyFactor
- Affordability
- Eligibility
- BureauCallType
- BureauStrategy

Bureau Strategy

Execution Results

Decision Table: Rule# (Cells)	Executed Rule	Variables and Values
ApplicationRiskScore: 1 (B5:F5)	THEN 'Application Risk Score' = 0	Application Risk Score=0
ApplicationRiskScore: 6 (B10:F10)	IF 'Age' >=50 THEN 'Application Risk Score' += 48	Age=51 Application Risk Score={old:0, new:48}
ApplicationRiskScore: 8 (B12:F12)	IF 'Marital Status' Is M THEN 'Application Risk Score' += 45	Marital Status=M Application Risk Score={old:48, new:93}
ApplicationRiskScore: 11 (B15:F15)	IF 'Employment Status' Is EMPLOYED THEN 'Application Risk Score' += 45	Employment Status=EMPLOYED Application Risk Score={old:93, new:138}
PreBureauRiskCategory: 4 (B8:E8)	IF 'Existing Customer' Is true AND 'Application Risk Score' >130 THEN 'Risk Category' = VERY LOW	Existing Customer=true Application Risk Score=138 Risk Category={old:?, new:VERY LOW}
PMT: 1 (B27:B27)	THEN 'PMT' = (Amount * Rate/12) / (1 - pow(1 +Rate/12,-Term))	PMT={old:0.0, new:3133.636546143113} Amount=100000 Rate=0.08 Term=36
RequiredMonthlyInstallment: 2 (B6:D6)	THEN 'Required Monthly Installment' = PMT + 20.00	Required Monthly Installment={old:0.0, new:3153.636546143113} PMT=3133.636546143113
DisposableIncome: 1 (B5:B5)	THEN 'Disposable Income' = Monthly Income - (Monthly Repayments + Monthly Expenses)	Disposable Income={old:0.0, new:4500.0} Monthly Income=10000.0 Monthly Repayments=2500.0 Monthly Expenses=3000.0
CreditContingencyFactor: 3 (B7:D7)	IF 'Risk Category' Is One Of LOW, VERY LOW THEN 'Credit Contingency Factor' = 0.8	Risk Category=VERY LOW Credit Contingency Factor={old:0.0, new:0.8}
Affordability: 1 (B5:C5)	IF 'Required Monthly Installment' < Disposable Income * Credit Contingency Factor THEN 'Affordability' = true	Required Monthly Installment=3153.636546143113 Credit Contingency Factor=0.8 Disposable Income=4500.0 Affordability={old:false, new:true}
Eligibility: 4 (B8:E8)	THEN 'Eligibility' = ELIGIBLE	Eligibility={old:INELIGIBLE, new:ELIGIBLE}
BureauCallType: 3 (B7:D7)	IF 'Risk Category' Is One Of VERY LOW, DECLINE THEN 'Bureau Call Type' = NONE	Risk Category=VERY LOW Bureau Call Type=NONE
BureauStrategy: 3 (B7:F7)	IF 'Eligibility' Is ELIGIBLE AND 'Bureau Call Type' Is NONE THEN 'Bureau Strategy' = THROUGH	Eligibility=ELIGIBLE Bureau Call Type=NONE Bureau Strategy={old:DECLINE, new:THROUGH}

Decision Model “Routing”



Decision Model “Routing”

