

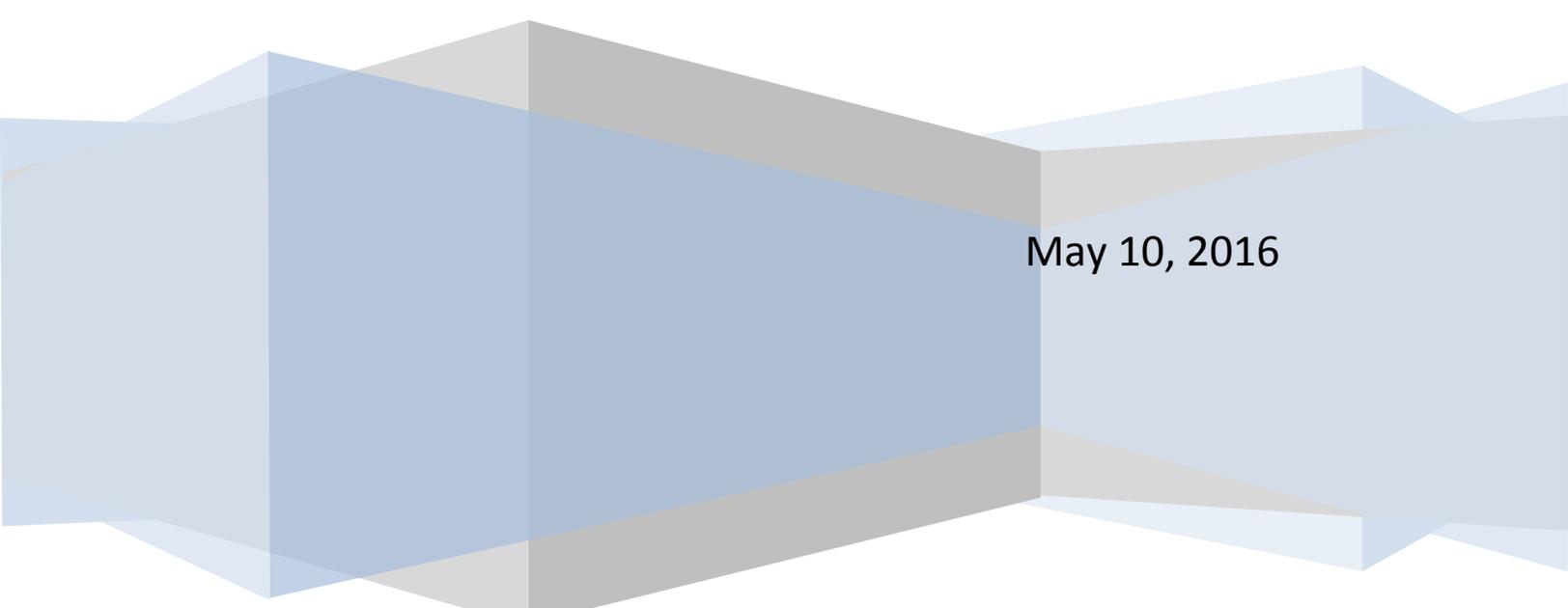
Oracle Corporation

Oracle Decision Service

Decision Model and Notation

Gary Hallmark

May 10, 2016

An abstract graphic at the bottom of the page consists of several overlapping, semi-transparent geometric shapes in shades of blue and grey, creating a layered, architectural effect.

Oracle's Decision Service is a cloud-based service for modeling, automating, and executing business decisions. It supports all standard OMG DMN FEEL expressions, including decision tables, boxed expressions, and all FEEL expressions allowed at DMN Conformance Level 3.

We will demonstrate the service by modeling, testing, and running a decision model abstracted from Oracle Application's Approvals Engine. This is a workflow engine that has both human-powered and automated decision making. We will focus on a small automated component - given a purchase order with multiple lines, each line having potentially multiple cost centers, return a list of approvers who must sign off on this PO.

Our DMN standard solution uses functions to encapsulate decision tables that must be executed for each line and for each cost center, and FEEL for-loops to invoke the functions for each line and for each cost center.

Decision models can be tested in two ways. The conventional method is to encapsulate all or part of a model in one or more decision services, and use a service testing framework. For more immediate feedback, the modeling tool allows you to attach sample data directly to the model inputs and see the resulting outputs as you build the logic.

