



ENTERPRISE ARCHITECT

用户指南系列

Decision Model and Notation (DMN)

Author: Sparx Systems

Date: 20/06/2023

Version: 16.1

创建于  **ENTERPRISE
ARCHITECT**

目录

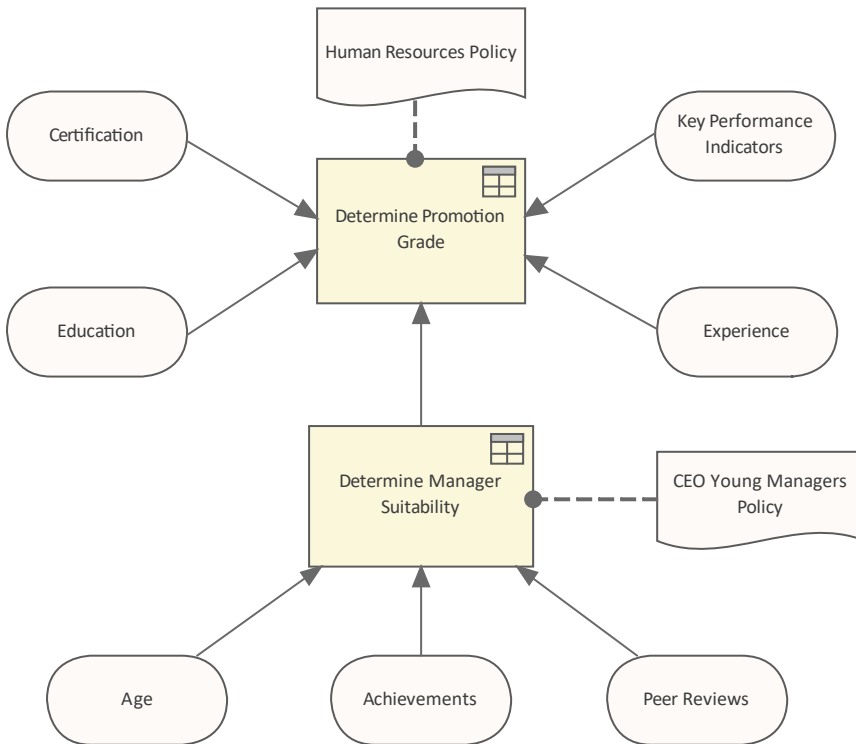
Decision Model and Notation (DMN)	4
使用 DMN建模	7
决策	8
业务知识模型	9
BKM 参数	11
用于仿真的输入参数值	13
决策表仿真示例	15
文字表达仿真示例	17
输入数据	18
InputData DMN 表达式	19
项目定义	21
项目定义工具栏	23
项目定义和数据集	24
部件	27
允许的值枚举	29
数据集	30
使用 DataObjects 交换数据集	33
服务决策服务	37
仿真决策服务	40
开始	42
更多信息	44
创建决策模型	45
集成BPSim进行仿真	51
示例：将 DMN 决策服务集成到决策数据物件和属性参数中	56
示例：将 DMN业务知识模型集成到属性参数中	57
导入 DMN XML	58
决策需求图表	60
决策表达式编辑器	61
关系	63
盒装清单	66
编辑DMN 表达式对话框	69
文字表达	71
文字表达式编辑器的工具栏	74
示例-偿还贷款	75
调用	77
调用编辑器工具栏	80
示例1 - 将输入数据绑定到业务知识模型	81
示例2 - 将上下文条目变量绑定到业务知识模型	83
DMN 表达式自动完成	84
DMN 表达式验证	88
盒装上下文	90
盒装上下文编辑器的工具栏	92
示例-贷款分期计算	93
决策表	97
决策表编辑器工具栏	103
目标决策表命中策略	105
决策表验证	107

集成到UML类元素	110
示例图表	116
代码生成和测试模块	118

Decision Model and Notation (DMN)

创建和仿真企业决策的详细模型

组织面临着日益艰难的运营环境，来自现有和新市场参与者的激烈且往往不可预测的竞争、政府和行业法规的变化以及客户群社会结构的剧变。组织在此上下文的决定对于组织的成功及其在这些未知的公司水域中引导安全路径的能力至关重要。使用企业架构师模型和表示法 (决策) 特征，您不仅可以对组织做出的决策进行建模，还可以从这些模型中进行运行，以根据示例数据集预测结果。这种语言的力量在于，业务人员可以很容易地理解和使用简单但富有表现力的决策需求图，这些图详细说明了细节是什么、决策的输入是什么以及预期的输出是什么。可以通过多种方式记录规则，包括易于定义的决策库表。一旦完成这些图表以及随附的输入数据示例，就可以模拟以显示决策的结果。



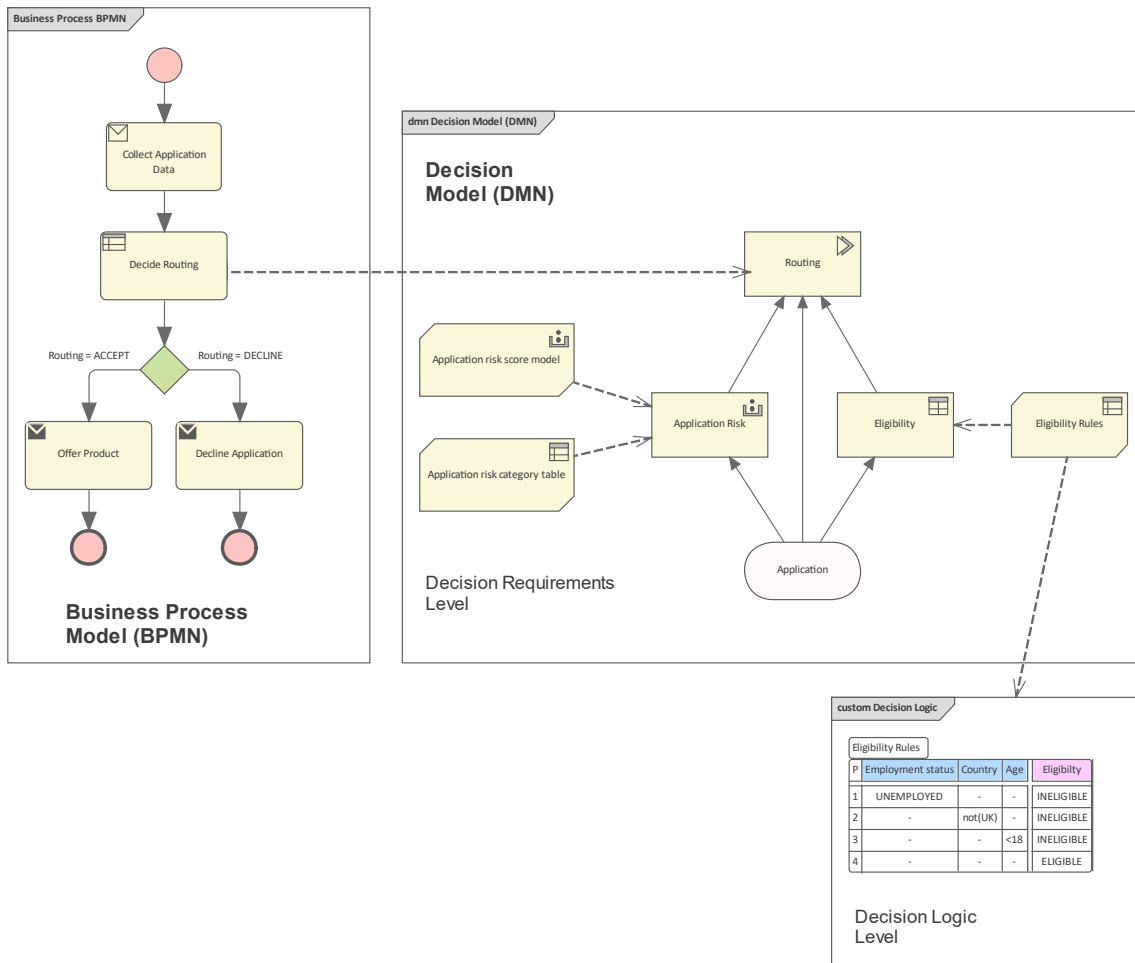
决策需求图显示了具有决策业务知识模型的决策和包括另一个决策在内的许多输入。

一旦企业定义、模拟和测试了这些模型，技术人员和工程师就可以改进这些模型，并直接从模型中自动生成包括编程代码在内的软件工件，从而减少解释错误的可能性并缩短实施时间。

什么是 DMN ?

DMN 旨在提供业务流程模型和决策逻辑模型之间的桥梁：

- 业务流程模型将定义业务流程中需要进行决策的任务
- 决策需求图表将定义在这些任务中要做出的决策、它们的相互关系以及它们对决策逻辑的要求
- 决策逻辑将在足够的细节中定义所需的决策，以允许验证和/或自动化



综合起来，决策需求图和决策逻辑允许您通过在细节中指定在流程任务中执行的决策制定来补充业务决策模型的完成决策模型。

DMN 提供了跨越决策需求和决策逻辑建模的结构。

- 对于决策需求建模，它定义了决策需求图表（决策）的概念，包括一组元素及其连接规则，以及相应的符号：决策需求图表（DRD）。
- 对于决策逻辑建模，它提供了一种称为 FEEL 的语言，用于定义和组合决策表、计算、if/then/ else 逻辑、简单的数据结构以及来自 Java 和 PMML 的外部定义逻辑，并形成具有正式定义语义的可执行表达式。

在 Enterprise Architect 中使用 DMN 的好处

建模使用 DMN 的决策过程允许您将复杂的决策过程记录、指定和分析为相互关联的决策、业务规则、数据集和知识源的系统。通过这样做，您可以将高度复杂的决策过程分解为支持决策和输入数据的网络。这有助于更轻松地理解整个流程，支持流程重构并简化验证流程的任务，让您可以轻松验证构成整个流程的各个步骤。

当您使用 DMN 在 Enterprise Architect 中构建决策模型时，您可以运行该模型的模拟来验证模型的正确性。验证您的模型后，您可以生成 Java、JavaScript、C++ 或 C# 的 DMN 模块。生成的 DMN 模块可以与 Enterprise Architect BPSim 执行引擎、执行可执行状态机一起使用，或者在您正在实施的单独软件系统中使用。

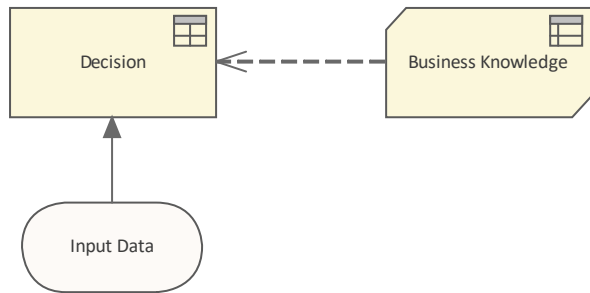
Enterprise Architect 还提供了“测试模块”功能，这是一个将 DMN 与 BPMN 集成的预处理。目的是生成 BPMN2.0::DataObject 元素，然后使用这些元素验证指定的目标决策是否已通过 DMN 模块正确评估。然后，您通过加载数据对象并将 DMN 模块决策分配给 BPSim 属性来配置 BPSim。

此特征在 Enterprise Architect 的统一版和终极版中可用，从 15.0 版开始。

决策需求图

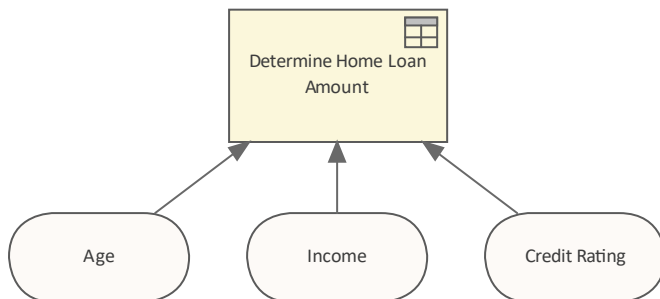
DMN 决策需求模型由一个或多个决策需求图表(决策)中描述的决策需求图 (DRG) 组成。建模的元素是决策、业务知识领域、业务知识来源、输入数据和决策服务。

DRG 是由由需求连接的元素组成A图，并且是自包含的，因为 DRG 中任何决策的所有建模需求（其信息、知识和权限的直接来源）都存在于同一个 DRG 中。将 DRG 的完成定义与呈现它的任何特定视图的 DRD 区分开来是很重要的，这可能是部分显示或过滤显示。



使用 DMN 建模

本主题向您介绍创建决策模型所需的最重要元素。正如前面所讨论的，决策模型有两个基本部分，即决策需求图和决策逻辑。创建决策图是直截了当的，可能练习中最繁重的部分将解开组织做出决策的方式以及这些决策的输入是什么。图表通常包含链接在一起的决策，描述一个决策可以为另一个决策提供输入等事实。



决策的需求图显示了使用决策表的决策输入。

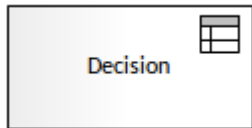
决策的逻辑使用许多设备来描述，但最常用和最容易理解的形式是决策决策表。决策表像电子表格一样包含行和列，并涵盖所有可能的输入组合以产生许多输出。例如，如果申请人年龄大于 21 岁且小于 65 岁，并且年收入 60,000 美元且信用评级良好，那么银行将借给他们 300,000 美元用于家贷款。

Determine Home Loan Amount				
	Age	Income	Credit Rating	Loan Amount
U				
1	-	-	-	-
2	-	-	-	-
3	-	-	-	-

显示三个输入和一个输出A决策表，将添加行来定义规则。

决策

决策元素用于基于一个或多个输入来评估输出。确定输出的逻辑要么在该决策元素中定义，要么调用与决策决策相关的决策业务知识模型中包含的决策逻辑。



输入

决策可以有一个任意数量的输入，包括在元素中定义输入值的选项。最常见的输入是使用 **Input Data** 元素。

输出

一个决策可以有零个或一个输出。输出可以是复杂的数据集。

值表达式

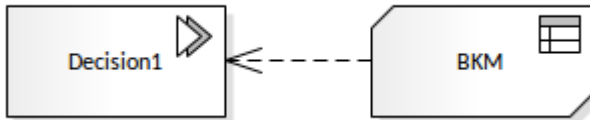
决策元素的输出是决策表达式确定的。值表达式包含元素的决策逻辑，可以采用以下四种形式之一：决策决策表、文字表达式、调用或盒装上下文。值表达式是使用 DMN 表达式编辑器定义和编辑的，它根据所使用的表达式类型显示四种格式之一。

当显示在图表上时，决策元素会在右上角显示一个图标，指示它正在使用哪种类型的价值表达。

类型	描述
	决策表是一个组相关输入和输出表达式的表格表示，组织成规则，指示哪个输出条目适用于一组特定的输入条目。
	文字表达式是 DMN 表达式的最简单形式。它通常定义为单行语句或 if-else 条件块。
	决策调用需要一个知识需求连接器来引用业务知识模型元素。决策元素只包含为评估业务知识模型（上下文）提供时间的参数，或者所有从部件返回的结果都可以设置为作为决策的输出决策。
	盒装上下文是上下文条目集合。每个上下文条目由一个变量和一个表达式组成。上下文也有一个结果值。

业务知识模型

A 业务知识模型(BKM)元素代表了一个可重用的决策逻辑。通常，它连接到调用决策并传递一组输入的决策元素。BKM 使用其内部逻辑评估传递回决策的输出。




除非 BKM 处理固定值，否则它通常需要定义一组输入参数以及输出的定义。使用DMN 表达式窗口定义参数和决策逻辑。

(Input 1, Input 2)			
U	Input 1	Input 2	Output 1
1	-	-	-
2	-	-	-
3	-	-	-

输入和输出

在决策模型中使用时，BKM 必须通过 KnowledgeRequirement 连接到决策或另一个决策，通过它接收其输入。

输入参数使用  图标定义。这些可以设置为使用 ItemDefinition 定义的简单类型或复杂类型。输入参数的命名会影响值表达式中的命名。

输出

BKM 输出是A KnowledgeRequirement 实现的，它必须是决策或另一个 BKM 的输入。输出定义使用：

- 文字表达式的  图标
- 用于决策表、盒装内容和调用的DMN 表达式表中的输出列。

输出可以是使用 ItemDefinition 定义的简单类型或复杂类型。



值表达式

为了定义评估输出的方法，基于决策逻辑，业务知识模型 (BKM) 元素包含一个值表达式。这是使用DMN 表达式窗口定义和编辑的，它有四种格式，格式由您要使用的值表达式的类型确定。

BKM元素可以使用这些结构为值表达式设置。每个都在模型中显示，带有一个图标。

类型	描述
	决策表是A组相关输入和输出表达式的表格表示，组织成规则，指示哪个输出条目适用于一组特定的输入条目。
	文字表达式是 DMN 表达式A最简单形式。它通常定义为单行语句或 if-else 条件块。
	决策调用要求A知识需求连接器来引用业务模型元素。它仅包含为评估上下文模型提供时间的参数。
	盒装上下文是上下文条目A集合。每个上下文条目由一个变量和一个表达式组成。上下文也有一个结果值。

验证和测试

为确保 BKM元素能够产生正确的输出，可以使用验证图标。BKM 也可以作为A单元进行测试，以确保它可以使用仿真按钮进行操作。有关更多详细信息，请参阅用于仿真帮助的仿真参数值主题。