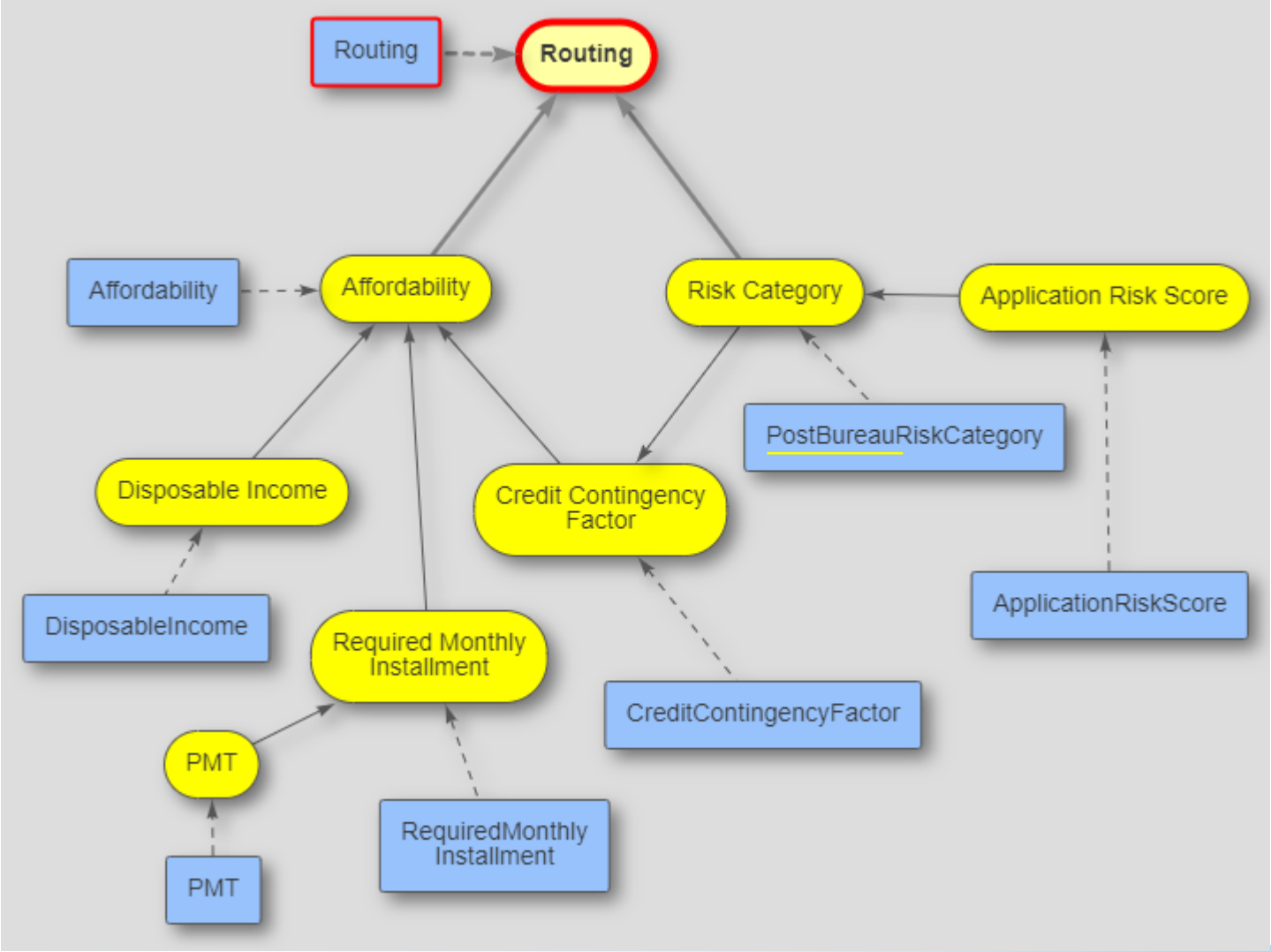
- Determines the goal “Routing” for post-bureau processing
- Rules.xls:

DecisionTable Routing				
If	If	If	If	Then
Affordability	**Bankrupt**	**Risk Category**	**Credit Score**	**Routing**
FALSE				DECLINE
TRUE	TRUE			DECLINE
TRUE	FALSE	DECLINE		DECLINE
TRUE		HIGH		REFER
TRUE			<580	REFER
TRUE			>=580	ACCEPT

Environment	
include	Rules.xls
	../Affordability/PostBureau/DecisionModel.xls

Decision Model “Routing” – Decision Requirement Diagram

Here is an automatically generated DRD and Execution Path:



Execution path:

- ApplicationRiskScore
- PostBureauRiskCategory
- PMT
- RequiredMonthlyInstallment
- DisposableIncome
- CreditContingencyFactor
- Affordability
- Routing

Decision Model “Routing”