Decision services created in a DMN Model can be executed as RESTful web services by posting values for the inputs and then receiving values for the outputs.

Slide 1



Today we will talk about a new decision service tool and engine we are developing for Oracle Process Cloud.

Slide 2

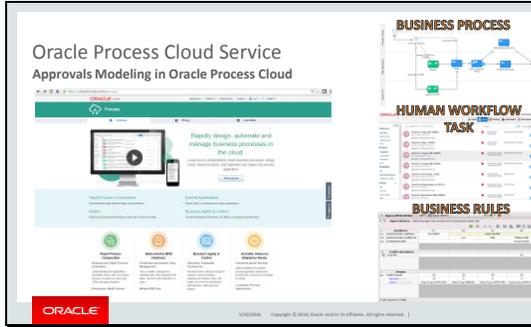


Oracle Process Cloud Service is a BPMN-based process design, execution, and management service running in the Oracle Public Cloud.

DMN is a new standard notation for modeling and automating parts of decision-heavy processes (e.g. expense approval, loan approval, spending approval, document approval).

Once you have mastered DMN's expression language, then you've mastered DMN. There's nothing hidden or stateful about DMN (unlike production rules). We will show how DMN easily handles some complex use cases that until now required a rules expert – or decision logic in your process diagram – including the processing of a master/detail/sub-detail hierarchy.

Slide 3

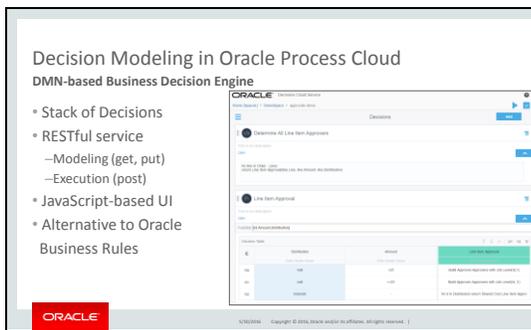


Oracle Process Cloud is our Platform-as-a-Service for business process automation. It offers many of the same features as the Oracle BPM Suite, including diagramming, KPI definition and dashboards, human workflow modeling and mobile-apps, and business rules based on production rules with layers of business-friendly syntax options.

Many use-cases are decision processes, e.g. approvals.

It is not always easy to see the decisions, and no standard place to document the decision/knowledge requirements.

Slide 4

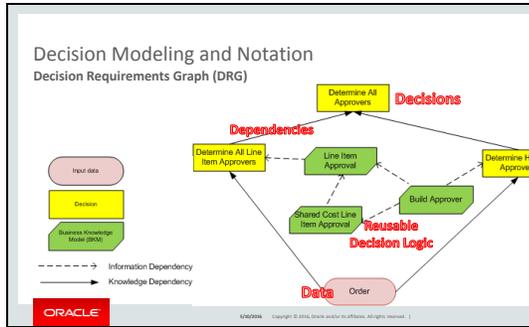


Our new Decision Model service is DMN-based decision modeling and execution engine.

We do not yet have a graphical editor, so the decisions are modeled in a stack, with the logic of one decision able to see the lower decisions in the stack, but not the higher decisions.

It will be offered in Oracle Process Cloud as an alternative to the existing Oracle Business Rules.

Slide 5



In DMN, a Decision Requirements Graph is for modeling decisions.

Here we have decisions, sub-decisions, dependencies, reusable decision logic called business knowledge models (BKMs), and input data.

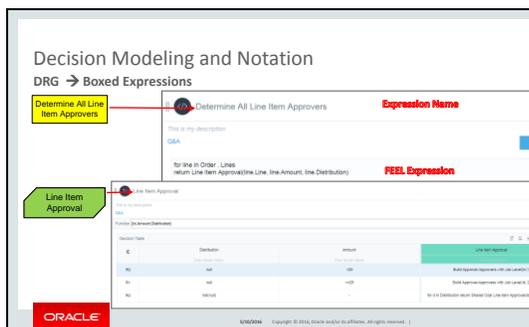
In our DMN-based product, we do not yet have the DRG. We can use an equivalent decision stack.

BKMs are represented as a decision whose logic is a function.

In words, the decision model determines all the approvers for an order by combining the approvers for the line items with the approvers indicated by the order header information.

An approver here is a little data structure that gets returned, and the data structure is built by Build Approver. To determine approvers for all lines, you must iterate over each line. And to determine approvers for a single line, you may need to iterate the Shared Cost detail.