BKM 参数

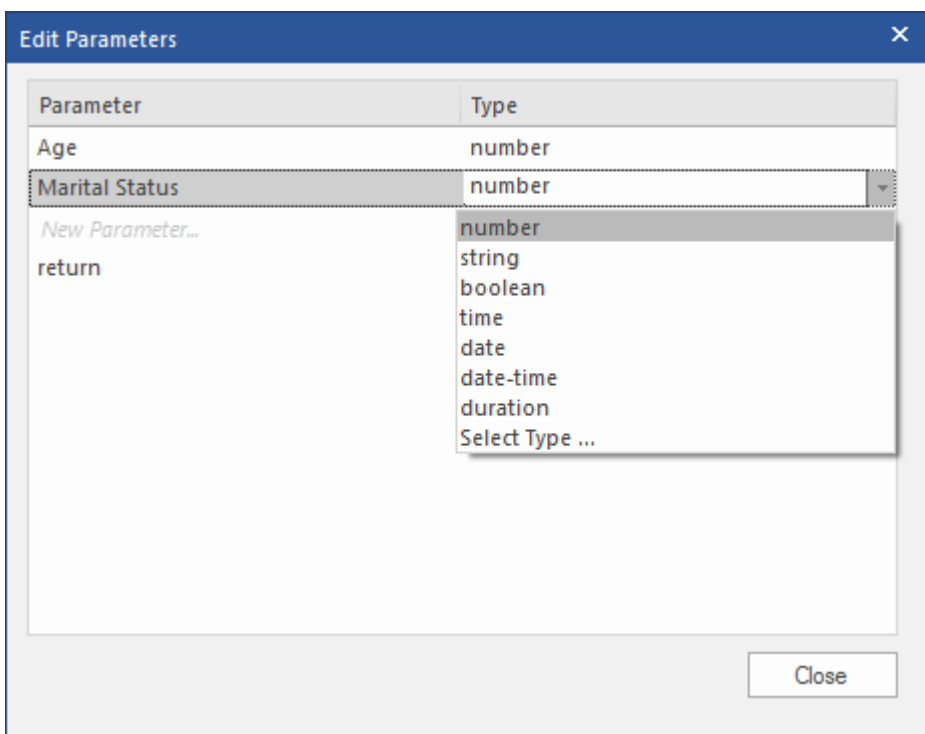
业务知识模型 (BKM) 以函数定义A形式实现，以参数和DMN表达式为主体（如决策表、盒装上下文或文字表达式）。

由于函数旨在独立运行，并由决策或其他 BKM 调用，因此有必要定义任何输入参数。此外，对于文字表达式，您必须定义输出参数。

定义任何输入参数时，您可以使用默认值设置它们以进行测试。创建 BKM 后，要验证它是否正常运行，您可以基于这些默认值运行模拟。

业务知识模型参数

要打开“编辑参数”对话框，请在DMN 表达式窗口中单击编辑参数按钮 ：



注记：这是包含返回类型的文字表达式的示例。

编辑参数

您可以对参数执行以下操作：

行动	描述
	通过在“新参数...”行中键入来添加新参数。
	通过在单元中就地编辑来修改现有参数的名称。
	使用上下文菜单删除现有参数。
	单击类型以启用下拉菜单。从下拉列表中选择参数的类型。

设置项目定义类型

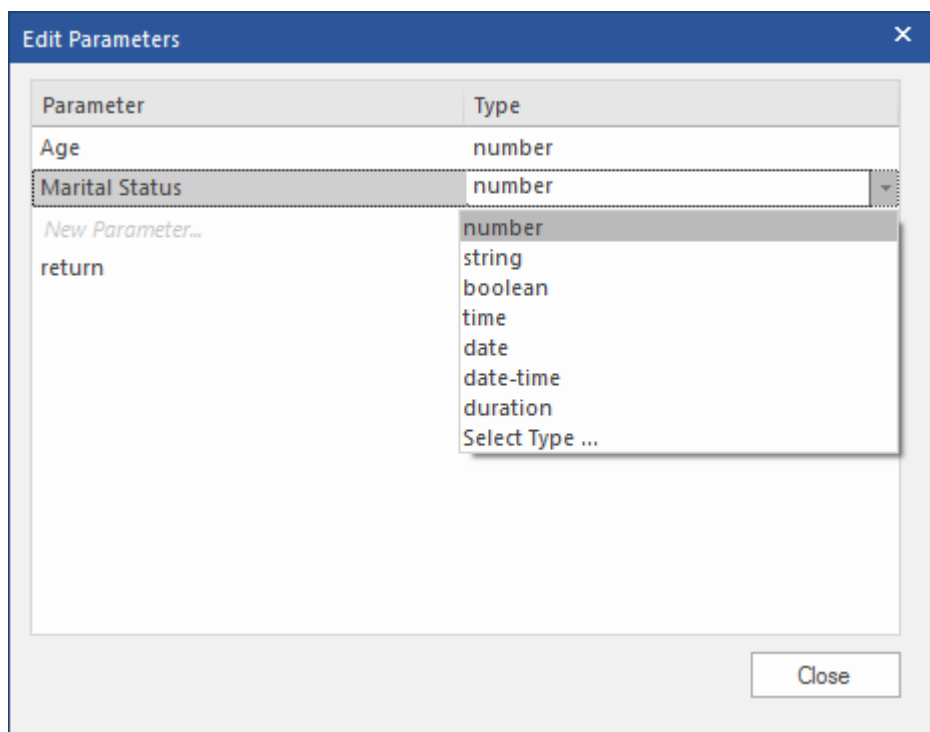
更改参数的类型时，可以选择从参数中选择预定义类型。选项是“选择类型...”。选择此选项，它将打开一个对话框以选择 `ItemDefinition`。

用于仿真的输入参数值

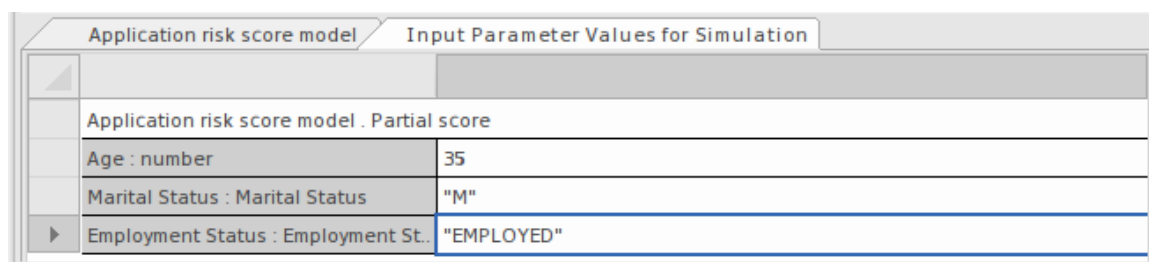
由于业务知识模型是独立的，因此可以通过提供一组默认值作为其参数的输入来执行模拟“单元测试”。这些值可以在DMN 表达式窗口的仿真输入参数值选项卡中定义。

业务知识模型参数(BKM)

使用工具栏上的编辑参数按钮  从DMN 表达式窗口访问 BKM 的参数：



这些参数默认值集（可用于 BKM 模拟）在“用于仿真的输入参数值”选项卡中定义，在DMN 表达式窗口上：



设置好这些参数后，BKM就可以使用仿真按钮进行测试 。

仿真例子

这是使用输入参数值进行仿真的两个示例。

类型	描述
决策表	基于输入参数仿真选项卡中设置的值的 BKM决策表元素的示例模拟。

文字表达	基于“仿真仿真参数值”选项卡中设置的值的 BKM 文字表达式元素的示例模拟。
------	--

决策表仿真示例

本节介绍的业务知识模型 (BKM) 示例可通过模型向导 (Ctrl+Shift+M) 获取。在您的模型中选择一个包，调用模型向导实验并 - 从蓝图下拉菜单中 - 选择 需求|决策建模”。

要访问本节中使用的示例：

- 为 DMN决策|创建模式” A完全的示例
- 在浏览器窗口中导航到A完全示例|业务模型”

Enterprise Architect示例模型 (示例) 中也提供了它：

- 在浏览器窗口中导航到 分析和浏览业务建模A > DMN 示例 >完全示例>业务模型”


双击 “Eligibility rules”元素，在DMN 表达式窗口中打开BKM

当为业务知识模型创建决策决策表时，我们可以通过绑定一些值来测试这个BKM：

Eligibility rules		Input Parameter Values for Simulation		
(Pre-Bureau Affordability, Pre-Bureau Risk Category, Age)				
P	Pre-Bureau Risk Category	Pre-Bureau Affordability	Age	Eligibility
	VERY LOW, LOW, MEDIUM			INELIGIBLE, ELIGIBLE
1	DECLINE	-	-	INELIGIBLE
2	-	false	-	INELIGIBLE
3	-	-	<18	INELIGIBLE
4	-	-	-	ELIGIBLE

我们可以提供如下测试值：

Eligibility rules		Input Parameter Values for Simulation	
Eligibility rules . Eligibility			
Pre-Bureau Affordability	true		
Pre-Bureau Risk Category	"VERY LOW"		
Age	16		

点击工具栏上的仿真按钮得到这个  结果：

Eligibility rules		Input Parameter Values for Simulation		
(Pre-Bureau Affordability = true, Pre-Bureau Risk Category = "VERY LOW", Age = 16)				
P	Pre-Bureau Risk ...	Pre-Bureau Affor...	Age	Eligibility
	VERY LOW	true	16	INELIGIBLE
1	DECLINE	-	-	INELIGIBLE
2	-	false	-	INELIGIBLE
3	-	-	<18	INELIGIBLE
4	-	-	-	ELIGIBLE

- 运行时参数值将代替模拟模式下的 “允许值”
- 突出显示有效规则

- 由于本决策表的命中策略是P（优先级），因此最终结果由输出值的顺序决定；由于 'INELIGIBLE' 和 'ELIGIBLE' 是输出值，并且 'INELIGIBLE' 出现在 'ELIGIBLE' 之前，规则 #3 将给出最终结果并且该申请人是 'INELIGIBLE'。

文字表达仿真示例

本节介绍的业务知识模型可从模型向导 (Ctrl+Shift+M) 获取。在您的模型中选择一个包，调用模型向导实验并 - 从蓝图下拉菜单中 - 选择 需求|决策建模”。

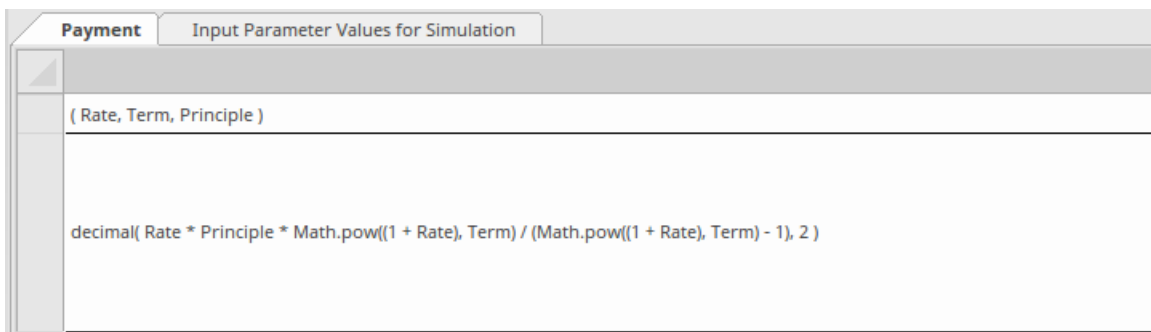
要访问本节中使用的示例：

- 创建 DMN业务知识模型>业务知识模型文字表达”模式
- 在浏览器窗口中导航至 业务知识模型文字表达 > 支付”

Enterprise Architect示例模型 (示例) 中也提供了它：

- 在浏览器窗口中导航至 模型仿真>DMN模型>业务知识模型>业务知识模型文字表达”

双击 支付”元素，打开DMN 表达式窗口中的BKM。



与决策决策表类似，作为盒装表达式实现的业务知识模型也可以进行测试。

以 支付”元素为例。此 BKM 将根据利率、期限数和本金金额计算每月还款额。

我们可以提供如下测试值：

Payment		Input Parameter Values for Simulation
Payment		
Rate		0.04 / 12
Term		30 * 12
Principle		300000

点击工具栏上的仿真  按钮；得到这个结果：

Payment		Input Parameter Values for Simulation
		(Rate = 0.0033333333333333335, Term = 360, Principle = 300000)return = "1432.25"
		decimal(Rate * Principle * Math.pow((1 + Rate), Term) / (Math.pow((1 + Rate), Term) - 1), 2) (evaluation result = "1432.25")

运行时参数和返回值将显示为等号 “=”，后跟运行时值。该值也显示为与其父图上的元素相对的标签。

在此示例中，假设 30 年的年利率为 4%，本金为 300,000 美元，则每月还款额为 1,432.25 美元。

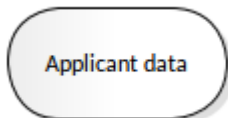
注记： DMN 库已经定义了 PMT 函数；此示例主要演示文字表达式的工作原理以及如何使用一组参数对其进行测试。

输入数据

InputData元素用于将源自模型之外的一组值输入到决策中。该组值用于评估决策。它从 **ItemDefinition** 派生其类型和一组值。

概述


InputData 元素是通过将  **Input Data** 图标从工具箱拖到 DMN 图上来创建的。



InputData元素的名称必须唯一，不能与其他决策模型、决策、业务知识模型、决策服务或导入决策模型中的名称重复。

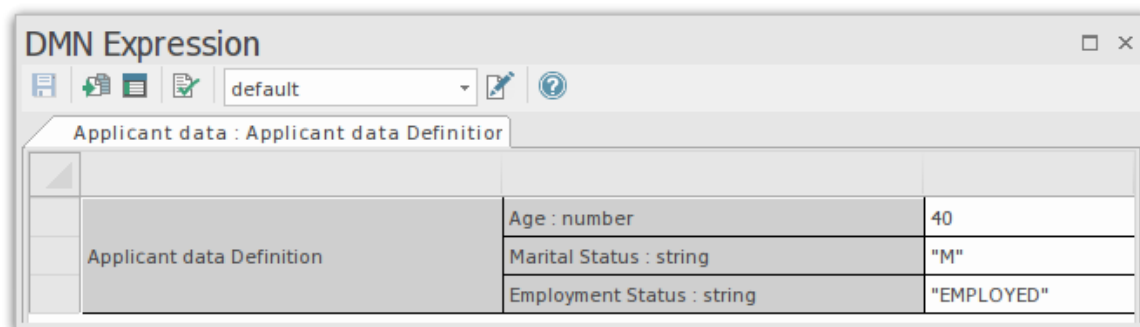
引用 ItemDefinition

数据的结构以及 **InputData**元素的值集在 **ItemDefinition**元素中定义。A DMN **InputData**元素必须由 **ItemDefinition** 引用（键入）：

- 单击 **InputData**元素或DMN 表达式窗口上的  图标
- 选择 **InputData**元素并按 **Ctrl+L** 从对话框中选择 **ItemDefinition**

InputData属性

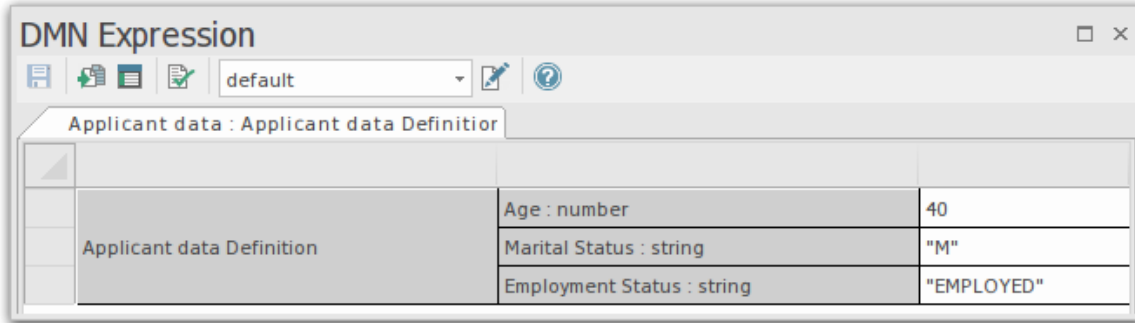
InputData元素的属性可通过DMN 表达式窗口访问。双击 **InputData**元素以打开此窗口。



DMN 表达式窗口提供了数据结构的视图以及对可在模拟中使用的数据集的访问。

InputData DMN 表达式






DMN 表达式窗口提供了 InputData 数据结构的视图、更改项值的选项以及对可在模拟中使用的数据集的访问。



访问

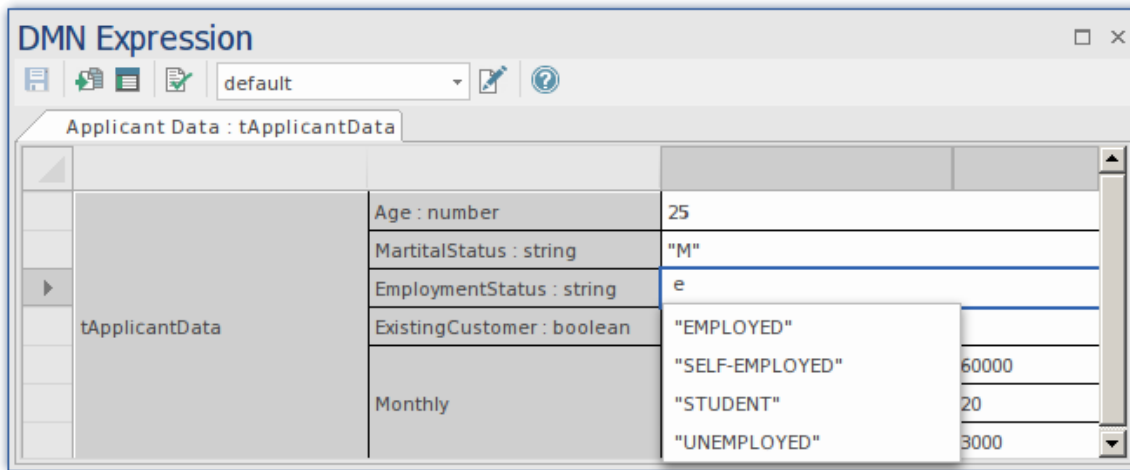
功能区	仿真>决策分析>DMN> DMN 表达式，然后选择/创建一个InputData
其它	双击一个 DMN InputData元素