Execution Results

Decision Table: Rule# (Cells)	Executed Rule	Variables and Values
PMT: 1 (B27:B27)	**THEN** 'PMT' = (Amount * Rate/12) / (1 - pow(1 +Rate/12,-Term))	PMT={old:0.0, new:**3133.6365378940145**} Amount=100000 Rate=0.07999999821186066 Term=36
RequiredMonthlyInstallment: 2 (B6:D6)	**THEN** 'Required Monthly Installment' = PMT + 20.00	Required Monthly Installment= {old:0.0, new:**3153.6365378940145**} PMT=3133.6365378940145
DisposableIncome: 1 (B5:B5)	**THEN** 'Disposable Income' = Monthly Income - (Monthly Repayments + Monthly Expenses)	Disposable Income={old:0.0, new:**4500.0**} Monthly Income=10000.0 Monthly Repayments=2500.0 Monthly Expenses=3000.0
ApplicationRiskScore: 1 (B5:F5)	**THEN** 'Application Risk Score' = 0	Application Risk Score=0
ApplicationRiskScore: 6 (B10:F10)	**IF** 'Age' >=50 **THEN** 'Application Risk Score' += 48	Age=51 Application Risk Score={old:0, new:**48**}
ApplicationRiskScore: 8 (B12:F12)	**IF** 'Marital Status' Is M **THEN** 'Application Risk Score' += 45	Marital Status=M Application Risk Score= {old:48, new:**93**}
ApplicationRiskScore: 11 (B15:F15)	**IF** 'Employment Status' Is EMPLOYED **THEN** 'Application Risk Score' += 45	Employment Status=EMPLOYED Application Risk Score= {old:93, new:**138**}
PostBureauRiskCategory: 7 (B11:F11)	**IF** 'Existing Customer' Is true **AND** 'Application Risk Score' > 130 **THEN** 'Risk Category' = VERY LOW	Existing Customer=true Application Risk Score=138 Risk Category={old:?, new:**VERY LOW**}
CreditContingencyFactor: 3 (B7:D7)	**IF** 'Risk Category' Is One Of LOW, VERY LOW **THEN** 'Credit Contingency Factor' = 0.8	Risk Category=VERY LOW Credit Contingency Factor= {old:0.0, new:**0.8**}
Affordability: 1 (B5:C5)	**IF** 'Required Monthly Installment' < Disposable Income * Credit Contingency Factor **THEN** 'Affordability' = true	Required Monthly Installment=3153.6365378940145 Credit Contingency Factor=0.8 Disposable Income=4500.0 Affordability={old:false, new:**true**}
Routing: 6 (B10:F10)	**IF** 'Affordability' Is true **AND** 'Bankrupt' Is false **AND** 'Credit Score' >=580 **THEN** 'Routing' = ACCEPT	Affordability=true Bankrupt=false Credit Score=600 Routing={old:?, new:**ACCEPT**}