Slide 6



Within each box in the decision diagram is a boxed expression. Boxed expression is the notation for modeling decision logic.

The logic in the boxed expressions is implemented with an expression language called FEEL.

Slide 7

Decision Modeling and Notation
Boxed Expressions → FEEL Expressions

Input Data = Context + Literal Expression

BKM = Decision Function Decision Table

The logic in the boxed expressions is implemented with an expression language called FEEL, which stands for Friendly Enough Expression Language. The boxed expressions can reference each other through their names.

Slide 8

Decision Logic	feature	example
	constants	true, 1.23, "red"
	ranges	[0..100], >0
	arithmetic	Net Amount * 1.055
	comparison	Transaction Amount > 50.00
	logical	not(A and B)
	navigation/filter	Order.Lines[id="1"].quantity
	if then else	if Transaction Amount > 50.00 then Large Discount else Small Discount
	for in return	for Line in Order.Lines return Line.Amount * Cust.Discount
	some/every in satisfies	some value in list of values satisfies value = "special"

FEEL expressions are interpreted with no side-effects. Every expression has a value. The value of a 'for' expression is a list of the return expression mapped to each element of the in expression. The value of an 'if then else' expression is the value of the 'then' or 'else' expression, depending on whether the 'if' expression is true or not.

Slide 9

FEEL - Decision Tables

Hit Policies
U – Unique
A – Any
C – Collect

Rules

Order Header Total Amount	Determine Header Approvers
>25	Build Approver:Approvers with Job Level(Order.Header.PD)
<=25	Build Approver:Approvers with Job Level(Order.Header.PD)

The decision table takes input expressions, matches them with the rules, and returns an output. The hit policy defines what happens when rules overlap.

Slide 10

FEEL – if then else expressions

```

|| Determine All Approvers
This is my description
Q&A
IF
Order-Header.Total Amount <2000
THEN
Build Approver.Auto-Approval()
ELSE
[Determine Header Approvers, Determine All Line Item Approvers]
  
```

If then else expressions are commonly used in rules expressions. We've added a new boxed expression for it, in an obvious way.

Inside the boxes, we can use literal expressions or another boxed expression.

Slide 11

FEEL – for in return expressions

```

for <name> in <list expression>
return expression
  
```

```

|| Determine All Line Item Approvers
This is my description
Q&A
for line in Order.Lines
return Line Item Approval(line.Line, line.Amount, line.Distribution)
  
```

The for in return expression loops through a list and evaluates an expression for each element, then returns the list result.

Slide 12

FEEL – for in return expressions

Apply boxed expression defined in another decision to each element of a list

Order

Line	Line Item Approval
1	Line Item Approval

Line Item Approval

```

for line in Order.Lines
return Line Item Approval(line.Line, line.Amount, line.Distribution)
  
```

Line Item Approval

Line	Line Item Approval
1	Line Item Approval

In the demo, we will show how to use the For expression to iterate through line items of an order.

Calculate the approvers for each line, and return a combined result.

Slide 13

FEEL - Item Definitions

- Define Type Structure of Data Items
- For
 - Input Data Elements
 - Function Definition Arguments
- Type Validation



The screenshot shows a 'Data Definition' window with a tree view of data items. The items listed are: Customer, Customer, CostBalance, Header InvDef, Total Amount, Line InvDef, Line, Description, Amount, Distribution, Order InvDef, Header, Line, Distribution InvDef, Cost Center, and ID.

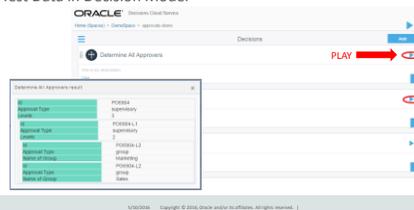
ORACLE

Item Definitions is a type definition in DMN used for defining the type of input data, function arguments, and decision results.

Slide 14

Testing Decision Models

- Supplying Test Data in Decision Model



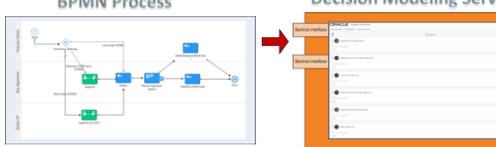
The screenshot shows the Oracle Decision Model testing interface. It displays a decision model named 'Determine All Approvals' with a 'PLAY' button. A data input window is open, showing a table of test data with columns for 'Decision ID', 'Approver', 'Request Type', 'Status', 'Request Date', 'Request Status', and 'Request ID'. The data includes entries for 'POKOR' and 'POKOR.L' with various 'Approver' and 'Request Type' values.