工具栏选项


选项	描述
	将配置保存到当前 InputData元素。
	通过选择对 ItemDefinition 的引用来设置 InputData 的类型。
	打开此 InputData 作为其类型定义引用的 ItemDefinition元素。
	运行 InputData 的验证。Enterprise Architect将执行一系列验证以帮助您识别 InputData 中的错误。
	用于选择引用此 InputData 的 ItemDefinition 中定义的数据集的选项。
	打开用于编辑此输入数据的数据集的对话框。每个 InputData 可以定义多个数据集。有了这个特征，DMN仿真可以通过选择不同的数据集来快速测试决策的结果。

自动完成

如果 InputData 有一个定义了“允许值”的字段，则可以通过选择该字段，按空格键，然后从下拉列表中选择一个选项来填充该字段。



数据集

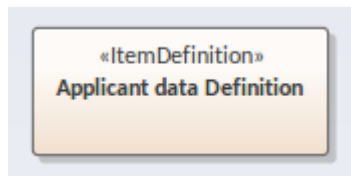
数据集在 InputData 元素引用的 ItemDefinition 中定义。使用工具栏下拉菜单，您可以从 ItemDefinition 中选择一个数据集。选择一组后，您可以更改项目的值。您还可以通过使用  图标打开编辑数据集窗口来添加新数据集。

项目定义

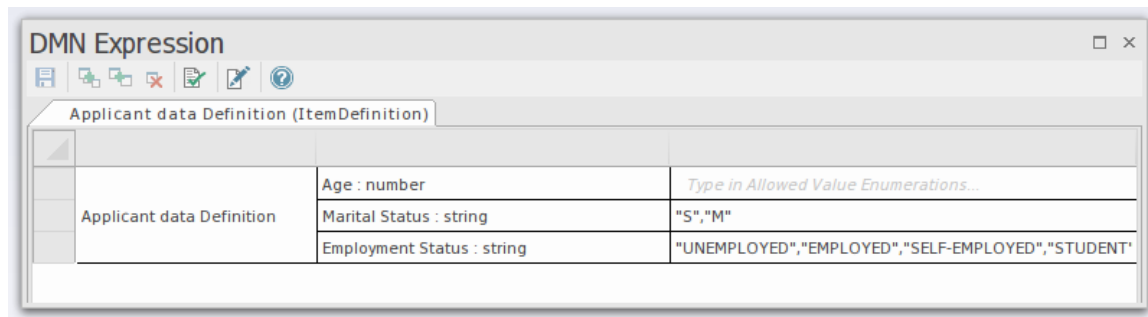
创建决策模型的基础是定义模型中使用的数据项的结构。ItemDefinition 用于定义输入数据的结构，并可选择限制数据允许值的范围。ItemDefinitions 的范围可以从简单的单一类型到复杂的结构化类型。

概述

ItemDefinition 元素是通过将  图标从 DMN 工具箱页面拖到 DMN 图上来创建的。



通过 DMN 表达式窗口访问属性元素的核心属性。



访问

打开 ItemDefinition 元素的 DMN 表达式窗口：

功能区	仿真>决策分析>DMN> DMN 表达式，然后选择或创建一个决策
其它	双击 DMN ItemDefinition

DMN 表达式和数据集

此图是 DMN 表达式窗口的概述，显示了复杂的数据项以及数据定义中使用的关键字段的布局。包括使用此 ItemDefinition 定义的数据集的视图。数据集是符合 ItemDefinition A 数据“实例”，其中包含要在 DMN 模拟中使用的一组值。

The image shows two screenshots from a DMN tool. The top screenshot is the 'DMN Expression' window for 'tApplicantData (ItemDefinition)'. It displays a table of properties for the item definition:

Age	number	Type in Allowed Value Enumerations...
MaritalStatus	string	"S","M"
EmploymentStatus	string	"EMPLOYED","SELF-EMPLOYED","STUDENT","U"
ExistingCustomer	boolean	Type in Allowed Value Enumerations...
Monthly	Income	number
	Repayments	number
	Expenses	number

Annotations with arrows point to:

- 'number' in the Age row: Type for Leaf Item
- '"S","M"' in the MaritalStatus row: Allowed Values
- '"EMPLOYED","SELF-EMPLOYED","STUDENT","U"' in the EmploymentStatus row: Allowed Values
- The 'Monthly' row: Composite Item

The bottom screenshot is the 'Edit Data Set For DMN ItemDefinition' window for 'tApplicantData'. It shows a table of data sets:

Data Set / Item	Type	Value
default		
tApplicantData . Age	number	25
tApplicantData . EmploymentSta...	string	"EMPLOYED"
tApplicantData . Existing Customer	boolean	true
tApplicantData . MaritalStatus	string	"M"
tApplicantData . Monthly . Expe...	number	3000
tApplicantData . Monthly . Income	number	60000
tApplicantData . Monthly . Repa...	number	20
60 Single Employed		
tApplicantData . Age	number	60

An annotation with an arrow points to the table: Data Set

由于项目定义是模型中的基础元素，因此建议在继续在模型中使用它们之前对其进行验证。这将确保在创建复杂模型的过程中及早解决任何问题。

有关设置帮助的更多详细信息，请参阅这些帮助主题：

- DMN 项目定义、数据集和输入数据
- 部件
- *ItemDefinition* 允许值
- DMN 表达式自动完成
- DMN 表达式验证

项目定义工具栏

此表提供了在选择 ItemDefinition 元素时在 DMN 表达式窗口中可访问的的特征的描述。

工具栏选项

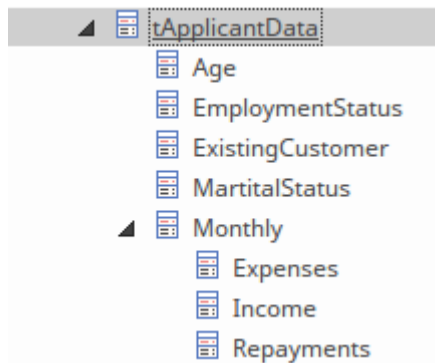
选项	描述
	保存当前 ItemDefinition 的配置。
	创建一个新的数据组件作为选定组件的子组件。
	创建一个新的数据组件作为选定组件的兄弟。
	删除选定的数据组件。
	验证 ItemDefinition；Enterprise Architect 将执行一系列验证以帮助您识别 ItemDefinition 中的任何错误。
	打开“编辑数据集”对话框，您可以在其中创建和编辑 ItemDefinition 的实例以供 InputData 元素使用。

项目定义和数据集

决策描述了决策模型中使用的数据项的类型和结构。它作为决策元素、决策元素和业务知识模型参数的数据类型定义。ItemDefinition 还可以定义数据集，这些数据集提供用于 DMN 模拟的值集。在不同数据集之间切换提供了使用决策模型进行“假设”分析的能力。

项目定义结构

复杂A ItemDefinition 由嵌套元素组成。例如，*tApplicantData*的结构如下：



tApplicantData ItemDefinition 示例是五个子项的复合类型。“每月”由三个子项（费用、收入和还款）组成。叶组件（非复合）将具有原始类型，例如数字、string或布尔值。

数据集

可以使用工具栏上的  图标查看和编辑 ItemDefinition 的数据集。通过“编辑数据集”对话框，您可以添加、删除和复制数据集。还支持数据集的 CSV 导入和导出。

ItemDefinition: tApplicantData

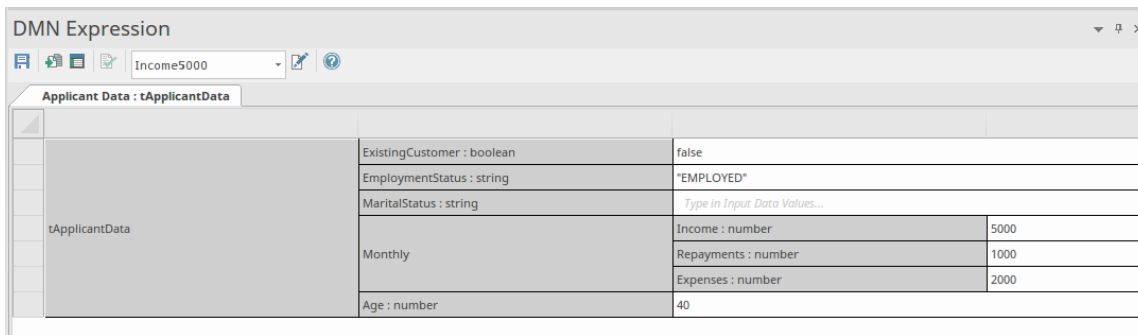
Data Set / Item	Type	Value
▲ default		
tApplicantData . Age	number	40
tApplicantData . EmploymentStatus	string	"EMPLOYED"
tApplicantData . ExistingCustomer	boolean	false
tApplicantData . MaritalStatus	string	"M"
tApplicantData . Monthly . Expenses	number	2000
tApplicantData . Monthly . Income	number	4000
tApplicantData . Monthly . Repayments	number	1000
▲ Income4000		
tApplicantData . Age	number	40
tApplicantData . EmploymentStatus	string	"EMPLOYED"
tApplicantData . ExistingCustomer	boolean	false
tApplicantData . MaritalStatus	string	"M"
tApplicantData . Monthly . Expenses	number	2000
tApplicantData . Monthly . Income	number	4000
tApplicantData . Monthly . Repayments	number	1000
▲ Income5000		
tApplicantData . Age	number	40
tApplicantData . EmploymentStatus	string	"EMPLOYED"
tApplicantData . ExistingCustomer	boolean	false
tApplicantData . MaritalStatus	string	"M"
tApplicantData . Monthly . Expenses	number	2000
tApplicantData . Monthly . Income	number	5000
tApplicantData . Monthly . Repayments	number	1000

OK

如示例所示，tApplicantData 的 *ItemDefinition* 定义了三个数据集：

- 默认
- 收入4000
- 收入5000

每个数据集都可以在输入到 *ItemDefinition* 的 *InputData* 元素中查看。例如，'Applicant Data' *InputData* 元素被键入到 'tApplicantData' *ItemDefinition*。此处所示的“申请者数据”的 DMN 表达式窗口根据窗口工具栏下拉列表中选择的数据集（本例中为 *Income5000*）显示数据值。




The screenshot shows the 'DMN Expression' window with a toolbar and a dropdown menu set to 'Income5000'. Below the toolbar, there is a tab labeled 'Applicant Data : tApplicantData'. The main area displays a table with the following data:

Property	Type	Value
ExistingCustomer	boolean	false
EmploymentStatus	string	"EMPLOYED"
MaritalStatus	string	Type in Input Data Values...
Income	number	5000
Repayments	number	1000
Expenses	number	2000
Age	number	40

将引用设置为参考

DMN InputData元素设置为由 ItemDefinition 引用（键入），使用以下任一A：

- InputData元素或DMN 表达式窗口上的  图标
- 选择 InputData元素并按 Ctrl+L 从对话框中选择 ItemDefinition

还有其他使用 ItemDefinitions 的情况；例如，在为 BKM 中的输入参数或决策表中的输出参数设置类型时。

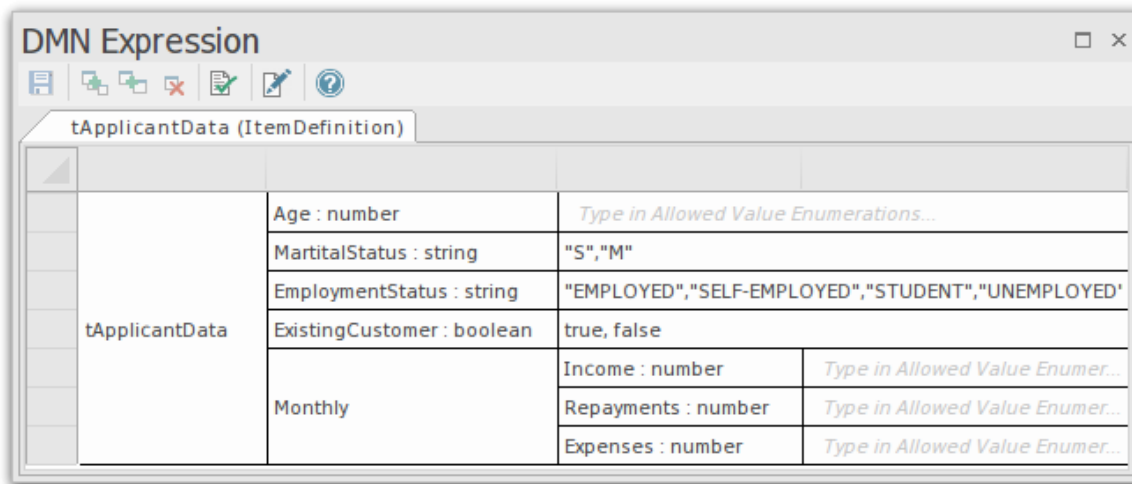
部件

一个 `ItemDefinition` 元素可以定义为一个组件树，它只包含以下之一：

- A 类型或
- A 组组合元素

在这个组件树中，如果组件是没有子组件的“叶子”，则必须将其设置为内置类型。如果 `ItemDefinition` 具有子组件，则将这些子/叶组件设置为内置类型。

例如，`Applicant Data` 和 `Monthly` 是组合，而 `Age` 和 `Expenses` 是设置为内置类型的叶子：



DMN Expression				
tApplicantData (ItemDefinition)				
tApplicantData	Age : number	Type in Allowed Value Enumerations..		
	MaritalStatus : string	"S","M"		
	EmploymentStatus : string	"EMPLOYED","SELF-EMPLOYED","STUDENT","UNEMPLOYED"		
	ExistingCustomer : boolean	true, false		
	Monthly	Income : number	Type in Allowed Value Enumer...	
		Repayments : number	Type in Allowed Value Enumer...	
Expenses : number		Type in Allowed Value Enumer...		

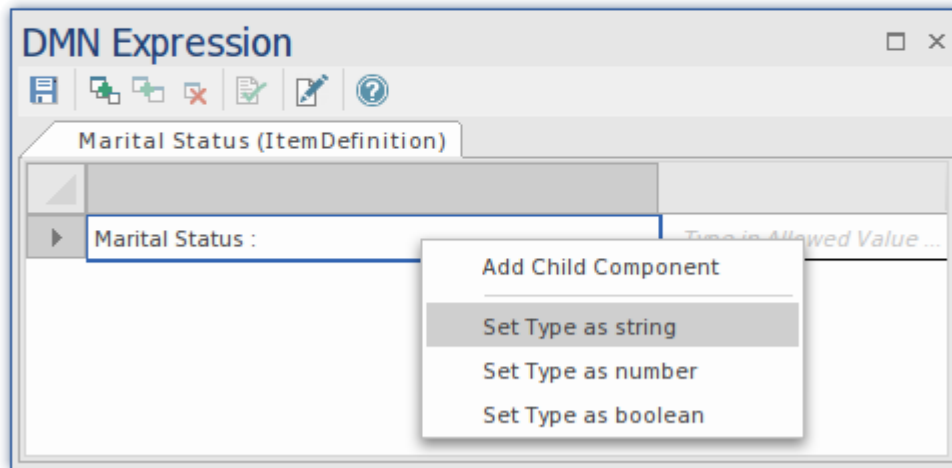
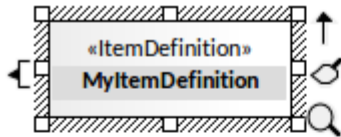
FEEL 语言具有以下内置类型：

- 数字
- **string**
- 布尔值
- 天数和持续时间
- 年和月持续时间
- 时间
- 日期和时间

注记：Enterprise Architect 支持 'number'、'string' 和 'boolean' 进行模拟。

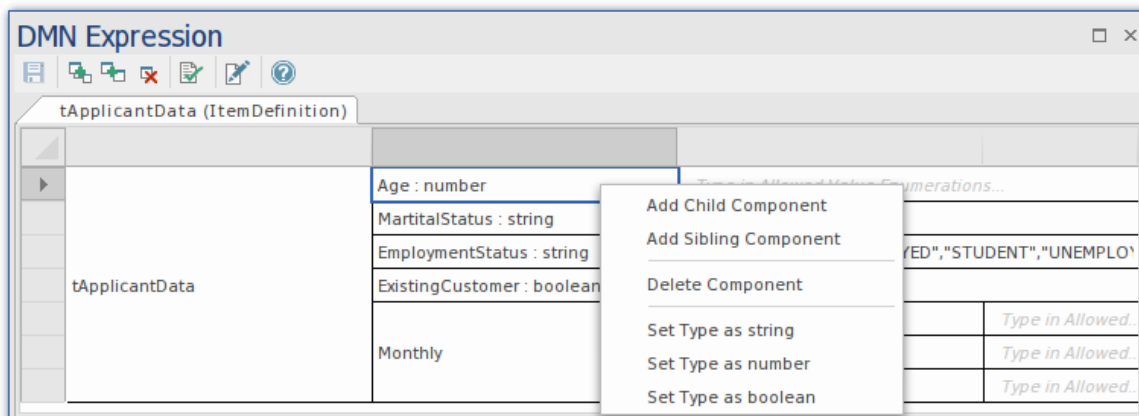
要为“叶子”`ItemDefinition` 设置类型，您可以使用以下三种方法之一：