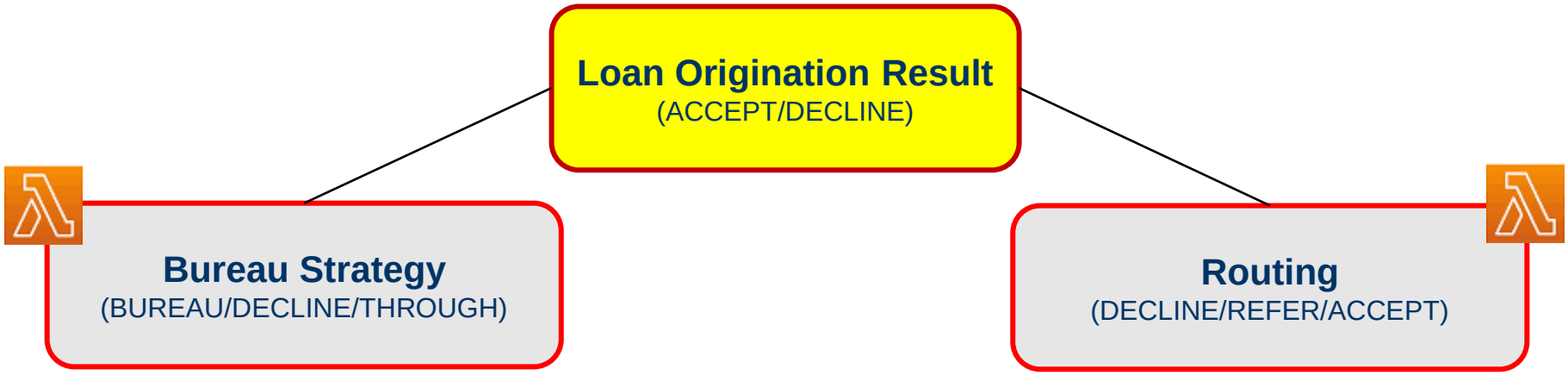
Common Glossary

Glossary.xls

Glossary glossary

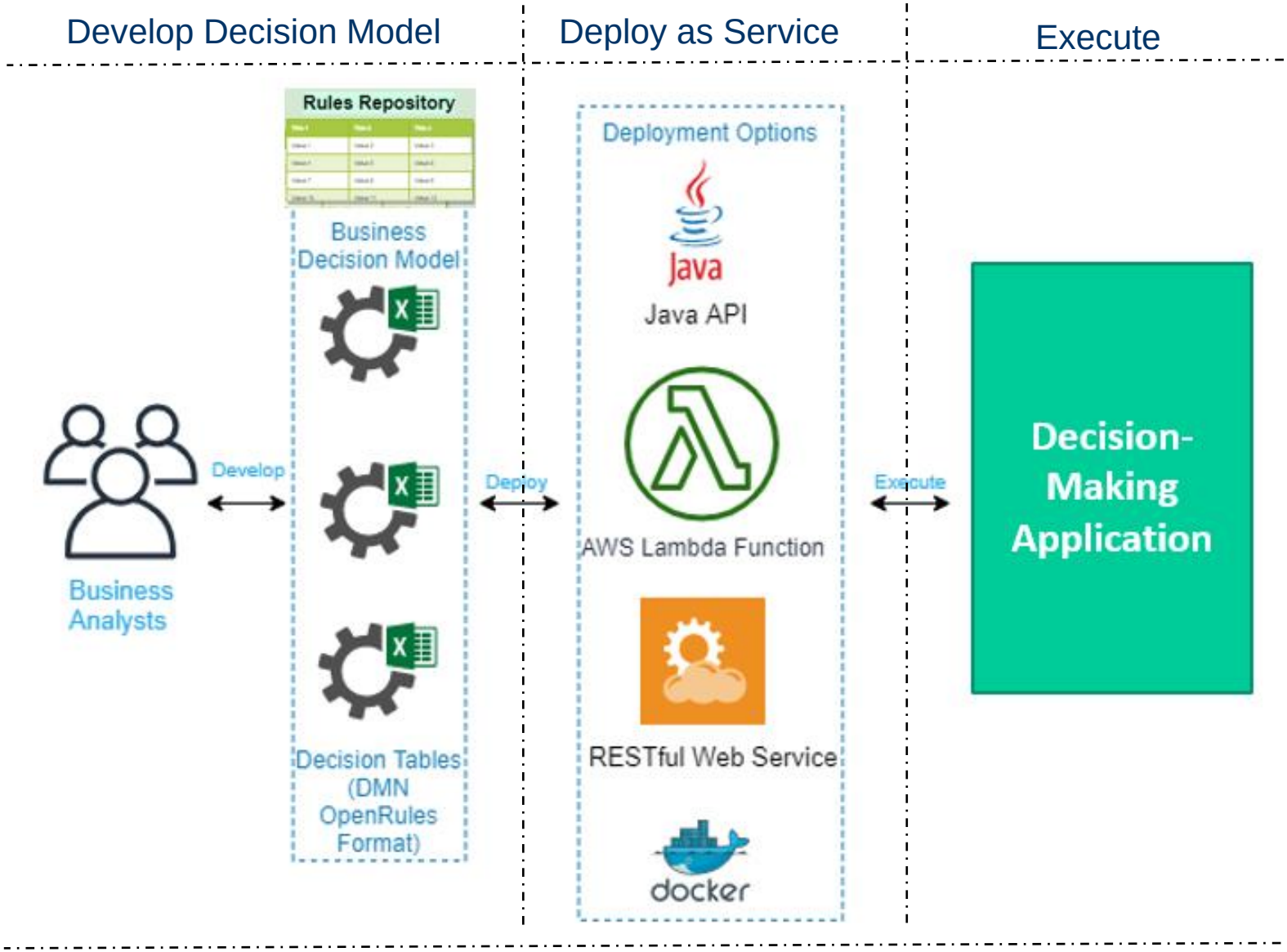
Variable	Business Concept	Attribute	Type	Domain
Name	Applicant	fullName	String	1-150
Age		age	Integer	
Marital Status		maritalStatus	String	SINGLE,MARRIED,OTHER
Employment Status		employmentStatus	String	EMPLOYED. UNEMPLOYED,RETIRED,OTHER
Monthly Income		monthlyIncome	Double	0-5000000
Monthly Repayments		monthlyRepayments	Double	0-5000000
Monthly Expenses		monthlyExpenses	Double	0-5000000
Existing Customer		existingCustomer	Boolean	TRUE,FALSE
Id	Application	id	String	
Application Risk Score		applicationRiskScore	Integer	0-900
Risk Category		riskCategory	String	DECLINE,HIGH,MEDIUM,LOW,VERY LOW
PMT		pmt	Double	0-5000000
Required Monthly Installment		requiredMonthlyInstallment	Double	0-1000000
Disposable Income		disposableIncome	Double	0-5000000
Credit Contingency Factor		creditContingencyFactor	Double	0-1
Affordability		affordability	Boolean	TRUE,FALSE
Eligibility		eligibility	String	INELIGIBLE,ELIGIBLE
Bureau Call Type		bureauCallType	String	FULL,MINI,NONE
Bureau Strategy		bureauStrategy	String	DECLINE,BUREAU,THROUGH
Routing		routing	String	DECLINE,REFER,ACCEPT
Loan Origination Result		loanOriginationResult	String	DECLINE,REFER,ACCEPT
Product Type	RequestedProduct	productType	String	STANDARD LOAN, SPECIAL LOAN
Amount		amount	Integer	1000-5000000
Rate		rate	Double	0.0 - 25.0
Term		term	Integer	36-360
Bureau Name	BureauData	bureauName	String	
Bankrupt		bankrupt	Boolean	TRUE,FALSE
Credit Score		creditScore	Integer	0-999

