Good Old UServ Product Derby in the Brave New World of Decision Management

Nov 4, 2015

DECISION MANAGEMENT SOLUTIONS

OPEN RULES

Presenters

- ## James Taylor
  - CEO of Decision Management Solutions
  - I work with clients to improve their business by applying technology to automate & improve decisions

- ## Dr Jacob Feldman
  - Founder & CTO of OpenRules
  - Hands-on developer helping OpenRules customers to create decisioning systems using BR, ML, and optimization technologies

Agenda

- Quick introduction to DMN

- UServ revisited

- Lessons from the DMCommunity submissions

- A more complete example

- Some thoughts on DMN
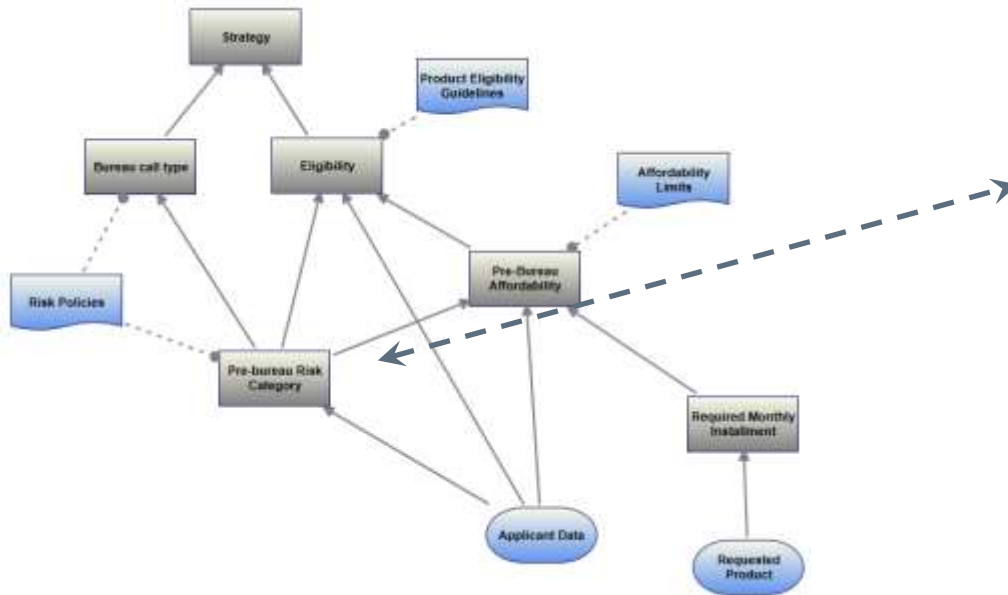
- Questions

Decision Model and Notation

Decision Modeling and Notation (DMN)

- "… provide a common notation that is readily understandable by all business users… DMN creates a standardized bridge for the gap between the business decision design and decision implementation."

- OMG Specification – a peer to BPMN, CMMN

- Decision Management Solutions, IBM, Oracle, TIBCO, FICO, Escape Velocity, KUL, Model Systems, KPI, Visumpoint

- DMN 1.0 Approved Q4-2014

- DMN 1.1 Revision under way
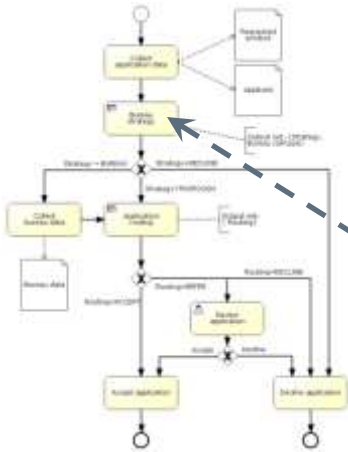
Two Layers of Detail