- 在 DMN 表达式窗口中选择合适的上下文菜单选项（推荐）



- 类型: 'string', 'boolean' or 'number' 在单元DMN 表达式窗口中的名称后面
- 类型 'string'、'boolean'或'number'作为类型窗口中属性'Type'的值

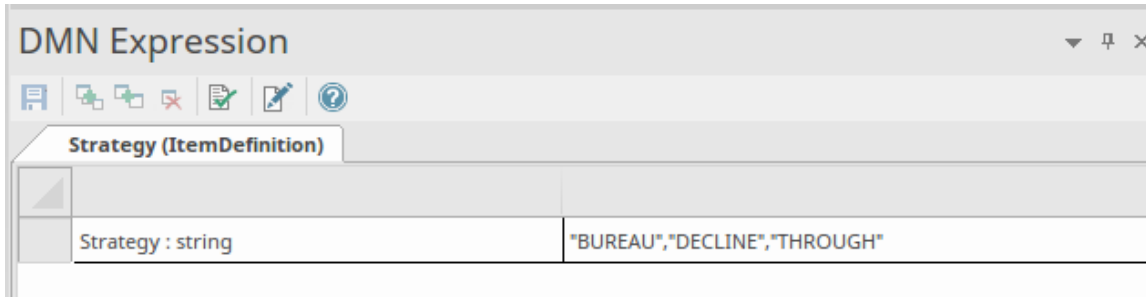
对于复合上下文，时间菜单还提供了创建子组件或同级组件或删除所选项目的选项：



允许的值枚举

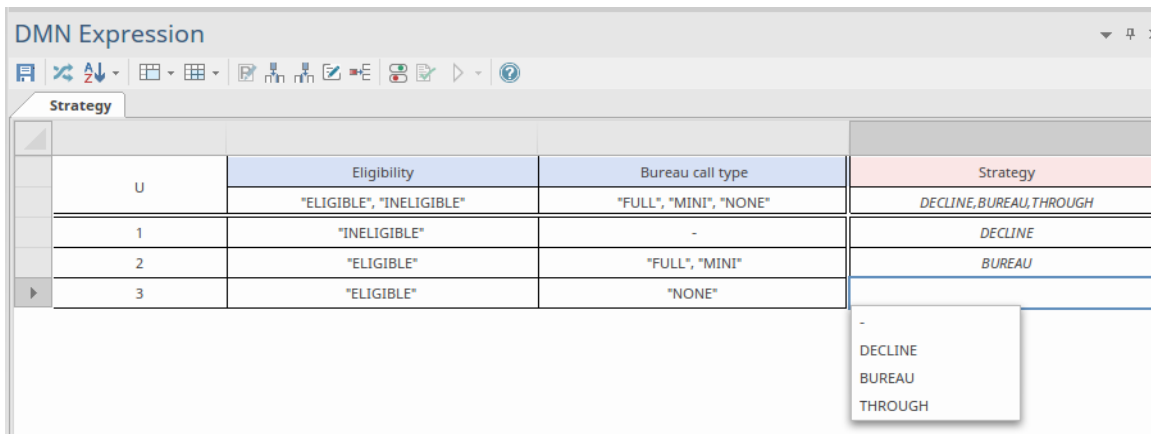
在为决策定义数据输入时，通常希望限制输入的允许值集。例如，您可能希望将婚姻状态的允许值限制为两个选项，*单身*和*已婚*。

您可以为 *ItemDefinition* 的任何叶组件指定允许的值。最初，叶组件的数据字段包含 *Allowed Value Enumerations* 中的文本类型。您只需使用允许的值输入此文本。例如，*ItemDefinition Strategy* 具有三个允许值 - *BUREAU*、*DECLINE* 和 *THROUGH*。



Strategy (ItemDefinition)	
Strategy : string	*BUREAU*, *DECLINE*, *THROUGH*

允许值枚举也用于支持自动完成。当为 *InputData* 元素或引用已定义允许值的 *ItemDefinition* 的输入参数指定值时，用户只需按空格键并从列表中选择一个值。

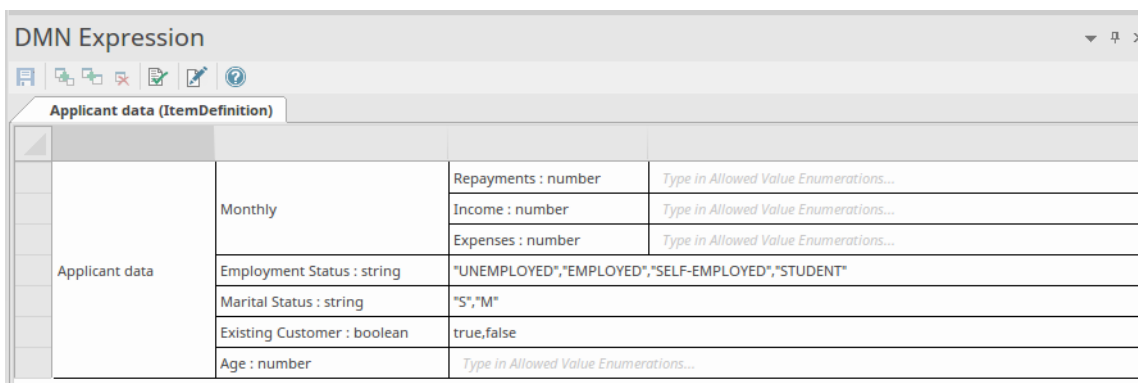


	Eligibility	Bureau call type	Strategy
U	"ELIGIBLE", "INELIGIBLE"	"FULL", "MINI", "NONE"	DECLINE, BUREAU, THROUGH
1	"INELIGIBLE"	-	DECLINE
2	"ELIGIBLE"	"FULL", "MINI"	BUREAU
3	"ELIGIBLE"	"NONE"	

您还可以通过键入要输入的选项的第一个字母来自动完成。

决策表的输入参数和输出子句也支持允许值的规范。这限制了在表中定义规则时可以使用的值，但也允许用户通过按空格键然后选择所需项目来快速填充规则。

A复杂的 *ItemDefinition* 可以包含许多 *Allowed Value Enumerations*；例如：



Applicant data (ItemDefinition)			
Applicant data	Monthly	Repayments : number	Type in Allowed Value Enumerations...
		Income : number	Type in Allowed Value Enumerations...
		Expenses : number	Type in Allowed Value Enumerations...
	Employment Status : string	"UNEMPLOYED", "EMPLOYED", "SELF-EMPLOYED", "STUDENT"	
	Marital Status : string	"S", "M"	
	Existing Customer : boolean	true, false	
	Age : number	Type in Allowed Value Enumerations...	

数据集

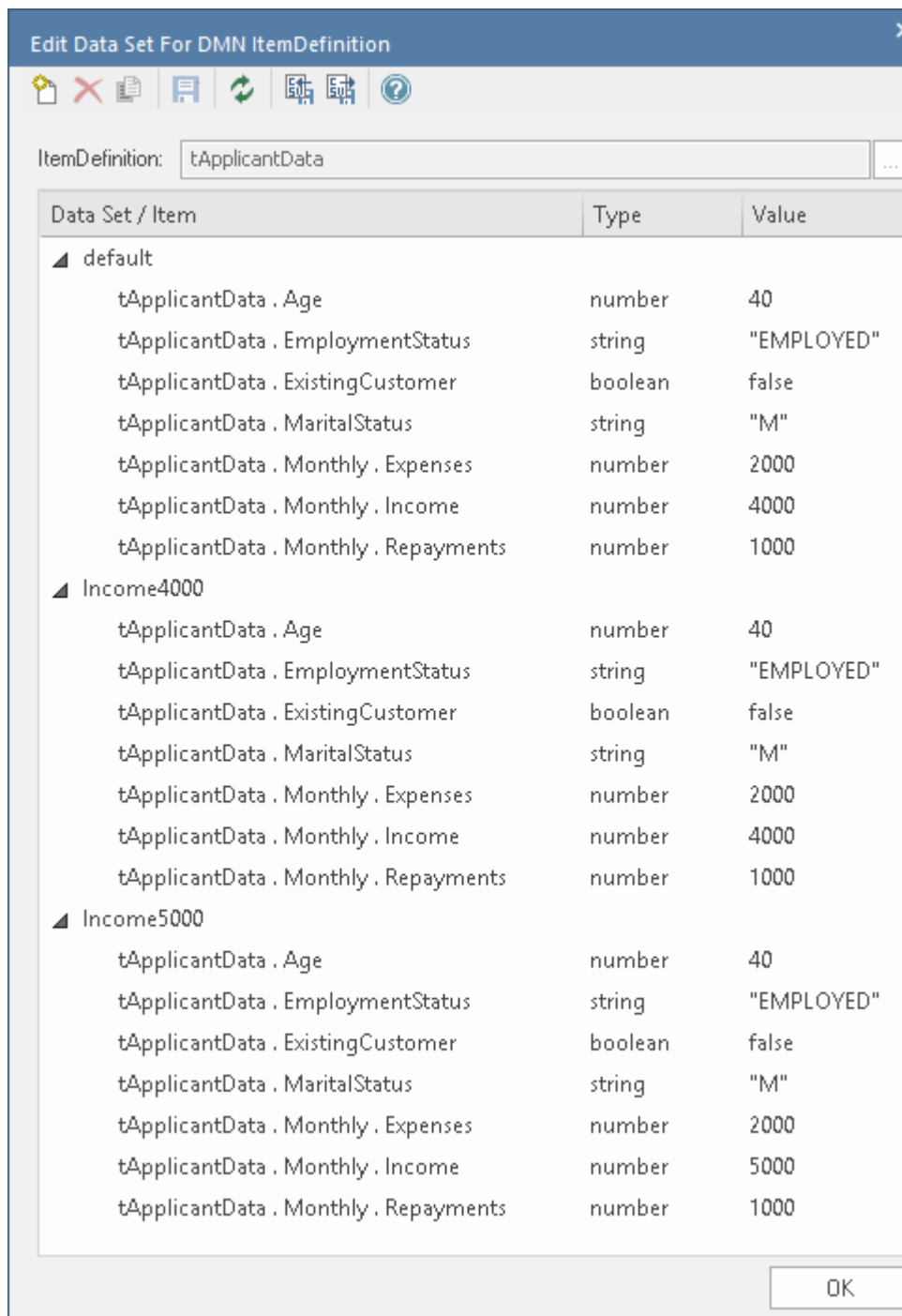
由 `ItemDefinition` 类型化的每个 `InputData` 元素都有一组组件，并且可以定义多个数据集来为这些组件提供不同的值集。有了这个特征，执行 DMN 仿真的用户可以通过选择不同的数据集来快速测试决策的结果。数据集与 `ItemDefinition` 相关联并基于 `ItemDefinition`，但您也可以通过 `InputData` 元素处理它们。

您可以使用“编辑数据集”对话框添加或更新数据集，您可以从 DMN 表达式窗口为 `ItemDefinition` 或 `InputData` 元素调用该对话框。最初，“编辑数据集”对话框显示一组没有值的组件，其名称为“默认”。您可以不设置任何值，也可以提供值；无论哪种方式，您都可以将其用作模板来复制新数据集。您不能删除“默认”数据集。



当您在 DMN 表达式窗口中访问 `InputData` 元素时，“默认”数据集中的值将根据元素的组件显示。然后，您可以单击工具栏中的下拉箭头并从列表中选择任何其他数据集。请记住，如果您保持“默认”数据集不变，您可以创建一个重复的“默认”数据集并分配值到它，并且“默认”设置将在您最初访问 `InputData` 元素时提供值。

您可以复制和删除您创建的任何其他数据集，将数据集导出到 CSV 文件并从 CSV 文件中导入它们。

注记：如果您创建一个数据集但不输入值，则在您关闭对话框时它会被丢弃。



访问

功能区	仿真>决策分析>DMN> DMN 表达式>点击InputData项：  图标
其它	在图表中，双击 DMN InputData元素：  图标。


工具栏选项

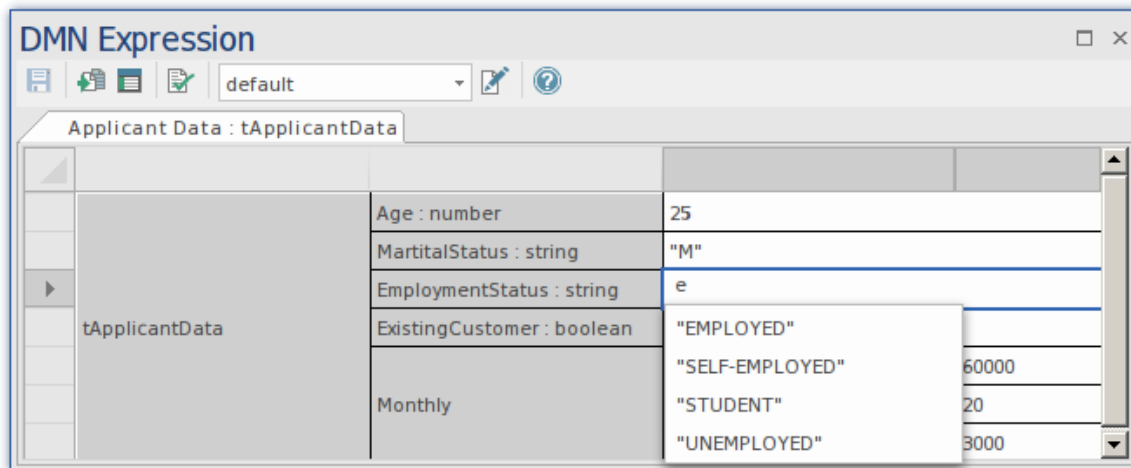
选项	描述
	单击此按钮可创建新数据集。
	单击此按钮可删除选定的数据集。
	单击此按钮可复制选定的数据集。
	单击此按钮可将数据集保存到 InputData。
	单击此按钮可重新加载 InputData 的数据集。
	单击此按钮可从 CSV 文件导入数据集。
	单击此按钮可将数据集导出到 CSV 文件。

使用 DataObjects 交换数据集

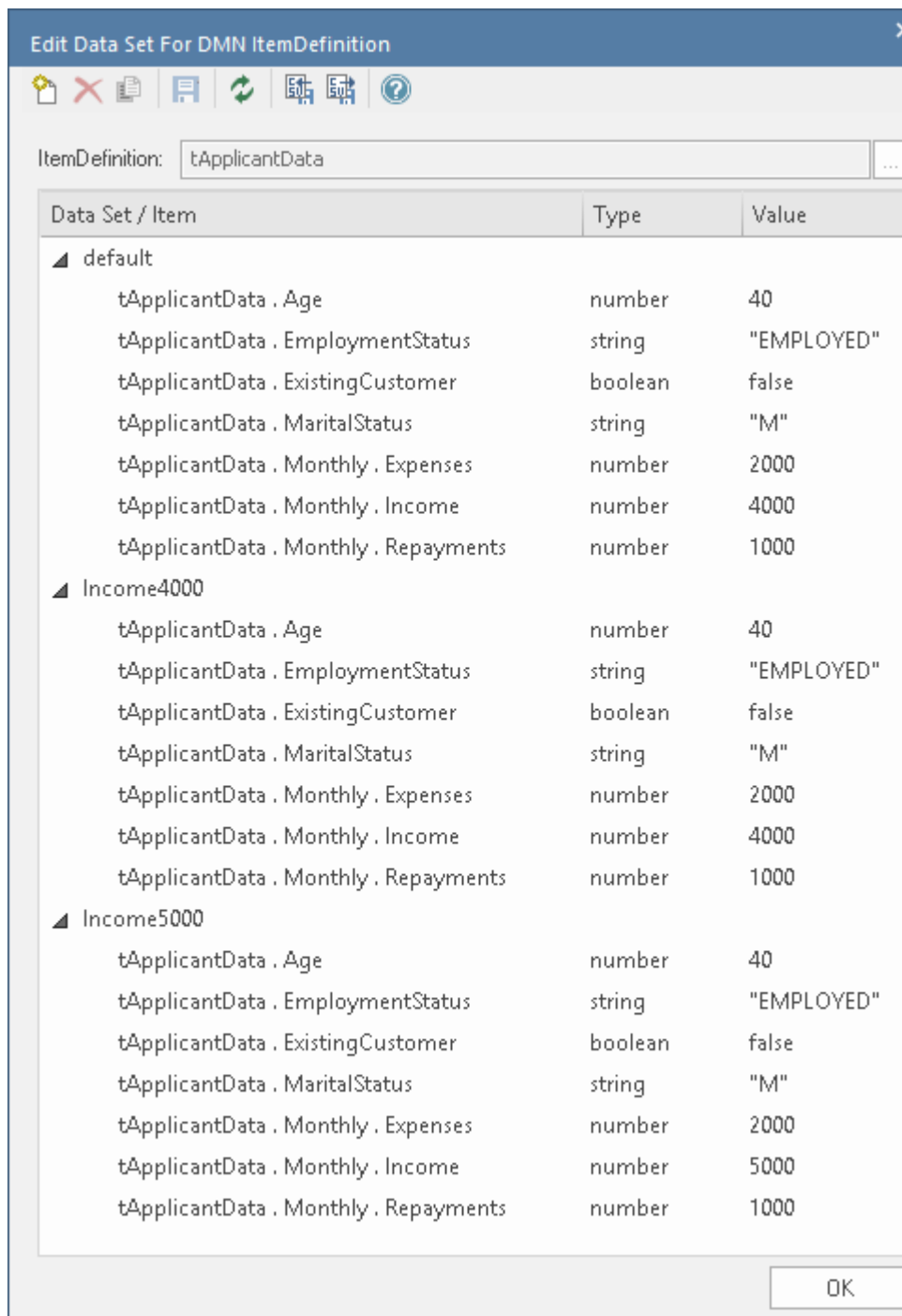
在测试由DMN模型生成的代码时，或者在模拟调用DMN模型的业务流程模型和表示法（BPMN）模型时，您需要一种交换数据集的方法。例如，在DMN模型的BPMN调用中，BPMN DataObject 用于存储将传递给它正在调用的DMN模型的变量集。此 DataObject 需要填充适合 DMN InputData 数据结构的数据，以准备传递给该 InputData object。在测试从DMN模型生成的代码时，使用相同的BPMN DataObject。