Deployment and Orchestration of Decision Services



- We cannot assemble decision model for “Loan Origination Result” using include-statements inside Environment table because decision models BureauStrategy and Routing share the same variables “Risk Category” and “Affordability”
- But we can invoke them as independent Decision Services, e.g. deployed as AWS Lambda functions

Deploying Decision Models



Decision Models => Decision Microservices

- **OpenRules supports one-click deployment of decision models on cloud as AWS Lambda functions**
  - **Deploy**: click on the standard file "**deployLambda.bat**"
  - **Test**: click on the automatically generated "**testLambda.bat**"
- **We deploy two main AWS lambda functions:**
  - BureauStrategy

```
Invoke URL: https://bfsu86u7u6.execute-api.us-east-1.amazonaws.com/test/bureau-strategy
```

  - Routing

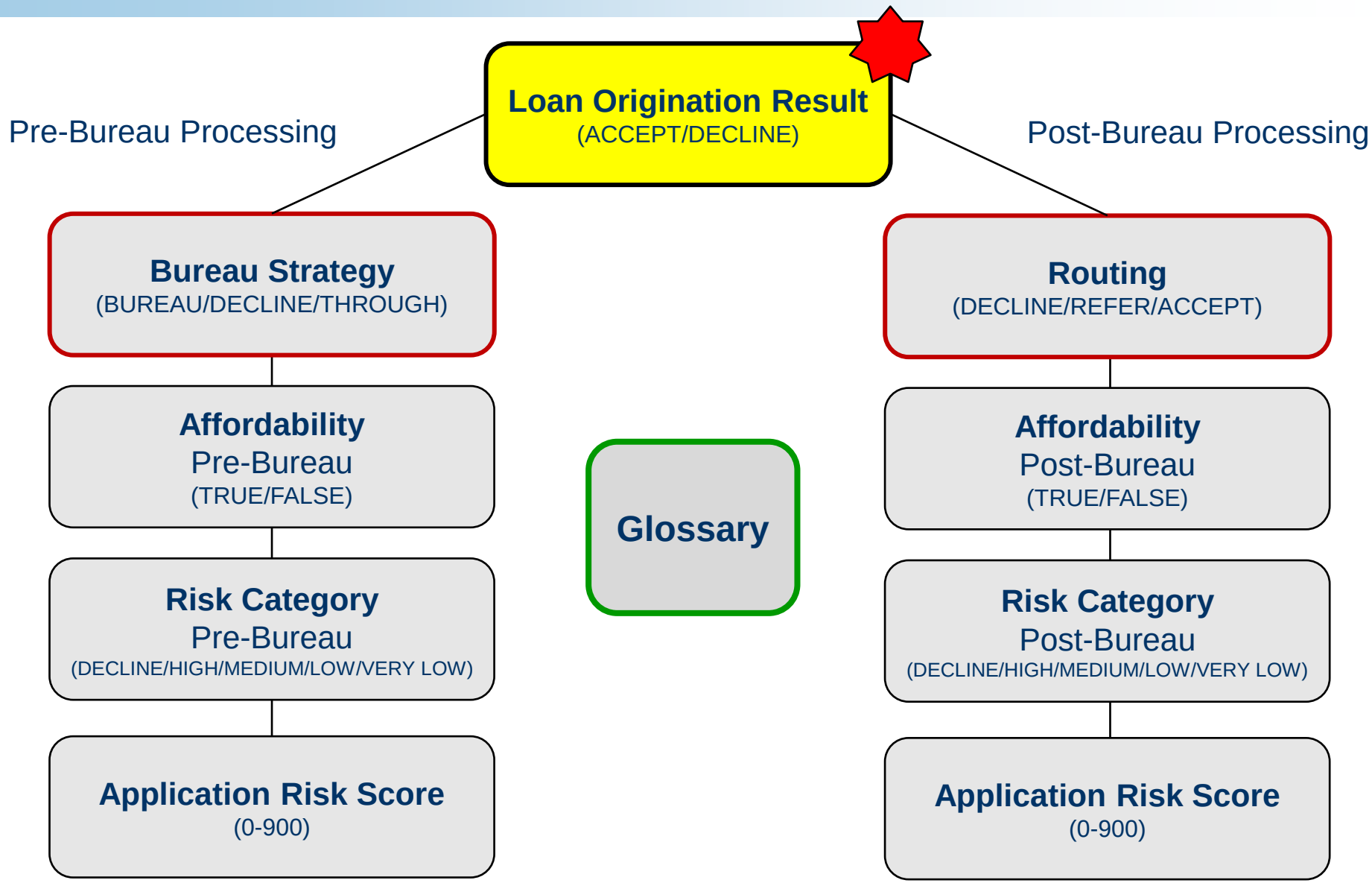
```
Invoke URL: https://vt72lhm6na.execute-api.us-east-1.amazonaws.com/test/routing
```