Decision Requirements

Decision Logic

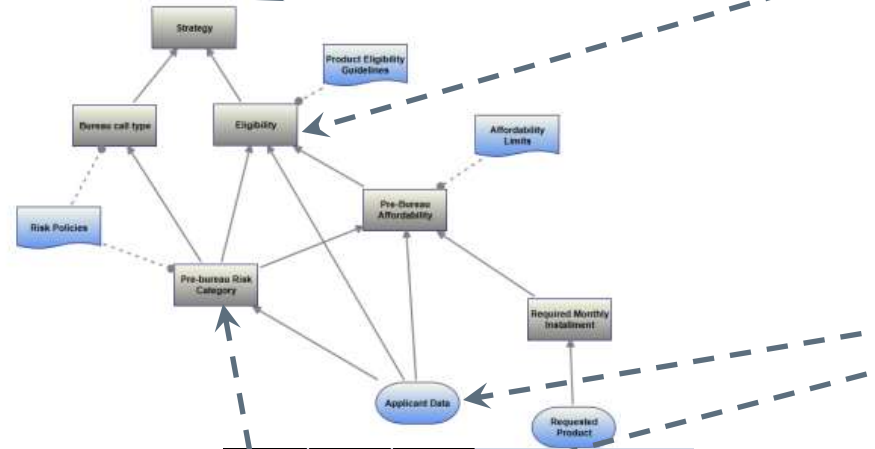| Pre-bureau risk category table			
UC	Existing Customer	Application Risk Score	Pre-Bureau Risk Category
1		<100	HIGH
2		[100..120[	MEDIUM
3		[120..130]	LOW
4	TRUE	>130	VERY LOW
5		<80	DECLINE
6		[80..90]	HIGH
7		[90..110]	MEDIUM
8	FALSE	>110	LOW

Process

Motivation

Organization

Decision
Logic

Data

Pre-bureau risk category table			
UC	Existing Customer	Application Risk Score	Pre-Bureau Risk Category
1		<100	HIGH
2		[100..120[	MEDIUM
3		[120..130]	LOW
4	TRUE	>130	VERY LOW
5		<80	DECLINE
6		[80..90]	HIGH
7		[90..110]	MEDIUM
8	FALSE	>110	LOW

Decision Models Manage Business Rules

Business Rules duplicated

Users must navigate unfamiliar repository

IT-managed Transformation

BRMS

Executable Rules
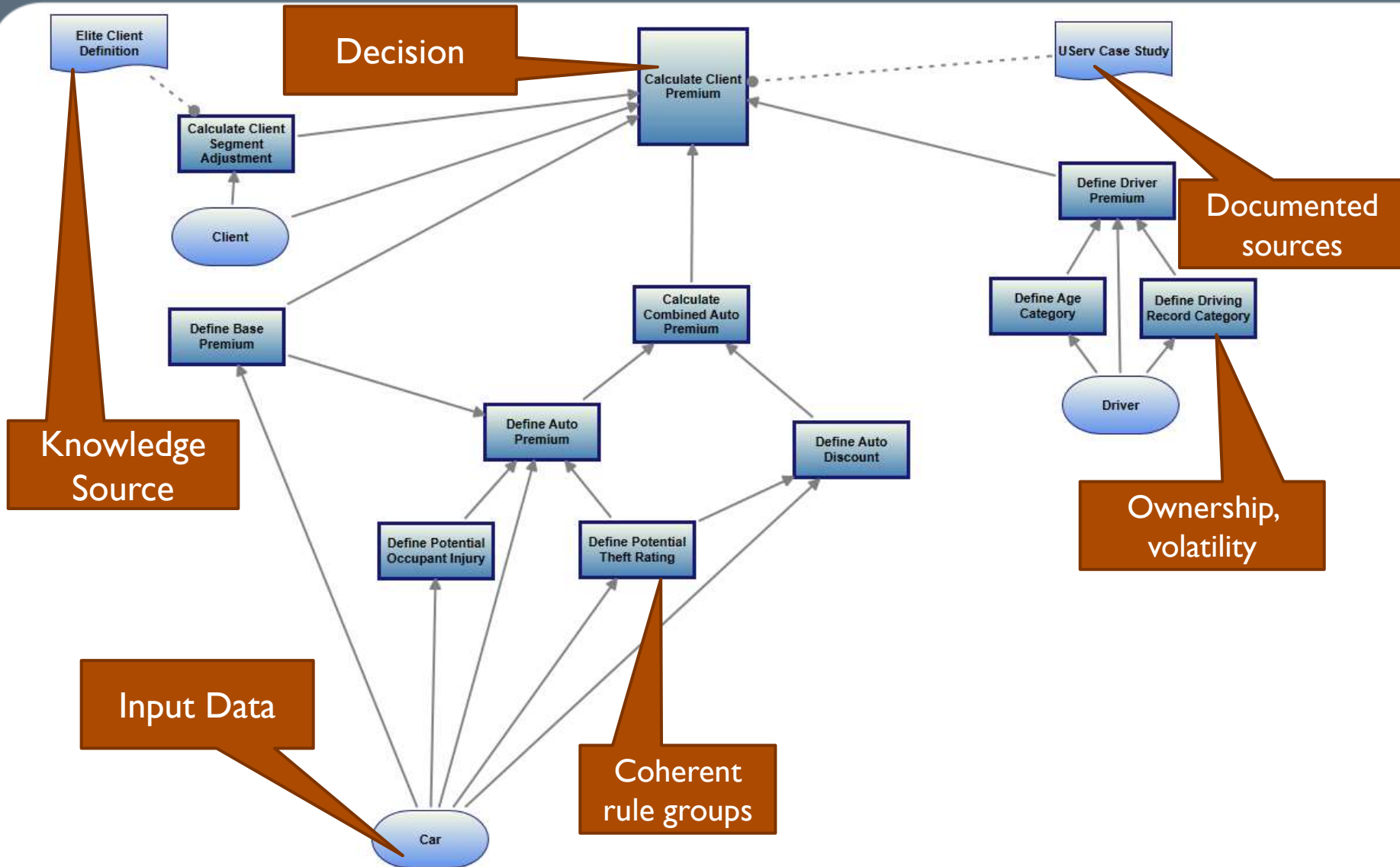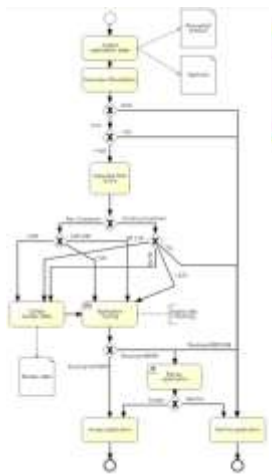
Over-complex process models