本主题描述从DMN数据集创建BPMN数据对象的过程。

数据集存储在DMN InputData元素中，可以使用DMN表达式窗口上的  图标进行访问。



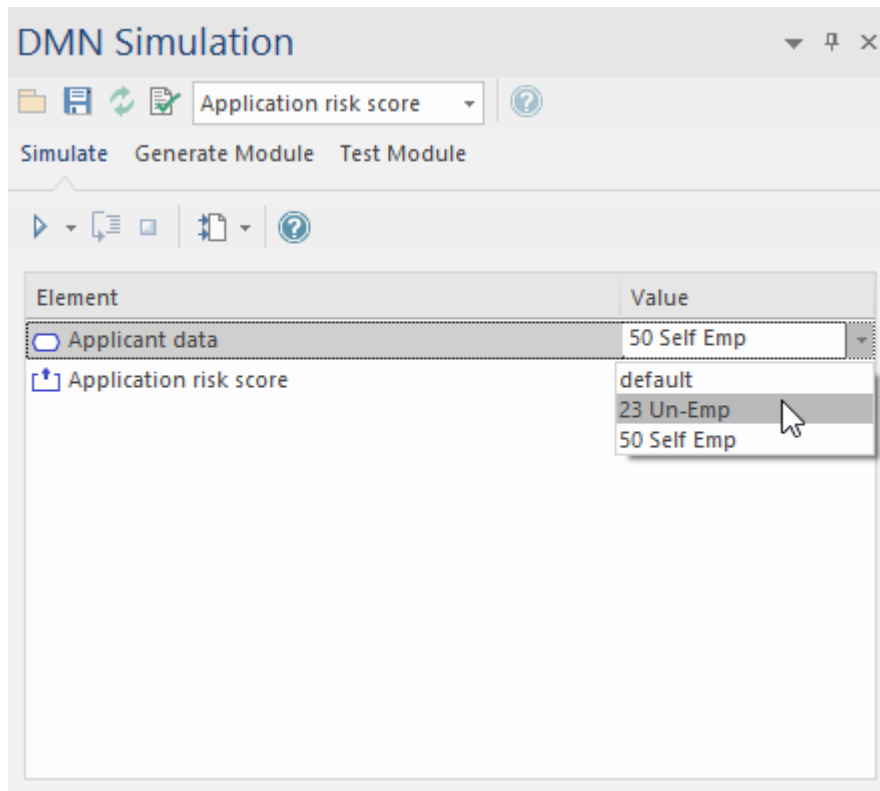
这将打开 InputData 的“编辑数据集”对话框，其中可以包含多组值：




将数据集传输到数据对象有两个选项：

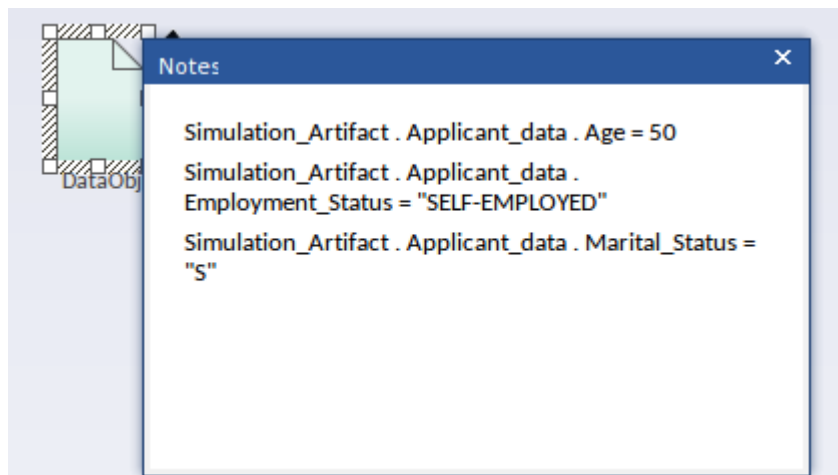
1.直接的

- 在浏览器窗口的一个包下创建一个BPMN DataObject。
- 打开DMN仿真窗口





- 从“值”下拉列表中选择数据集
- 单击 DMN仿真窗口上的  图标；这将打开“选择元素”对话框
- 选择 BPMN DataObject元素
- 点击确定按钮

现在可以在 DataObject 的注记中查看数据集。

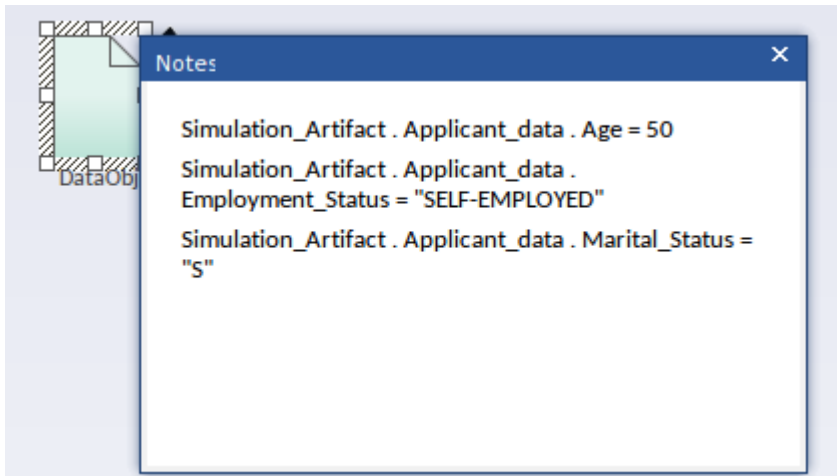


2. 手册

要手动交换此数据集：

- 打开 InputData元素的DMN 表达式窗口
- 单击编辑数据集图标 ；这将打开“编辑数据集”对话框
- 使用 CSV导出的使用图标  将这些详细信息 到文件

CSV 文件中的文本可以作为文本添加到 BPMN DataObject元素的注记中。



服务决策服务

本主题的部分内容已逐字使用或自由改编自 *DMN* 规范，该规范可在以下网址获得：<https://www.omg.org/spec/DMN>。该站点包含对 *DMN* 及其功能的完整描述。

决策决策A决策模型中的一个或多个决策公开为可重用元素，该元素可能由决策模型决策模型任务在外部调用。

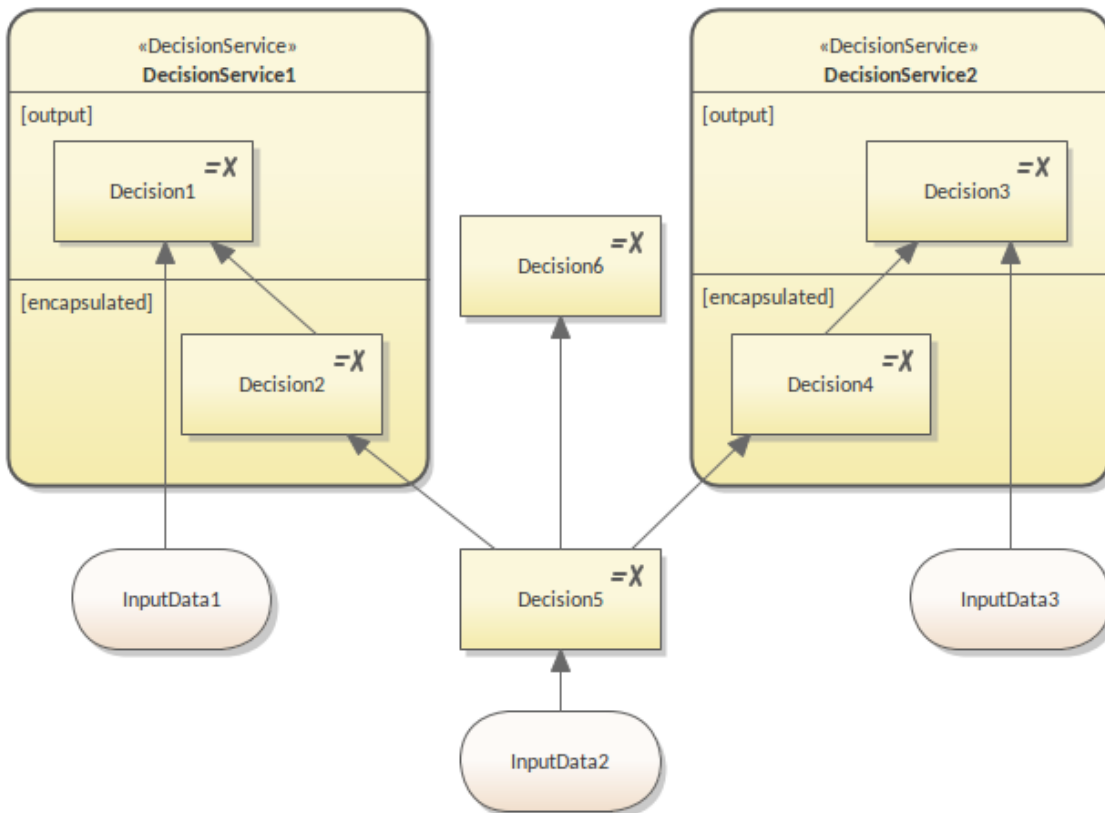
当使用必要的输入数据和输入决策调用决策服务时，它会返回公开决策的输出。

接口服务的决策

决策服务的接口包括：

- 输入数据 - 封装决策所需的所有输入数据的实例
- 输入决策 - 所有输入决策结果的实例
- 输出决策 - 使用提供的输入决策和输入数据评估（至少）所有输出决策的结果

当使用必要的输入数据和输入决策调用决策服务时，它会返回公开决策的输出。



该图显示了一个决策模型，包括六个决策和三个输入数据项。

对于DecisionService1，：

- 输出决策是 {Decision1}
- 输入决策是 {Decision5}，并且
- 输入数据是 {InputData1}

由于Decision1需要Decision2，而Decision2并不作为输入提供给服务，因此服务也必须封装Decision2；因此封装的决策是 {Decision1， Decision2}。

从图中可以明显看出，DecisionService1的任何决策都不需要 Decision6、Decision3、Decision4 和 InputData3。InputData2 呢？虽然Decision5 需要，DecisionService1 需要InputData2，但DecisionService1 实际上不需要

InputData2。这是因为 Decision5 被定义为输入决策。从决策服务的角度来看，我们忽略了输入决策所需的任何决策或输入决策。

对于 DecisionService2，：

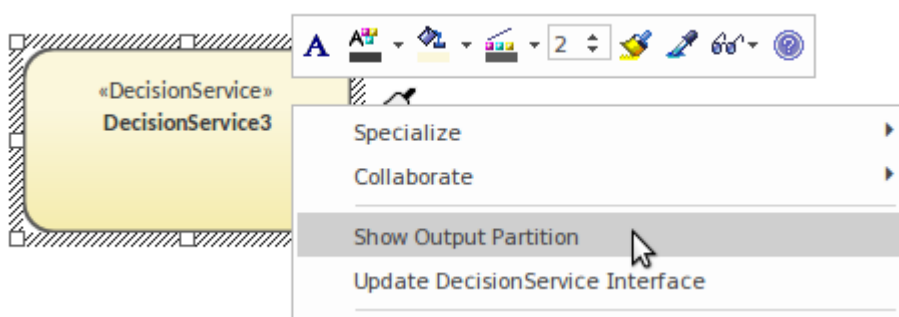
- 输出决策是 {Decision3}
- 输入决策是 {Decision5}，并且
- 输入数据是 {InputData3}

由于 Decision3 需要 Decision4，而 Decision4 并不作为输入提供给服务，因此服务也必须封装 Decision4；因此封装的决策是 {Decision3, Decision4}。

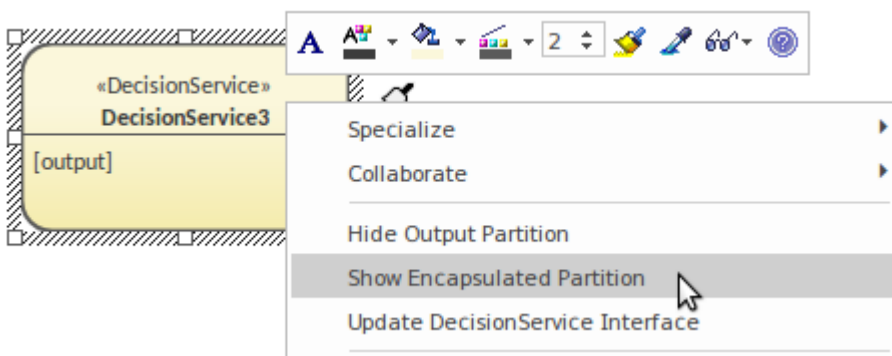
为每个决策服务创建单独的图表是一种很好的做法。这样，图将只包含决策的接口元素和封装的决策；不相关的元素不会出现在图表上。

建模决策服务

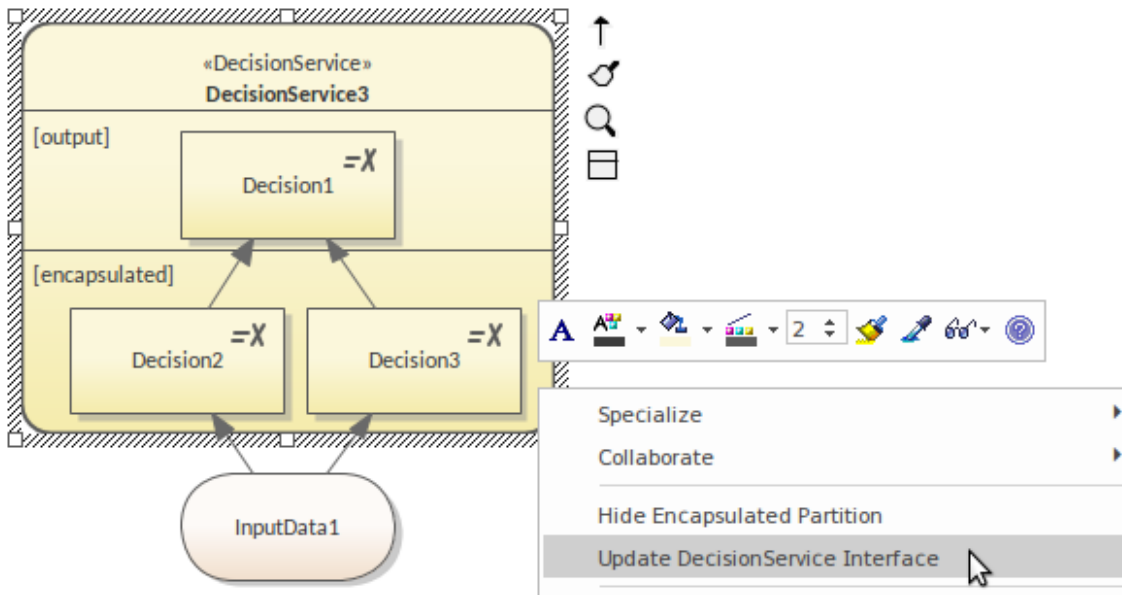
我们可以从图表工具箱的决策页面创建决策服务元素，并从上下文菜单切换 [输出] 和 [封装] 分区。