- **Test deployed decision services**

```
Running tests for DecisionModelBureauStrategy decision service at https://bfsu8
6u7u6.execute-api.us-east-1.amazonaws.com/test/bureau-strategy

Running test suite  'testCases'

Running test Test 1
Test 'Test 1' - OK. Roundtrip time 1680 ms. Rules Execution time 0.388547 ms.

Running test Test 2
Test 'Test 2' - OK. Roundtrip time 46 ms. Rules Execution time 0.379534 ms.

Running test Test 3
Test 'Test 3' - OK. Roundtrip time 44 ms. Rules Execution time 0.39773 ms.

All tests completed successfully
```

- **Now we are ready to orchestrate them**

Orchestration Decision Model “Loan Origination Result”



Orchestration Logic is Business Logic!

- We will build the Orchestration Decision Model “Result”

Decision LoanOriginationResult					
Condition		Condition		ActionExecute	Action
Bureau Strategy		**Routing**		**Execute**	**Loan Origination Result**
				BureauStrategyService	
Is	DECLINE				DECLINE
Is Not	DECLINE			RoutingService	
Is Not		Is	DECLINE		DECLINE
Is Not		Is	REFER		REFER
Is Not		Is	ACCEPT		ACCEPT

ActionExecute can execute
- internal decision tables or
- external decision services (!)

DecisionService decisionServices		
Service Name	**Service Type**	**Service Endpoint**
BureauStrategyService	REST	https://bfsu86u7u6.execute-api.us-east-1.amazonaws.com/test/bureau-strategy
RoutingService	REST	https://f7b53vlrel.execute-api.us-east-1.amazonaws.com/test/routing