Multiple lists and documents about rules

BEFORE

AFTER

One set of Business Rules

Users link directly from familiar diagram

BRDMS

Executable Decisions
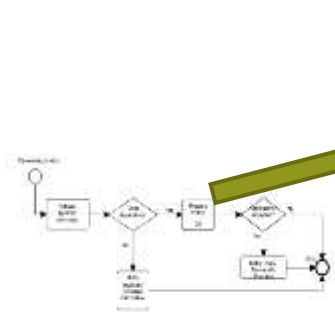
Simplified process models
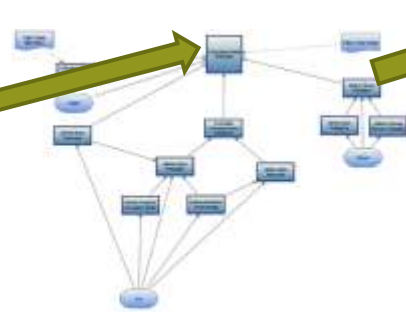
Clear decision models for business users

Many Use Cases

- Human Decision-making
  - Documenting human decision-making
  - Improving human decision-making with analytics
  - Training human decision-makers

- Requirements for automated Decision-making
  - Business rules discovery and analysis
  - Framing predictive analytics
  - Dashboard design

- Implementing automated Decision-making
  - Completely specifying business rules
  - Acting as a BRMS front-end
  - Orchestrating complex decisioning technology

DMN Support but Not Conformance

- In general there is a lot of interest in DMN

- 11 vendors already announced their support for DMN – see the DMN Tools Catalog

- Others will offer DMN support as Decision Tables were already popular in a majority of BRMSs

- But

  - Decision Requirements Diagrams support is much less widespread

  - Few if any 100% implementations are "conformant" with DMN 1.0

UServ Product Derby

UServ Product Derby

- Business Rules vendors competition popular at Business Rules Forums for 10 years

- Vendors had a chance to demonstrate their product capabilities. Potential users had an opportunity to compare different solutions

- This year UServ Derby was revived as the Decision Management Challenge

Decision Model: "Vehicle Insurance – UServ Product Derby" Solutions

10+ years ago BR vendors had a chance to demonstrate their capabilities at Business Rules Forums (now BBC) using the sa highly popular use case known as "UServ Product Derby". Th majority of BR vendors openly competed during these major events and potential users had an opportunity to compare different solutions. The use case dealt with vehicle insurance

www.DMCommunity.org

UServ Product Derby

- The Derby deals with vehicle insurance problems including eligibility and premium calculation policies for a hypothetical insurance company.

- The problem is described [here](#) and deals with the following rules:

Business Rules...
  Client Segmentation Business Rules ....
  Eligibility Business Rules..
    Automobile Eligibility ....
    Driver Eligibility..
    Eligibility Scoring..
  Pricing Business Rules...
    Auto Premiums ....
    Auto Discounts...
    Driver Premiums...
    Market Segment Discounts ....
    Base Premium ....
Scenarios ....
  Grandfathered Rule Sets ....
  Eligibility Within and Outside an Elite Client Relationship...