当显示 [output] 分区时，您只能显示 [encapsulated] 分区。



将决策和输入数据放入正确的分区后，您必须运行上下文菜单中的“更新决策服务接口”命令来更新模型。



重要提示：为了使 DMN 模拟正常工作，请在以下情况下更新决策服务接口：

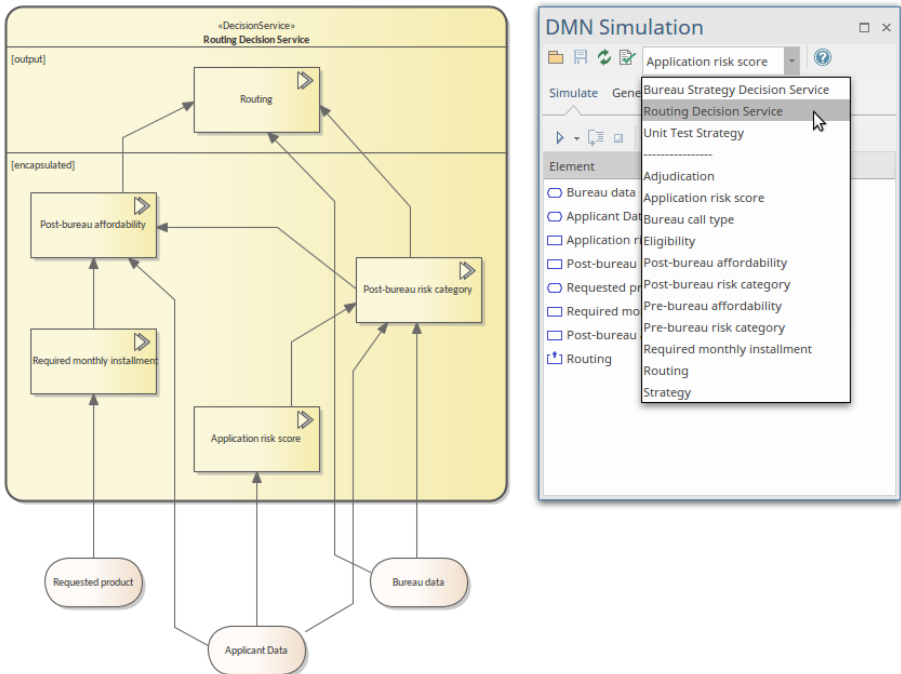

- 显示/隐藏决策服务分区
- 向决策服务添加决策
- 从决策服务中删除决策
- 在分区之间移动决策
- 添加/删除决策服务输入：输入数据或输入决策

仿真决策服务

可以对决策服务执行模型模拟。

决策仿真

要对决策服务执行模型模拟，请执行以下步骤：

节	描述
1	<p>将 仿真工具组件”页面上的 工具箱组件”元素工具的 配置组件”图表中，在工作仿真窗口中双击打开。</p>  <p>默认情况下，所有决策服务元素和每个单一决策都会在对话框工具栏的下拉字段中列出以供选择。</p>
2	<p>选择运行模拟的决策服务元素。在示例中，我们选择了“路由决策服务”，因此在模拟列表中加载了三个输入数据项和五个封装决策（包括一个输出决策）。</p> <p>重要：此列表来自决策服务的内部数据；每当决策服务模型图发生更改时，请确保运行上下文菜单中的“更新决策服务接口”命令。通过单击 DMN 仿真窗口工具栏上的“刷新”图标（左起第三个）重新加载决策模型。</p>
3	<p>输入数据和决策的执行顺序正确。例如，“应用风险评分”将在'Post-bureau risk category'、'Post bureau affordability'和'Routing'之前执行。对于每个输入数据元素，单击“值”字段中的下拉箭头并选择数据集以提供输入数据值。</p> <p>验证输入数据和决策，并使用DMN 表达式窗口进行任何必要的更正。</p> <p>在 DMN 仿真窗口上，单击保存图标和工具栏上的  按钮。</p>
4	<p>运行时执行结果显示在列表和图表中。您也可以单击工具栏上的“Step-through”图标来调试 DMN 模型。</p>

The diagram illustrates a DMN for a 'Routing Decision Service'. It features several nodes: 'Requested product' (input), 'Applicant data' (input), 'Required monthly installment' (output), 'Application risk score' (output), 'Post-bureau affordability' (output), 'Post-bureau risk category' (output), and 'Routing' (output). The 'Routing' node is the final decision point, which is 'ACCEPT'. The 'Post-bureau affordability' node is 'true', and the 'Post-bureau risk category' node is 'VERY LOW'. The 'Required monthly installment' is 668.9574698454752, and the 'Application risk score' is 133. The simulation window shows the following values:

Element	Value
Bureau data	{ "Bankrupt": false, "..." }
Applicant data	{ "Age": 40, "Employ..." }
Requested product	{ "Amount": 100000, "..." }
Application risk score model	133
Application risk score	133
Post-bureau risk category table	"VERY LOW"
Post-bureau risk category	"VERY LOW"
Installment calculation	668.9574698454752
Required monthly installment	668.9574698454752
Credit contingency factor table	0.8000000000000004
Affordability calculation	true
Post-bureau affordability	true
Routing rules	"ACCEPT"
Routing	"ACCEPT"

A好的做法是在调试时保持DMN 表达式窗口打开。运行（如决策表、盒装上下文、状态表达式或调用）的运行时间将显示决策决策封装或调用的决策业务知识模型的逻辑细节。

开始


决策模型和符号 (DMN) 是由物件管理组 (OMG) 发布和管理的标准。

本主题的部分内容已逐字使用或自由改编自 *DMN* 规范，该规范可在 *OMG DMN* 网页 (<https>) 上找到。*DMN* 及其功能完整描述可以在 *OMG* 网站上找到。

DMN 的目的是提供模型决策所需的结构，以便组织决策可以很容易地在图表中描述，由业务分析师准确定义，并且 (可选) 自动化。它还旨在促进组织之间决策模型的共享和交换。

选择蓝图

Enterprise Architect 将工具的广泛特征划分为蓝图，确保您可以聚焦于特定任务并使用您需要的工具，而不会分散其他特征的注意力。要使用决策模型和符号特征，您首先需要选择此蓝图：

 <perspective name> >需求>决策建模

设置蓝图确保决策模型和符号图、它们的工具箱和蓝图的其他特征将默认可用。

示例图表

示例图提供了对该主题的可视化介绍，并允许您查看在指定或描述决策建模方式时创建的一些重要元素和连接器。决策图将引入需求决策表、知识来源、输入日期等要素。Enterprise Architect 的大部分功能依赖于模拟或“运行”决策模型以及基于不同数据集预测结果的能力。此功能将在后面的主题中进行描述，但从创建决策需求图开始。

使用 DMN 建模

本主题向您介绍创建决策模型所需的最重要元素。这包括创建决策需求图，描述决策如何相关以及每个决策可能包含哪些输入，包括其他决策。您将了解最重要的元素，包括：决策、业务模型、输入数据项定义、数据集和决策服务。

代码生成和测试模块

本主题向您介绍语言的主要概念，包括其结构、架构以及用于创建决策模型和表示法 (DMN) 模型的元素和连接器。了解语言的意图和结构将有助于分析师创建有意义且富有成效的决策模型。

集成 BPSim 进行仿真

Enterprise Architect 允许运行 (模拟) 决策模型，这使您可以可视化决策的结果。除了这个核心功能之外，您还可以将决策模型与 BPSim 集成，决策是用于模拟 BPMN 图的模拟引擎。在本主题中，您将通过多种不同的方式将 DMN 与 BPSim 集成。

集成到 UML 类元素

在本主题中，您将了解将 DMN 模型与 UML 类元素集成的过程。DMN Module 可以与 UML 类元素集成，因此从类元素生成的代码可以重用 DMN Module 并且结构良好

导入 DMN XML

本主题描述如何从不同的Enterprise Architect存储库或其他 DMN 兼容工具导入 DMN XML 文件。开放标准的承诺之一是能够在不同工具之间共享模型。Enterprise Architect经常成为建模的首选工具，因为它支持的特征和标准的广度。这允许决策模型与战略、需求业务、流程、软件实施元素等相关。

更多信息

本节提供了指向其他主题和资源的有用链接，您在使用决策模型和符号工具特征时可能会发现它们很有用。

更多信息

创建决策模型

在我们描述的模型示例建模决策我们展示了如何使用决策决策表对决策进行建模，其中决策结果是通过在表中找到输入值与表匹配的行来确定的正在考虑的值，给出特定的输出结果。

我们现在将通过逐步完成为 Airline Cabin Upgrade 示例创建决策模型的过程来了解如何在 Enterprise Architect 中创建这样的模型。

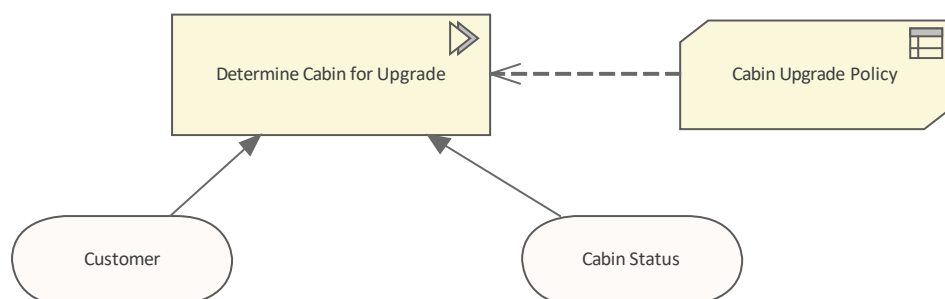
此示例中涉及许多模型元素，例如输入数据元素、用于描述输入数据（定义数据类型）的项目定义、决策元素以及保存决策决策表的业务知识模型元素定义。

创建决策需求图表

这些步骤将指导您创建简单的决策需求图表(DRD)。在本例中，我们将从头开始创建模型，而不是使用模型向导中的模式。

节	描述
1	选择视角 需求 决策建模”。 (会显示首页的 从模式创建”选项卡 (模型向导) ，但我们不会在本示例中使用它。)
2	创建一个新的 DMN 图。名称为 “Airline Cabin Upgrade” 。
3	使用图表工具箱，在图表上放置决策元素。选择 调用”作为类型——我们将使用该元素 调用”来自业务知识模型元素的决策。名称元素 确定升级舱” 。
4	在图表上放置一个名称元素。将这个元素为 顾客” 。
5	在图上放置另一个元素元素。名称为 客舱状态” 。
6	在图上放置一个业务知识模型元素模型。选择类型 决策决策表” 。名称为 元素升级政策” 。
7	从决策 确定升级客舱”到输入数据 顾客” ，划一个 信息需求”连接器。
8	从决策 确定升级客舱”到输入数据 划状态”的 信息需求”连接器。
9	从决定 确定升级客舱”到BKM 客舱升级政策” ，划一个 知识需求”连接器。

在这个阶段，我们应该有一个简单的 DRD，如下所示：



我们现在可以指定构成此模型的每个元素的详细信息。

定义决策表

双击业务知识模型元素“舱室升级政策”，弹出“DMN 表达式”窗口，显示为空的决策表。这是我们定义客舱升级政策规则的地方。

Cabin Upgrade Policy		Input Parameter Values for Simulation	
	(Input 1, Input 2)		
U	Input 1	Input 2	Output 1
1	-	-	-
2	-	-	-
3	-	-	-

默认情况下，新的决策表创建有两个输入列和一个输出列、一个标题行和三个空规则行。

表格中最左边的一列显示了表命中策略，并为规则编号。默认情况下，命中策略是 'U' 代表 'Unique'。这是我们将用于示例的策略，因此您无需更改此列标题。

有关命中策略的更多信息，请参阅决策决策表命中策略帮助。

决策表输入和输出的名称和定义类型

节	描述
1	在“DMN 表达式”窗口的工具栏上，单击“编辑参数”按钮  。将显示“编辑参数”对话框。
2	将参数名称“Input 1”替换为“Num of Pax Overbooked”。如有必要，单击“类型”下拉箭头并将此参数的类型设置为“数字”。
3	将参数名称“Input 2”替换为“Num of Flights in Last Month by Pass”。将此参数的类型也设置为“数字”。关闭“编辑参数”对话框。
4	编辑将为列1评估的输入表达式。 选择标题单元（包含文本“Input 1”）然后再次单击或按 F2 进入“编辑”模式。选择所有单元文本，然后按空格键。显示输入参数列表。单击“超额预订人数”，然后按“Enter”。第1列的表达式设置为“Num of Pax Overbooked”。 注记： 为每一列计算的输入表达式通常只使用相应的输入参数；但是，您可以使用复杂的表达式。
5	右键单击第1列表表达式并检查其数据类型是否设置为“数字”。
6	编辑将为第2列计算的输入表达式。 选择所有文本，然后按空格键。显示输入参数列表。选择“机票上个月的航班数量”，然后按“Enter”。

	第 2 列的表达式设置为 “通行证上个月的航班数量”。
7	右键单击第 2 列表达式并将其数据类型设置为 “数字”。
8	编辑决策表输出的名称。 将 “输出” 替换为 “Upgrade Cabin”，然后按 “Enter”。
9	设置决策输出的数据类型。 右键单击输出列标题并选择 “string”。
10	设置决策输出的允许值。 在输出列单元正下方（但在第 1 行上方）的单元中，定义输出的允许值。输入 “业务类，第一类”。 注记：由于数据类型已指定为 “string”，因此不需要在值周围加引号。


定义决策表的规则

在单元表中输入值以匹配此图像。

Cabin Upgrade Policy				
Input Parameter Values for Simulation				
(Flights in the last month, Number of Pax Overbooked)				
U	Flights in the last month	Number of Pax Overbooked	Upgrade Cabin	Annotation
			Business Class, First Class	
1	<=1	<=2	Business Class	
2	<=1	(2..8]	Business Class	
3	<=1	>8	First Class	Start Filling First Class when heavily overbooked
4	(1..5]	<=2	Business Class	
5	(1..5]	(2..8]	Business Class	
6	(1..5]	>8	First Class	
7	>5	<=2	Business Class	
8	>5	(2..8]	Business Class	
9	>5	>8	First Class	Reward Frequent Flyers

单击一个单元以选择它，然后再次单击以对其进行编辑。

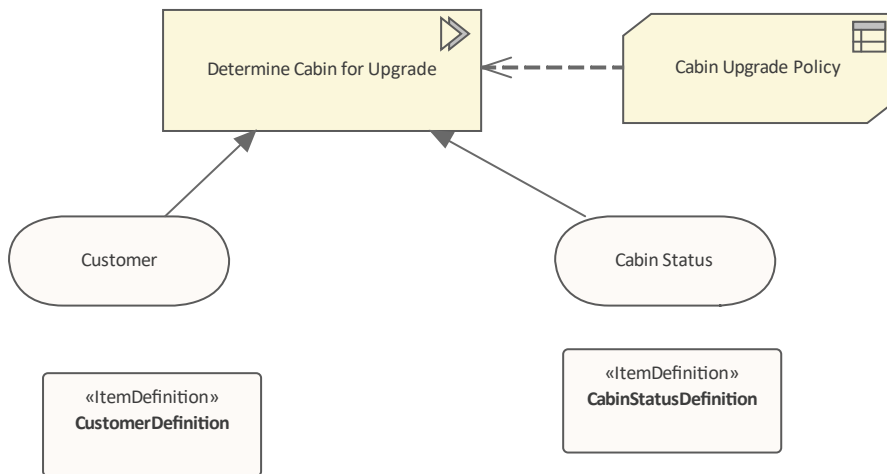
您可以通过选择要复制的行来复制和粘贴现有规则（Shift+单击添加到选择中），右键单击并选择 “复制”，然后右键单击并选择 “附加”。

完成规则编辑后，单击保存按钮 。

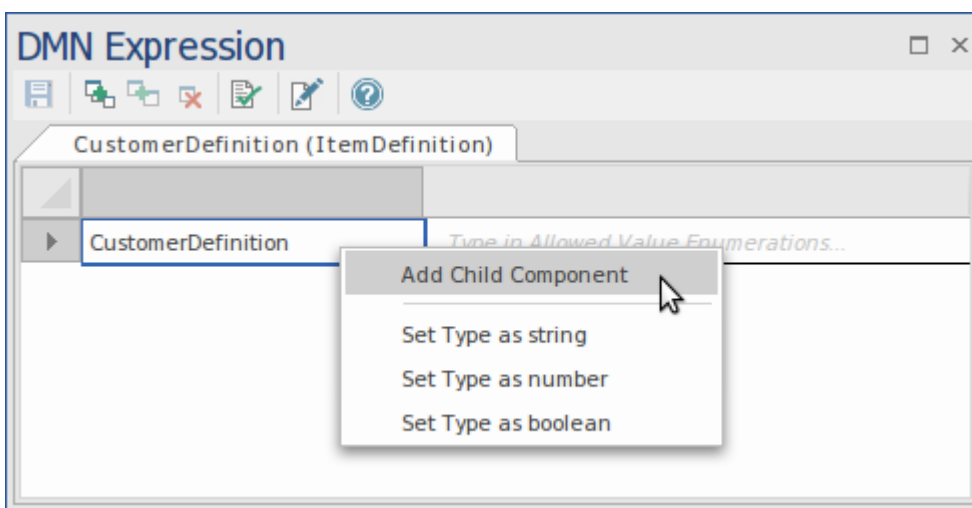
最后，单击验证按钮 ，以检查规则表中的错误。

创建 ItemDefinition 元素

将两个 ItemDefinition 元素添加到图表中，每个 InputData 元素一个。名称一个元素 “CustomerDefinition”，另一个元素 “CabinStatusDefinition”。



双击名为“CustomerDefinition”的 ItemDefinition 以编辑定义。显示DMN 表达式窗口。

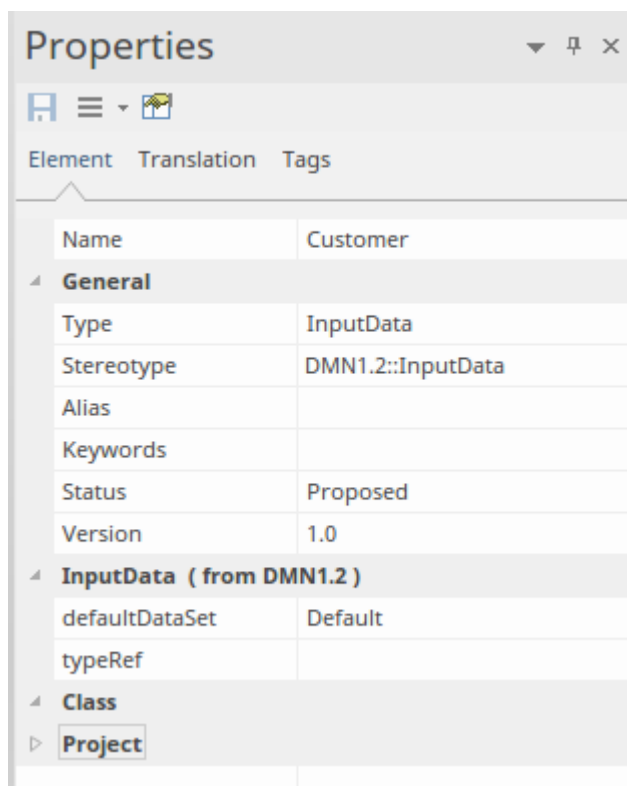


右键单击单元“客户定义”并选择“添加子部件”。用“上个月的航班数量”改写子组件的名称，用“数字”改写其数据类型。单击“保存”按钮保存更改，然后关闭窗口。

同样，双击名为“CabinStatusDefinition”的 ItemDefinition，添加一个名为“Num of Pax Overbooked”的子组件，并将其数据类型设置为“number”。保存更改并关闭窗口。