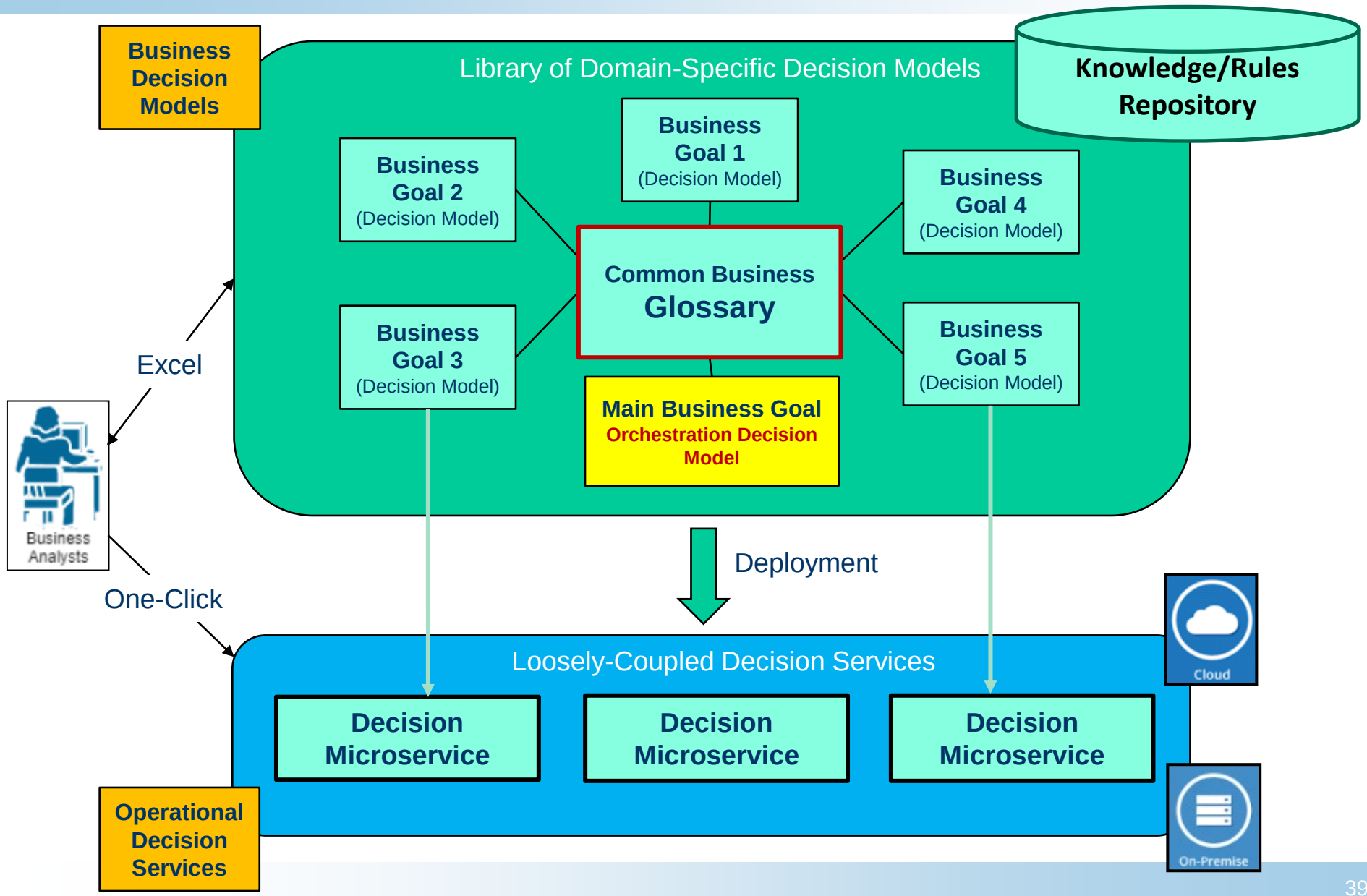
Test and Deploy the final decision model “Result”

- We can test decision model “Result” by a click on “*test.bat*”:

Decision Table: Rule# (Cells)	Executed Rule	Variables and Values
LoanOriginationResult: 1 (B17:G17)	**THEN** 'Execute' = BureauStrategyService	
LoanOriginationResult: 3 (B19:G19)	**IF** 'Bureau Strategy' Is Not DECLINE **THEN** 'Execute' = RoutingService	Bureau Strategy=THROUGH
LoanOriginationResult: 6 (B22:G22)	**IF** 'Bureau Strategy' Is Not DECLINE **AND** 'Routing' Is ACCEPT **THEN** 'Loan Origination Result' = ACCEPT	Bureau Strategy=THROUGH Routing=ACCEPT Loan Origination Result= {old:?, new:**ACCEPT**}

- Now we can also deploy and test the top-level decision model “Loan Origination Result” as an AWS Lambda function

Summary: We've created a Library of executable Decision Models and Decision Microservices (without programming)



QnA

jacobfeldman@openrules.com

support@openrules.com

+1 732 993 3131

www.OpenRules.com