UServ Product Derby

- DMCommunity.org received 7 solution submissions:

Solutions:

- **Blueriq** – submitted by Maarten Schadd from www.blueriq.com
- **Corticon** – submitted by Michael Parish from Progress Corticon
- **IBM ODM** – submitted by Rafael Ortiguela from DECIDE
- **Sapiens** – submitted by Gil Segal from www.SapiensDecision.com
- **OpenRules** – submitted by Jacob Feldman from www.OpenRules.com
- **OpenL Tablets** – submitted by Yuliya Bastun from EIS Group
- **RuleML** – submitted by Matthias Tylkowski from http://binarypark.org
  (complete RuleML file)

- What did we learn?

Lessons Learned from DMCommunity.org

Key Observations From The Derby

1. Specification text rather than models

2. Many different decision table formats

3. Decision flows not decision requirement models

4. Interoperability is not happening yet

Specification Text Rather Than Models

- Original UServ description had some models
  - Data model
  - Process model
- All the business logic was written as text
  - Dozens of rules
  - Some grouped with like rules
  - Some mixed in large groups
  - Some shown as tables
- Actually better than most real-world specifications!
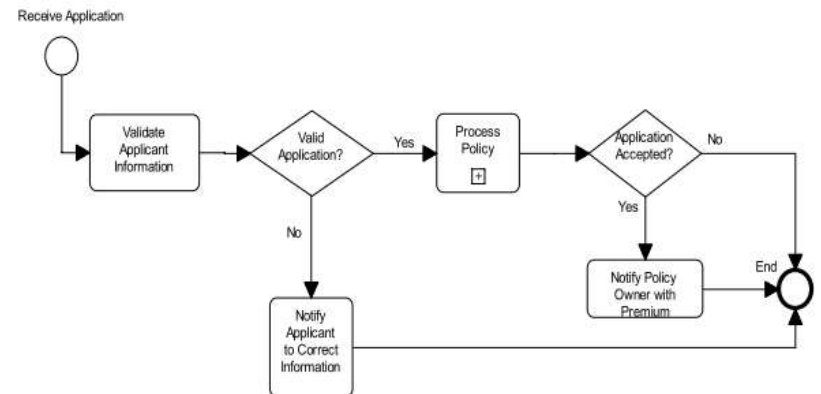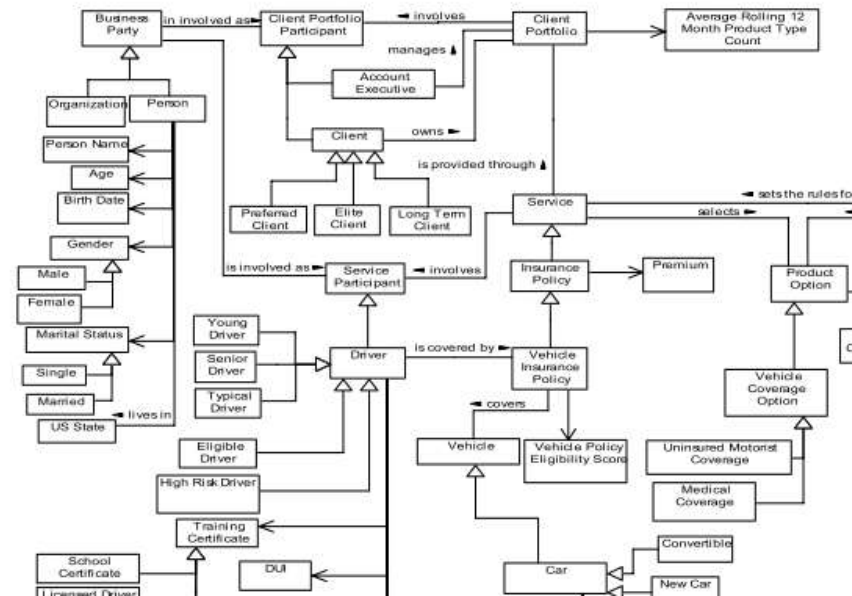
Many Different Decision Table Formats

- Each submission used its own decision table style
- Good News:
    - Decision tables are reasonably easy to read
    - Differences of presentation but not really of core concepts
    - Not impossible to compare and use different formats
- Bad News:
    - Makes it harder to understand how they would actually work
    - Writing them requires learning the specific tool involved
- Presentation consistency would improve understanding
- DMN focuses on tables that can be broadly understood
- Review: Different tables for "Define Driver Premium"