ORACLE

After a decision model is completed, users can define test data in the decision model for testing. In our tools, we have “play” buttons to test the decisions against the sample data.

Slide 15

Decision Services
Invoking the decision Model



The diagram illustrates a BPMN Process (Business Process Model and Notation) on the left, which is connected by a red arrow to a Decision Modeling Service on the right. The BPMN process shows a flow from a start event through several tasks and decision points. The Decision Modeling Service is shown as a web interface with a 'Decision' dropdown menu and a 'Evaluate' button.

ORACLE

A Decision Service encapsulates one or more decisions in a decision model for external consumption. For instance, a BPMN Process can call a decision service to evaluate a decision.

Slide 16

Demo Overview - Purchase Order Approval Demo
Header and Line Item Approvals

Input Data: Purchase Order

- Header
- Line Items

Multiple Line Items

The screenshot shows a 'Purchase Order' form with fields for 'Header' and 'Line Items'. An arrow labeled 'Multiple Line Items' points from the 'Line Items' section to a detailed view of a single line item on the right. The Oracle logo is in the bottom left corner.

For our demo, we are going to show a decision model that takes input data from a Purchase Order and derive a list of approvers needed for approving the purchase order.

Slide 17

Demo Overview - Purchase Order Approval Demo
Header and Line Item Approvals

Output Result - Approver Group Attributes

- Item ID
- Approval Type (Auto, By Group, By Supervisory Level)
- Approval Attribute

Determine All Approvers result

id	PO6904
Approval Type	supervisory
Levels	3
id	PO6904-L1
Approval Type	supervisory
Levels	2
id	PO6904-L2
Approval Type	group
Name of Group	Marketing
id	PO6904-L2
Approval Type	group
Name of Group	Sales

The screenshot displays a table titled 'Determine All Approvers result' with columns for 'id', 'Approval Type', and 'Levels'. It lists three approval levels: a supervisory level 3 for the entire PO, a supervisory level 2 for line item PO6904-L1, and a group-level approval for line item PO6904-L2, with sub-groups for Marketing and Sales. The Oracle logo is in the bottom left corner.

The demo shows how DMN can handle multiple line items. The output is a list of approver attributes. The attributes tell how the item should be approved, and who should be approving it.

Slide 18

Demo Overview - Purchase Order Approval Demo
Header and Line Item Approvals

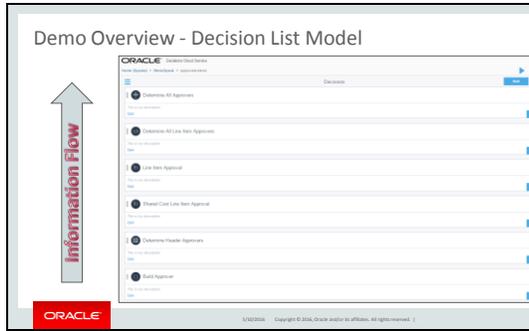
- Header Approval Logic
 - An Approver from different Management Level based on PO's Total Amount
- Line Approval Logic
 - An Approver from different Management Level based on Line Item's Amount
 - For Shared-Cost Line Items, Approvers from Multiple Cost Centers Required

The screenshot shows the 'Approval Logic' section of the demo. It details the logic for header and line item approvals, including management levels and shared-cost requirements. The Oracle logo is in the bottom left corner.

For header approval, we will need a manager from a particular supervisory level based on the PO total amount. For line items, it may require one or more approvers from different cost centers.

Slide 19

Demo Overview - Decision List Model



The screenshot shows the Oracle Decision List Model interface. On the left, a vertical grey arrow points upwards with the text 'Information Flow' written vertically inside it. The main area displays a list of decision items, each with a blue checkmark and a '1' in a blue box. The items are: 'Decision: All Approvals', 'Decision: All Low Risk Approvals', 'Low Risk Approval', 'Shared Card Low Risk Approval', 'Decision: Public Approvals', and 'Email Approval'. The Oracle logo is visible in the bottom left corner of the screenshot.

Slide 20



We will take a look at the live demo now.