为每个 InputData 元素指定数据类型

选择 InputData 元素“顾客”。在属性窗口中，选择属性“属性”并单击 按钮。



选择顾客'顾客定义' 作为类型。点击 确定”。

同样，指定 客舱状态定义”作为 状态”的类型。

指定决策元素的输入

双击决策元素'Determine Cabin for Upgrade'

在DMN 表达式窗口中，找到第一列中包含文本 Num of Pax Overbooked”的表行。单击该行第二列的单元，然后按空格键。A可能的输入值列表。选择 客舱状态”。超额预订人数”，然后按 Enter”。选择写入单元。

对第二个表行 上个月的航班数量”重复此过程，选择 顾客。上个月的航班数量”。

单击保存按钮。


单击验证按钮。


定义数据集

您的决策模型的 正确性”可以通过使用一系列代表性数据集运行模拟来测试，以验证模型在所有情况下都能产生正确的结果。

您可以使用一系列数据值创建具有不同名称的大量数据集。您可以将其中一个数据集设置为默认值。

我们现在将为每个现在元素创建一个数据集。

节	描述
1	双击 InputData元素 顾客”。 显示DMN 表达式窗口。
2	在DMN 表达式窗口中，单击 编辑数据集”按钮  。


	将显示“编辑数据集”窗口。
3	单击  按钮。 创建一个新的数据集。
4	如果您愿意，请覆盖数据集的名称。 将类型保留为“数字”。输入一个值，例如 3。 点击保存图标和确定按钮。
5	对 InputData 的“客舱状态”重复此操作。输入一个值，例如 4。

添加一个工件物业

仿真配置“模拟箱”工件 in the 图表的工具箱将其中之一也拖放到图表上。

双击它以在“仿真”选项卡上打开 DMN 仿真窗口。

在仿真窗口中，您可以运行已完成的决策模型的模拟。您还可以执行验证、生成代码和生成测试模块。

节	描述
1	在此窗口的工具栏中找到编辑字段。
2	单击此字段中的下拉箭头。 A 列表显示了与决策配置相关的包中的所有决策和决策工件。在这种情况下，“确定升级客舱”是列表中的唯一项目。
3	单击“确定要升级的客舱”。
4	窗口的主体现在显示 InputData 元素和可用作所选决策输入的决策结果。 单击保存按钮。
5	使用“值”  运行使用选定的数据集运行模拟。

集成BPSim进行仿真

DMN 的优势在于它能够通过决策需求图来描述业务需求，并将复杂的逻辑封装在多种表达方式中，例如决策表和 Boxed Context。

同样，BPMN 的优势在于它能够用任务和事件的序列流来描述业务流程，或者用信息流来描述流程的协作。

决策需求图在业务流程模型和决策逻辑模型之间架起了一座桥梁：

- 业务流程模型定义业务流程中需要决策的任务
- 决策需求图定义了在这些任务中要做出的决策、它们的相互关系以及它们对决策逻辑的要求
- 决策逻辑以足够的细节定义所需的决策，以允许验证和/或自动化

决策提供了一个完成决策模型，通过在细节中指定在流程任务中执行的决策来补充业务流程模型。

本主题中演示的两个示例可以从以下位置访问：

- [EA示例模型|模型仿真|BPSim 模型](#)
- [蓝图|业务建模|BP模拟|BPSim 案例研究](#)

BPSim 表达式使用 DMN模型有两种方式：

- DMN 的决策服务 - 由贷款申请进程演示
- DMN 的 BusinessKnowledgeModel - 由交付成本计算证明

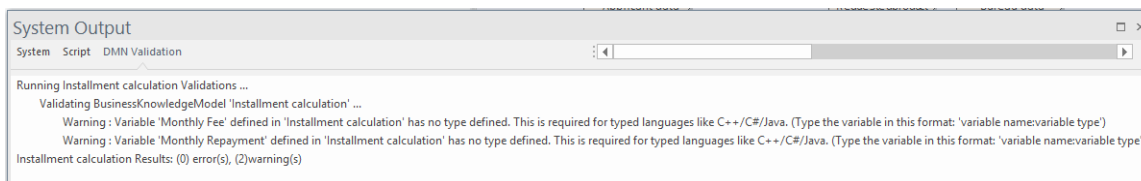
将 DMN模型与 BPSim模型集成的过程包括：

- DMN模型验证、仿真、代码生成和生成模块的测试
- 设置从 BP工件到工件使用依赖关系
- 从 DMN 数据集中生成或更新 BPMN 数据对象
- 在 BPSim 中创建属性参数以用于从网关流出的任务和序列
- 将 DMN 接口绑定到属性Parameters

针对Java等编译语言的 DMN模型验证

当您创建 DMN模型并在Enterprise Architect中对其进行仿真时，驱动仿真的代码是JavaScript；这意味着不需要显式键入变量（变量类型是从分配给它的值推断出来的）。

但是，对于 C++、C#、Java等语言，编译器会报一个变量没有类型的错误。

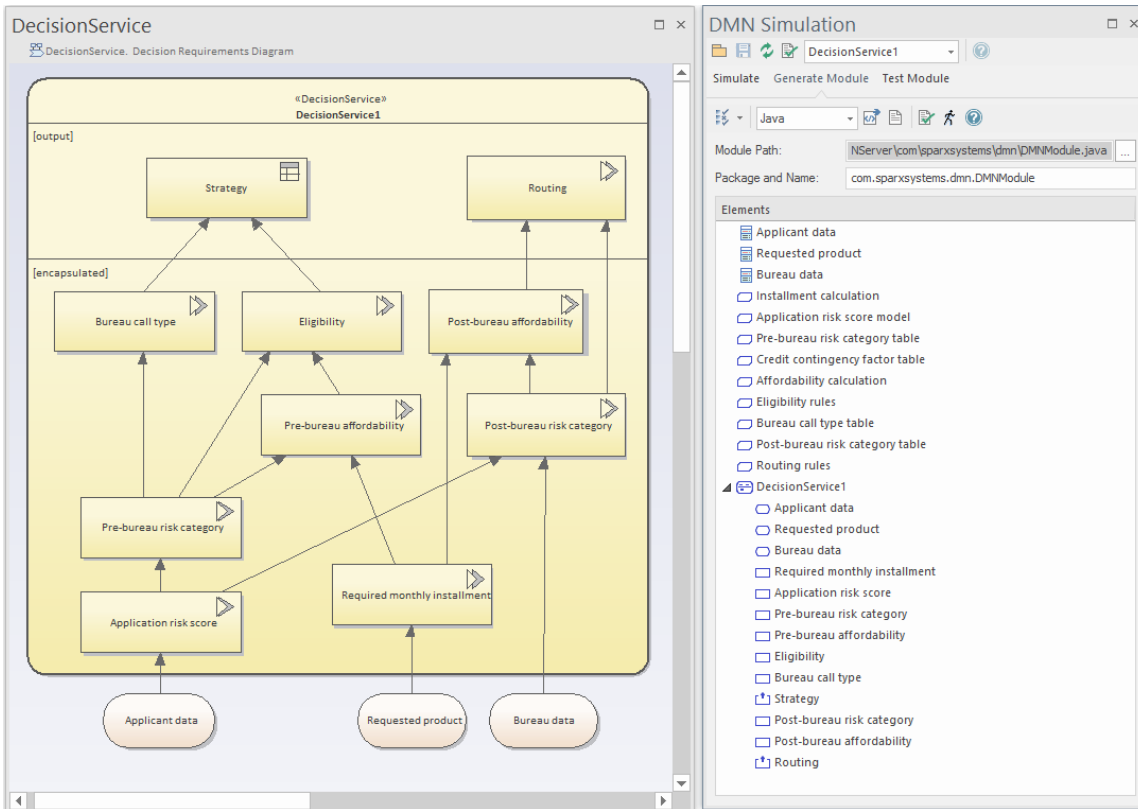


要生成这些语言，您必须在模型上运行验证并使用结果来查找需要其类型集的变量。例如：


- 业务知识模型参数——选择要在DMN 表达式窗口中查看的BKM元素，点击第二个按钮打开“参数”对话框，指定参数类型
- 决策类型 - 选择决策元素，打开属性窗口，对于属性'variableType'从'Value'字段中选择
- 决策表输入/输出 - 在决策表输入/输出子句上，右键显示输出上下文菜单并选择类型
- Boxed Context 变量 - 请参阅[Boxed Context](#)帮助主题

Java中的DMN代码生成

在使用验证修复任何变量类型问题后，我们可以进入 DMN仿真窗口中的“生成模块”页面。

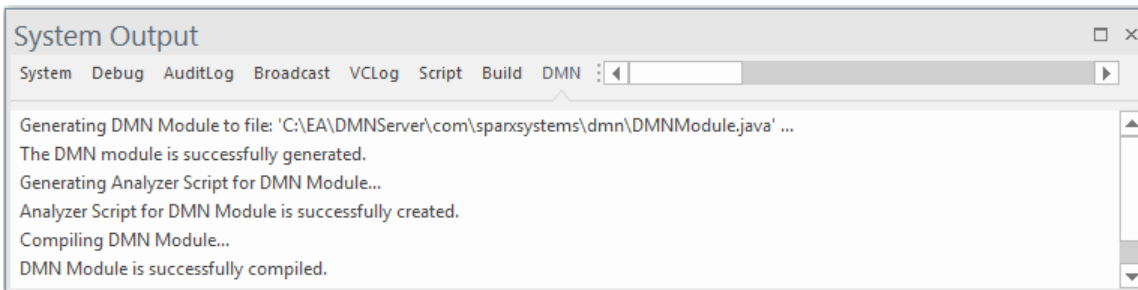




- 在顶部工具栏数据输入字段中选择 *DecisionService1* ; *DecisionService1* 中涉及的所有元素现在都将包含在列表中
- 项目定义和业务知识模型是全局元素
- 输入数据和决策被封装在 *DecisionService* 元素中
- 支持的语言有 C++、C#、Java 和 JavaScript ; 注记对于 JavaScript 生成的 .js 文件与模拟脚本相同 (“仿真” 选项卡 | 运行按钮下拉菜单 | 生成新脚本 (脚本窗口)) 除了省略了仿真相关的代码
- 对于 Java , 模块路径“值必须 包结构匹配; 在本例中 , 必须将 *DMNModule.java* 生成到一个目录以形成一个以 “\com\sparxsystems\dmn\DMNModule.java” 结尾的文件路径 - 您必须手动为现在创建目录结构

单击工具栏上的生成代码按钮 () 。这个例子将使用 Java ; 但是 , C++ 和 C# 是相同的。执行这些操作 :

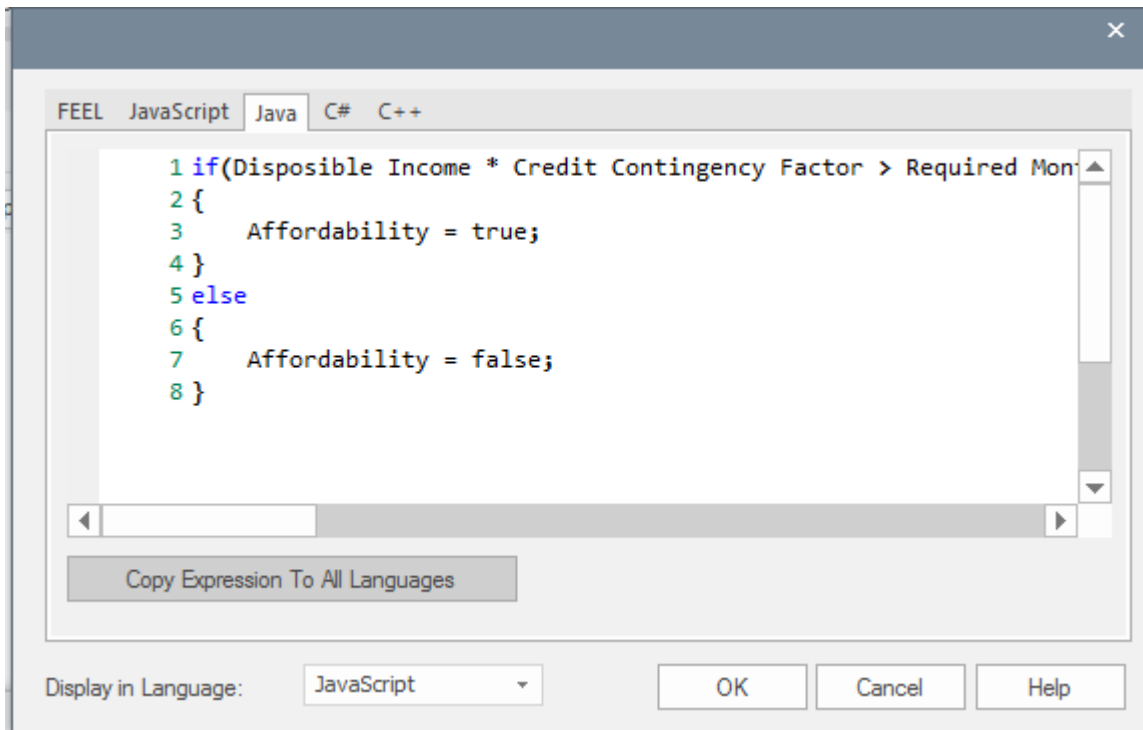
- .java 文件生成到指定的路径
- 为这个工件创建的分析器编译脚本)
- 这个分析器脚本编译脚本被执行
- 在系统输出窗口中报告进度消息

如果模型有效 , 此过程将返回消息 :



如果出现编译错误 , 您可以通过点击工具栏上  按钮旁边的  按钮打开生成的 .java 文件 , 手动修复问题 , 并使用生成的脚本编译 , 直到成功。

编译失败的一个常见原因是语言可以有不同的表达式语法。您可能需要为一种语言提供一个值以覆盖默认值（右键单击 DMN Literal Expression | Edit Expression）。