Define Driver Premium: Specification

Driver Premiums

For each driver on the policy:

If young driver and married and located in CA, NY or VA, then increase premium by $700.
If young driver and single and located in CA, NY or VA, then increase premium by $720.
If young driver and married and not located in CA, NY or VA, then increase premium by $300.
If young driver and single and not located in CA, NY or VA, then increase premium by $300.
If senior driver and located in CA, NY or VA, then increase premium by $500.
If senior driver and not located in CA, NY or VA, then increase premium by $200.

Driver is a Typical Driver is all of the following are true:
   o   Not a Young Driver
   o   Not a Senior Driver

If a Typical Driver, then increase premium by $0.
If a High Risk Driver, then increase premium by $1,000.
Raise the premium by $ 150 per accident

Define Driver Premium: DMN 1.0

Hit Policy (C+=Multi-Hit, Collect)

Aggregation Method (+)

Completeness (C)

Domains

Decision Table Define Driver Premium					
C+C	**Driver Age Category**	**Marital Status**	**State**	**Driving Record Category**	**Driver Premium**
	Young, Senior	Single, Married, Divorced		Low Risk, Mid Risk, High Risk	
1	Young	Married	CA,NY,VA		700
2		Single	CA,NY,VA		720
3		Married	CA,NY,VA		300
4		Single	CA,NY,VA		300
5	Senior		CA,NY,VA		500
6	Senior		CA,NY,VA		200
7				High Risk	1000
8					Number of Accidents * 150

FEEL Expression

Define Driver Premium: OpenRules

Hit Policy

DecisionTableMultiHit DefineDriverPremium									
Condition		Condition		Condition		Condition		Conclusion	
Driver Age Category		Marital Status		State		Driving Record Category		Driver Premium	
								=	0
Is		Is	Married	Is One Of	CA,NY,VA			+=	700
Is	Young	Is	Single					+=	720
Is		Is	Married	Is Not One Of	CA,NY,VA			+=	300
Is		Is	Single					+=	300
Is	Senior			Is One Of	CA,NY,VA			+=	500
Is	Senior			Is Not One Of	CA,NY,VA			+=	200
						Is	High Risk	+=	1000
								+=	::= $I{Number of accidents} * 150

Aggregation Methods

Define Driver Premium: Progress Corticon

Conditions	0	1	2	3	4	5	6	7	8	9
Driver age classification?		'Young'	'Young'	'Young'	'Young'	'Senior'	'Senior'	'Typical'	-	-
Driver marital status?		'Married'	'Single'	'Married'	'Single'	-	-	-	-	-
Driver residence state?		{'CA', 'NY', 'VA'}	{'CA', 'NY', 'VA'}	other	other	{'CA', 'NY', 'VA'}	other	-	-	-
Is the driver classified as high risk?		-	-	-	-	-	-	-	T	-
How many accidents has the driver...		-	-	-	-	-	-	-	-	> 0

Actions	◄									
Post Message(s)		✉	✉	✉	✉	✉	✉	✉	✉	✉
Initialize the driver premim to zero	☑									
Add this amount		700	720	300	300	500	200		1000	
Add this amount for each accident										150

Conditions	0	1			8	9
driver.driverAgeClass		'Young'			-	-
driverDetails.maritalStatus		'Married'			-	-
driverDetails.usState		{'CA', 'NY', 'VA'}			-	-
driver.isHighRisk		-			T	-
driver.numOfAccidents		-			-	> 0

Actions	◄					
Post Message(s)		✉			✉	✉
driver.participantPremium = 0	☑					
driver.participantPremium += cellValue		700			1000	
driver.participantPremium += cellValue * driver.numOfAccidents						150

Define Driver Premium: Sapiens

Driver Policy Annual Premium ▽

Is Incremented By ‖ ⬡ <u>Driver Annual Premium</u>

Driver Annual Premium ▽

Is Incremented By ‖ ∫ SUM($"High Risk Driver Annual Premium",$"Driver Demographic Annual Premium")

Driver Age Category ▽		Marital Status ▽		Driver Address State ▽		Driver Demographic Annual Premium ▽	
Is ‖ ν Young Driver		**Is** ‖ ν Married		**Is In** ‖ ν {CA,NY,VA}		**Is Incremented By** ‖ ν $700.00	
Is ‖ ν Young Driver		**Is** ‖ ν Single		**Is In** ‖ ν {CA,NY,VA}		**Is Incremented By** ‖ ν $720.00	
Is ‖ ν Young Driver		**Is In** ‖ ν {Married,Single}		**Is Not In** ‖ ν {CA,NY,VA}		**Is Incremented By** ‖ ν $300.00	
Is ‖ ν Senior Driver				**Is In** ‖ ν {CA,NY,VA}		**Is Incremented By** ‖ ν $500.00	
Is ‖ ν Senior Driver				**Is Not In** ‖ ν {CA,NY,VA}		**Is Incremented By** ‖ ν $200.00	
Is ‖ ν Typical Driver						**Is Incremented By** ‖ ν $0.00	

Define Driver Premium: Others

SimpleRules DoubleValue **DriverPremium** (DriverType driverType, MaritalStatus maritalStatus, String state)				
Driver Age	**Marital Status**	**State**		**Driver Premium**
Young Driver	Married	CA, NY, VA		$700
Young Driver	Single	CA, NY, VA		$720
Young Driver	Married			$300
Young Driver	Single			$300
Senior Driver		CA, NY, VA		$500
Senior Driver				$200
				$0

SimpleRules DoubleValue **DriverRiskPremium** (DriverRisk driverRisk)	
Driver Risk	**Risk Premium**
High Risk Driver	$1,000
Standard Risk Driver	$0

SimpleRules DoubleValue **AccidentPremium** ()
Per Accident Premium
$150

Driver.YoungDriver	TRUE			[]	
Driver.SeniorDriver	*			TRUE	
Driver.LocatedInCA_NY_VA	TRUE		FALSE	TRUE	FALSE
Driver.Married	TRUE	FALSE	*	*	*
Driver.PremiumDeltaVarious	700	720	300	500	200

Flow v Dependency

Rule or Decision Flows
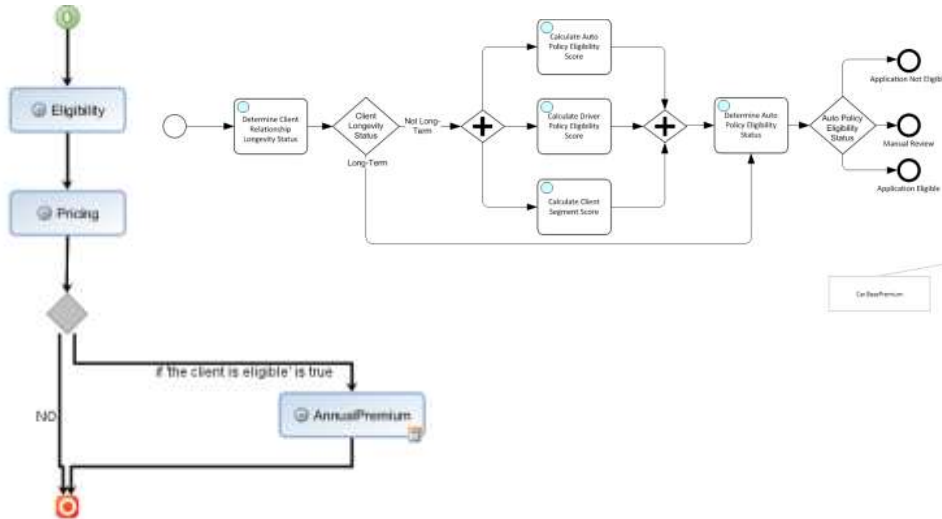
DMN Requirements Model
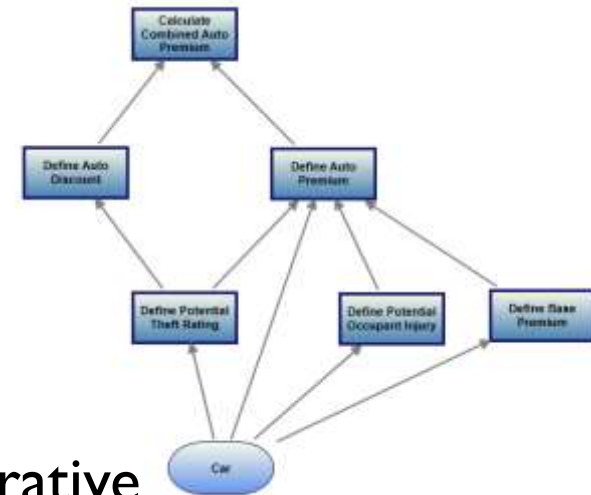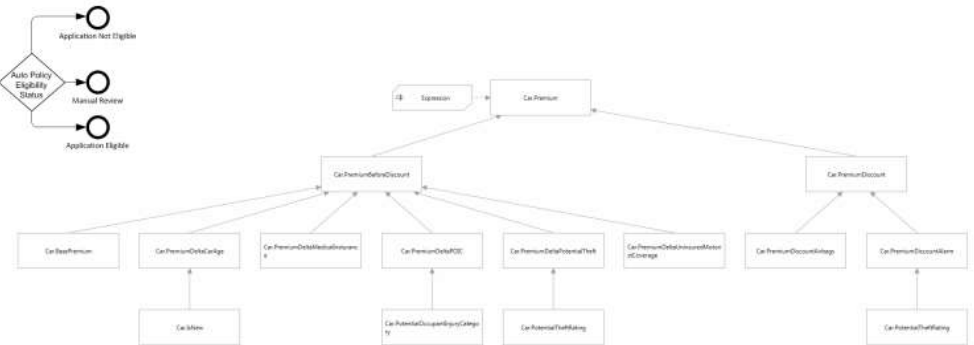
More declarative
Less implementation detail

Interoperability

- **All solutions**
    - Used very different decision table formats
    - Had varying approaches to the underlying logic
    - Took a different approach to defining the decision structure

- **At this point it looks impossible to even think about interoperability – the replacement of one UServ implementation with another**

- **DMN may offer a solution**
    - Common representation of Decision Tables
    - Standardized approach to decision structure
    - Common representation of Decision Requirements Diagrams
    - A standardized interchange format

Some other (minor) observations

- Many implementations use one diagram for everything but DMN allows reuse across diagrams

- Some implementations encode logic only in BKMs which more or less doubles the number of objects in a model
  - BKMs allow reuse of logic across multiple decisions
  - But decisions can be reused too

- Input Data: Fields or Entities?
  - Input Data objects can be individual terms OR business entities
  - Input Data as terms rapidly overwhelms diagrams

- Even though DMN does not specify how to build a common Glossary, one is clearly required

A DMN-based Executable UServ Decision Model

Demo

DecisionsFirst Modeler + OpenRules

- **DecisionsFirst Modeler**
  - Collaborative, social, cloud-based environment
  - Provides a simple yet precise definition of your Decision Requirements
  - Builds a map for your implementation

- **OpenRules®**
  - Open Source general purpose Business Rules and Decision Management System
  - Supports Executable Decision Models created by subject matter experts in Excel or Google Docs
  - Validates, deploys, manages, and effectively executes decision models

Multiple Diagrams for Perspective
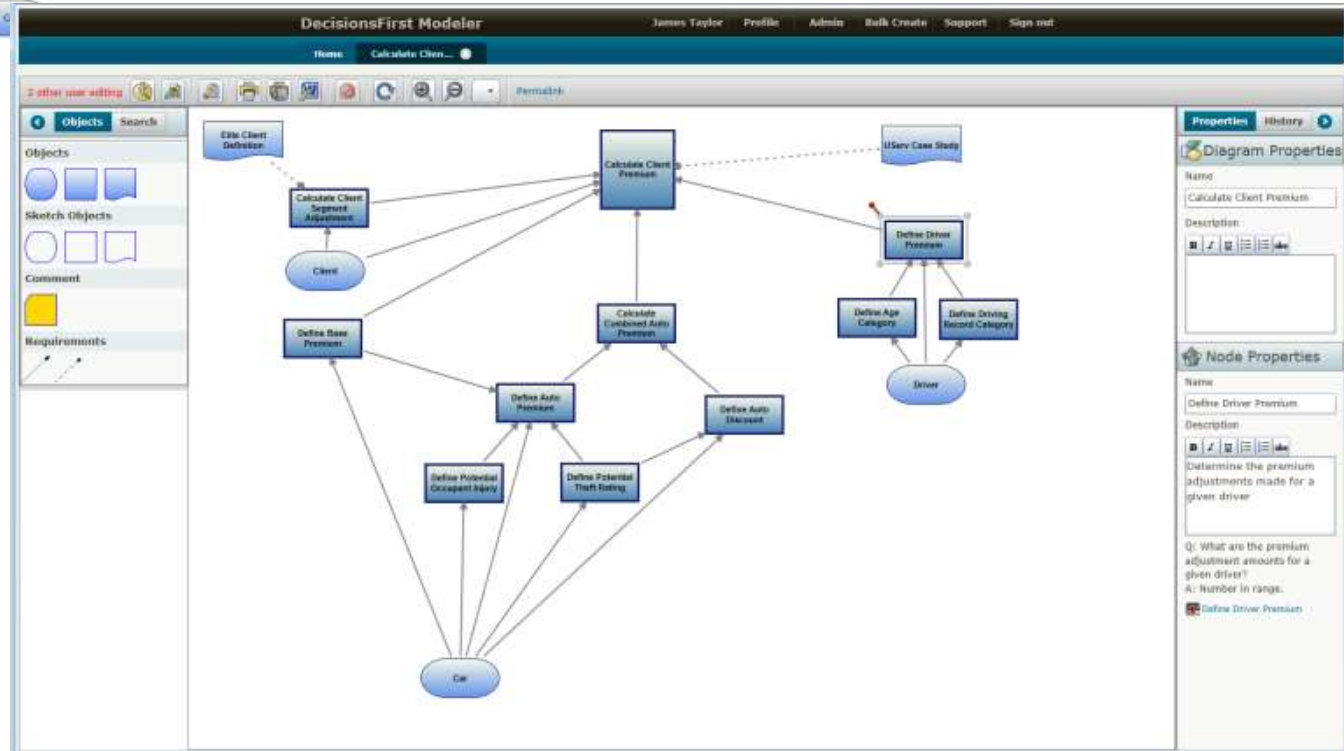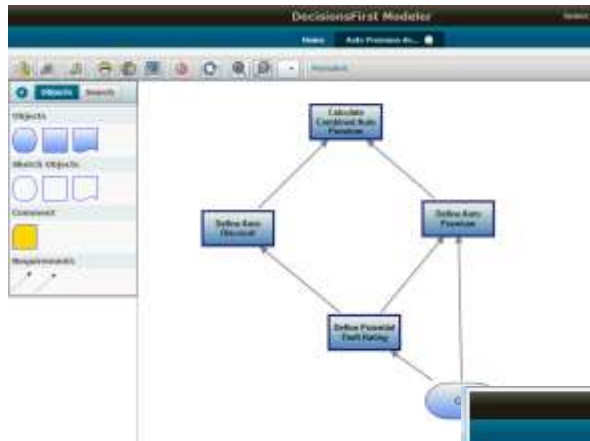
Additional Properties and Associations

Nearest Neighbors

Question and Allowed answers

Business Context

Implementation

Integration With OpenRules Tables

Improvements and additions to DMN

Explicitly Defined Business Glossary

- Most business rules/ decision management tools have a business glossary that defines:
    - Terms (or decision variables)
    - Domains or allowed values for these decision variables
    - Business entities or concepts that group terms or variables
    - Mappings to outside data sources
- DMN
    - Defines the need for defined Information Items
    - Expects these to be mapped to Decisions and Input Data
    - Does not define much of a standard format for them
- A standardized Business Glossary that could be exchanged by DMN implementations would be useful

Interoperability

- An ability to interchange the same decision model between different DMN compliant tools

- Common visual representation of diagrams and decision tables will allow a user **manually transform** a decision model from one implementation to another

- A standardized interchange format (XML/XSD) in DMN 1.1 would allow an **automatic transformation** of a decision model between different DMN tools

- What's needed
  - Fixed XML/XSD definitions
  - A complete example in the standardized interchange format
  - DMN Conformance Test Cases

Simplified Common Constructs

- ## Hit Policies
  - Focus on two core types: Single-Hit and Multi-Hit
  - Remove Priority-based tables
  - Consider if Any, First and other types are really useful

- ## Friendly-Enough Expression Language (FEEL)
  - Refine Simple-FEEL defined on concrete examples
  - Eliminate Range Exclusive confusion [a..b] v]a..b[v (a..b)
  - Express Aggregation methods as FEEL operators

> Discuss these and other DMN issues at the special meeting "DMN at BBC"
> on Nov. 5 at 5:50PM room: Milano V
> Anyone interested in DMN is welcome

Questions?

Thank You

DMN

James Taylor

james@decisionmanagementsolutions.com

www.DecisionsFirst.com

Dr Jacob Feldman

jacobfeldman@openrules.com

www.OpenRules.com

DECISION MANAGEMENT SOLUTIONS

OPEN RULES

Stop by at the OpenRules Booth T2 to see the complete demonstration