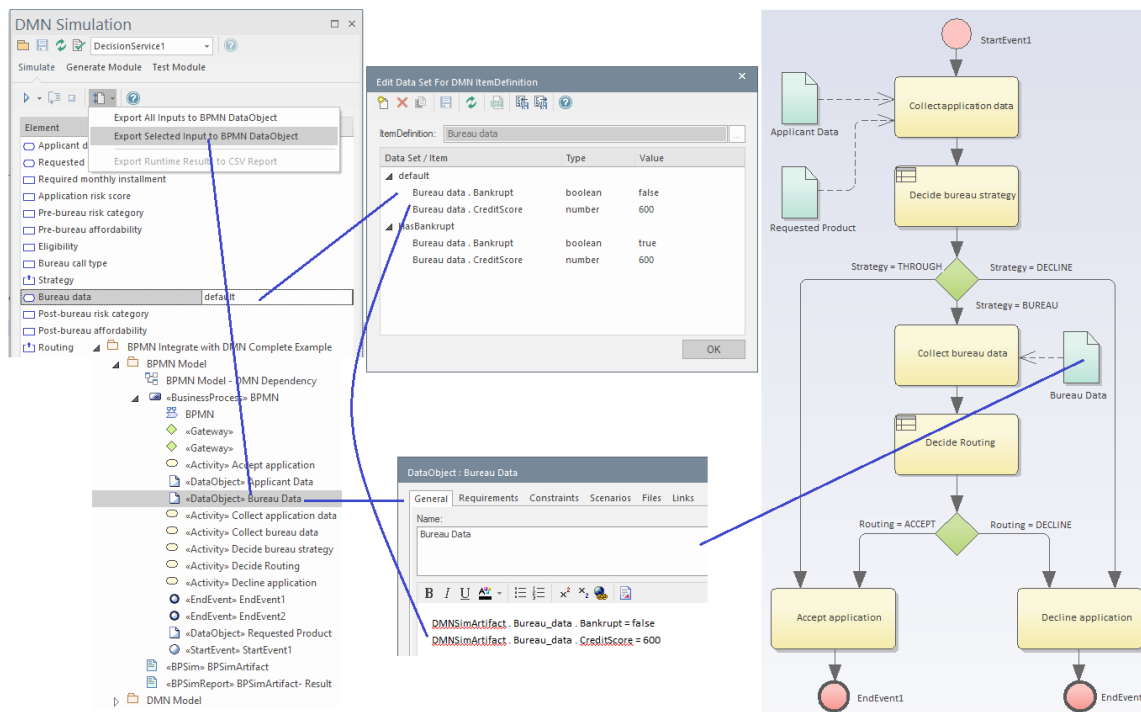


在外部使用之前测试DMN Modules

将模型生成到 java 代码并成功编译后，我们现在想要：

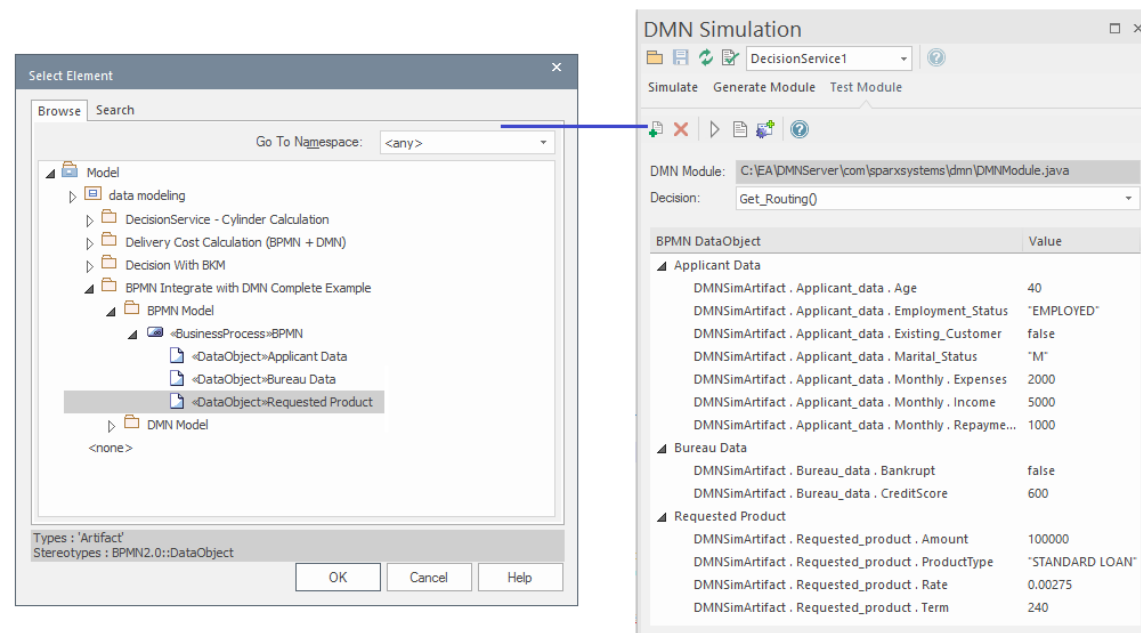
- 测试这个模块的正确性
- 为其提供输入
- 获取输出决策值

生成BPMN数据对象

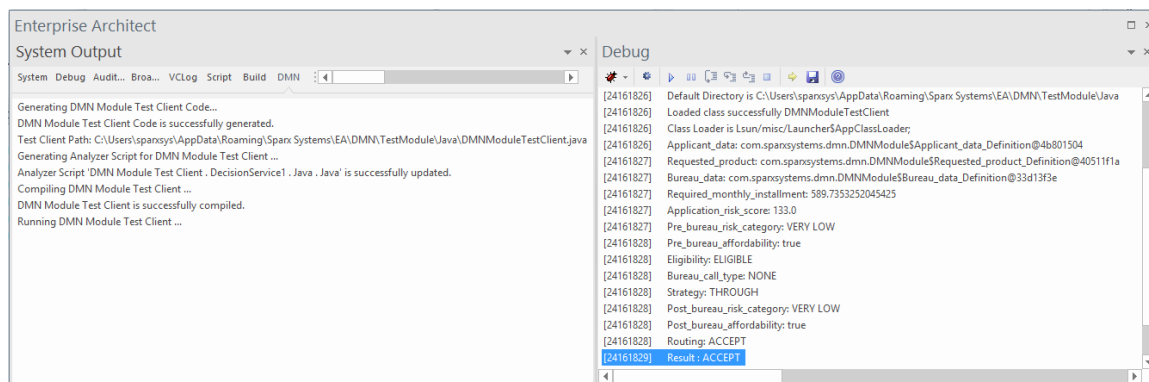


所选数据集携带的数据将生成到BPMN DataObject的 注记”字段。

- 单击 按钮 (生成模块”选项卡工具 右侧第二个) 打开 测试模块”选项卡



- 单击工具栏上的 以选择输入 BPMN DataObject 元素
- 从 决策”组合框中选择可用的输出，例如决策(), 然后单击工具栏上的运行按钮



执行结果将显示在调试窗口中。也可以打开测试模块文件，在行上设置断点，在DMN模块中调试，进行行级调试。

我们强烈建议您使用此窗口测试您的 DMN 模块，以确保 DMN 模块在给定输入（来自 BPMN 数据对象）下正常工作，并且可以成功计算输出结果。

笔记模块路径：工件保存在 DMN 的 'Filepath' 属性中。

现在，是时候将 DMN 模块与 BPSim 模型集成了。

第一步是在工件和工件仿真之间建立使用依赖关系。



注记：如果需要，A 工件可以使用多个 DMN 模块。这只是简单地将所有工件支持的图表放在这个图表上，并从该图表中绘制出从 BP 到每个工件仿真的连接工件。

这些帮助主题提供了使用这些方法的两个示例。看：

- 示例：将 DMN 决策服务集成到决策数据物件和属性参数中
- 示例：将 DMN 业务知识模型集成到属性参数中

示例：将 DMN 决策服务集成到决策数据物件和属性参数数中

BPSim模型向导模型中提供了将DMN决策服务集成到决策中的示例。

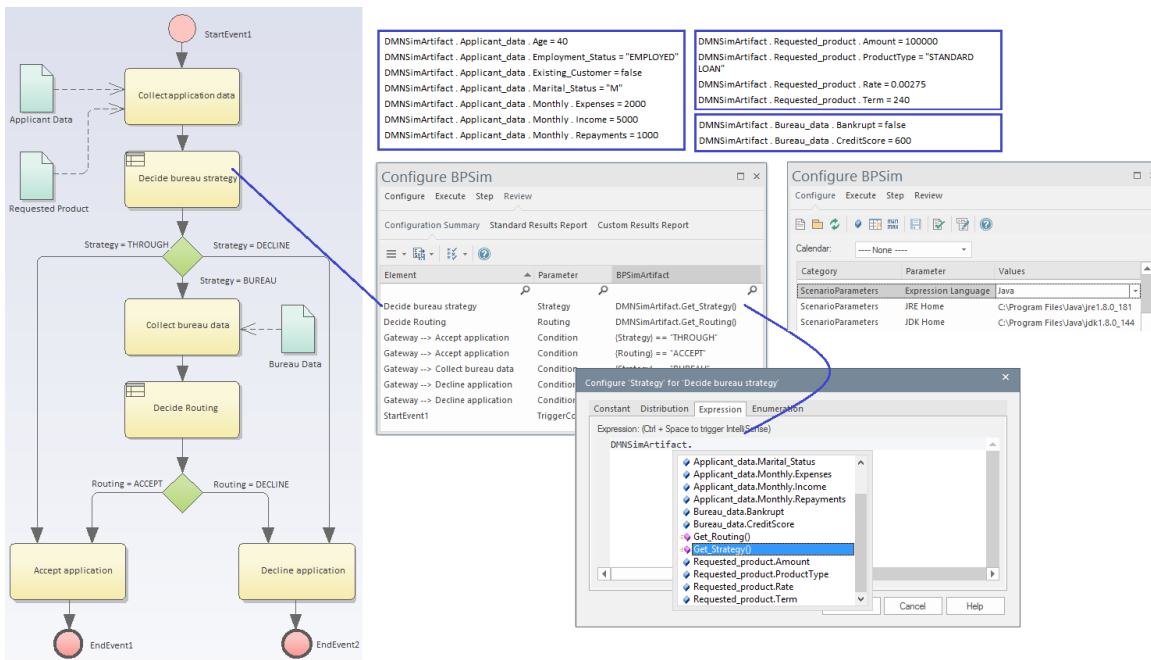
要访问这个：

- 将蓝图设置为业务建模>蓝图
- 打开模型向导（首页'Create from模式' tab）
- 从 BPSim 案例研究组中选择完全 *Integrate with DMN* 完整示例
- 单击创建模型按钮。

这将创建配置为模拟从 BPMN 模型调用 DMN模型的 BPMN 和 DMN模型。

注记：为了集成DMN Module，Expression Language 必须使用Java并且JRE 和JDK 必须正确配置（Java最低版本为1.7）。请参阅帮助主题帮助[BPSim Business Simulations](#)中的安装 *BPSim* 执行引擎。

在这个 BPMN 图中，有三个 DataObjects (aqua) 连接到 BPMN 活动。这些 DataObject 元素携带从 DMN仿真窗口生成的输入数据。



- 当模拟运行时，它会在模拟令牌通过时自动加载所有连接到任务的DataObjects
- 第二业务规则任务“决策局策略”配置为将属性“策略”设置为值 `DMNSimArtifact.Get_Strategy()`；您无需输入 - 按 `Ctrl+Space` 可帮助您编辑表达式

设置好这些后，单击 执行“选项卡并模拟模型。然后您可以查看报告或进入“节”页面对 BPSim模型进行分步调试。

示例：将 DMN 业务知识模型集成到属性参数中

在某些情况下，您可能只想设计一个决策表以在 BPMN 模型中使用。如果是这样，则无需通过创建决策服务决策、输入数据甚至项目定义的过程，因为业务知识模型 (BKM) 可以直接接口。

BPSim 模型中提供了将 DMN BKM 集成到模型向导中的示例。

要访问这个：

- 将蓝图设置为业务建模 > 蓝图
 - 打开模型向导 (首页 'Create from 模式' tab)
 - 从 BPSim 案例研究组中选择 BPMN 与 DMN 集成 - 交付成本计算
 - 单击创建模型按钮
1. 使用参数创建一个简单的业务知识模型作为决策决策表 (您也可以创建其他表达式，例如 Boxed Context 或模型 Expressions)，然后对逻辑 (输入子句、输出子句、规则) 进行建模并对其进行测试 (输入参数值“用于仿真的选项卡在 DMN 表达式窗口上)。

(Total Weight, Amount)			
U	Amount	Total Weight	Delivery Cost
1	<=200	>40	60
2	<=200	(30..40]	50
3	<=200	(20..30]	40
4	<=200	(10..20]	30
5	<=200	<=10	20
6	(200..300]	>40	30
7	(200..300]	(30..40]	25
8	(200..300]	(20..30]	20
9	(200..300]	(10..20]	15
10	(200..300]	<=10	5
11	>300	-	0

2. 使用知识需求连接器将决策连接到决策。该决策用作许多 BKM 功能的组名；您可以简单地在表达式中输入一个数字，例如 '10'。例如，如果你想生成只有五个 BKM 的 Java 代码 (考虑到你的模型可能有一百多个 BKM)，你可以将这五个 BKM 连接到一个决策，并在决策仿真窗口中选择这个决策，那么所有五个决策将被自动包含。
3. 生成 Java 代码和 (假设一切正确) 编译将成功。
4. 在智能感知配置中，我们简单地使用智能计算器来构造任务“计算交付成本”的表达式。

在本例中，“生成家具价格和重量”任务会为属性“重量”和“价格”生成机值，然后“计算交付成本”任务会将值传递给业务知识模型，结果为带回属性 'DeliveryCost'。

您可以现在执行模拟，并逐步通过调试过程来观察，例如，属性值的变化。

导入 DMN XML


Enterprise Architect支持 DMN 1的导入。1或1.2 XML 文件到项目中，同时包含模型语义和图表交换信息。

访问

在浏览器窗口中，选择要导入 XML 文件的包。然后使用此处列出的方法之一从 DMN 1打开“导入包”。1 XML'对话框。

功能区	发布>模型交换>导入>DMN 1 . 1
键盘快捷键	Ctrl+Alt+I :其他XML 格式 >其它1 . 1

导入DMN 1.1 XML

节,节	行动,行动
1	在“文件名”字段中，输入源文件路径和名称，或单击  图标定位并选择文件。
2	单击导入按钮将文件导入包中。

从 OMG导入示例

1. 在[此链接](#)下载 zip 文件并将其解压缩到您的文件管理器。
2. 浏览文件夹示例/Chapter 11/。
3. 单击文件Chapter 11示例并将其作为 DMN 1导入。1格式文件。

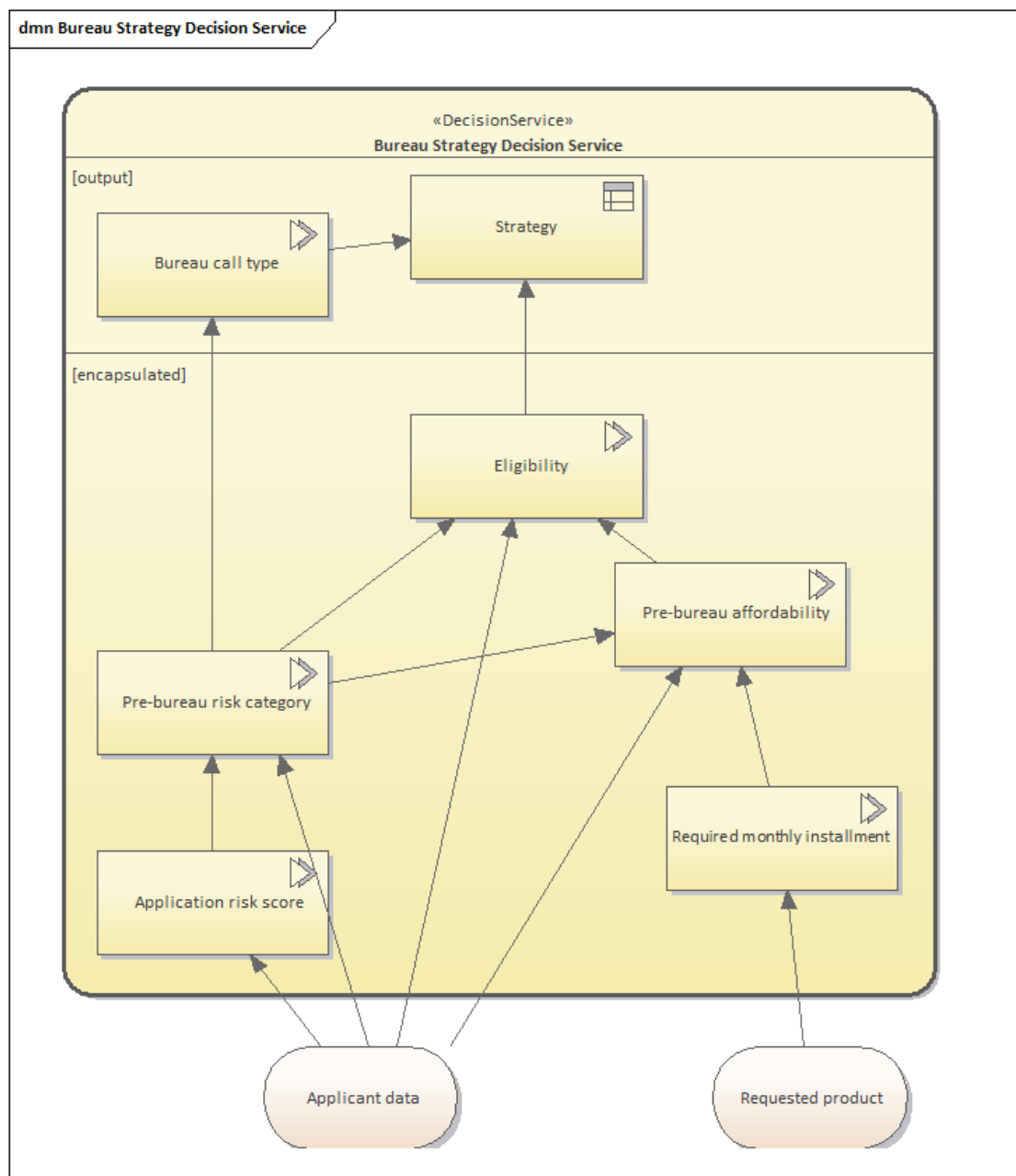
导入这些图表以显示模型的不同视角：

- 所有自动化决策的 DRD
- 用于审阅应用决策点的 DRD
- 决定路由决策点的 DRD
- 决策局战略决策点的 DRD

导入这些图以定义决策服务：

- 局战略决策服务
- 路由决策服务

此处显示了“局战略决策服务”图。它有两个输入数据元素（申请人数据、请求的产品）、两个输出决策（局呼叫类型、策略）和五个封装决策。注记，图中未显示调用的业务模型。



为了从模型生成生产代码，您可能需要运行验证和模拟以确保导入的模型具有正确的表达式。

1. 在任何列出的图表上创建一个工件Sim配置，然后双击它以在 DMN仿真窗口中打开它。
2. 决策服务和决策列在目标下拉字段中。指定目标后，窗口中会列出所有必需的元素。
3. 单击“验证”按钮（工具栏上的第4个）。如果显示任何错误或警告消息，我们建议您在执行模拟之前按照错误或警告说明的指示修复问题。
4. 为输入提供适当的值，然后运行仿真或逐步调试模型。

注记：“Bureau Strategy DecisionService”示例也可在决策模型中找到。从“开始”图中，选择“业务建模 > DMN 示例 > 局战略决策服务”。

决策需求图表

决策需求图 (DRG) 和决策需求图 (决策) 决策的要素是决策、业务知识模型、输入数据、知识源和决策服务。这些元素之间的依赖关系表达了三种需求——信息、知识和权威。

决策需求图表的组成部分

本表总结了决策需求图的所有组成部分的符号。

部件	描述
决策	决策元素表示从大量输入 (输入数据或决策) 中确定输出A行为, 使用以文字表达式、决策表、调用或盒装上下文表示的决策逻辑。
业务知识模型	业务知识模型是指A函数为代表的可复用的决策逻辑模块, 包括零个、一个或多个参数。
服务决策服务 (扩展)	A可以决策可重用的决策, 这些决策在内部调用 - 例如, 由另一个决策决策或决策业务知识模型- 或在外部 - 例如, 由进程流程调用。 A好的做法是使用图表来描述单个扩展的决策服务。
服务决策服务 (折叠)	如果一个决策服务元素是一个可调用的元素, 通过调用逻辑将知识需求与其他元素连接起来, 我们可以隐藏决策服务的细节, 以聚焦在大图的决策层次上。
输入数据	输入数据元素表示用作一个或多个决策的输入的信息。
项目定义	项定义用于定义决策模型中使用的数据项的类型和结构。它主要由 Input Data 元素引用, 作为预期输入的数据类型和结构的基础。也可以参考它来设置输出的结构。 项目定义包含数据集, 这些数据集提供了在执行各种模拟时有用的值集。
知识源	知识源元素表示业务知识模型或决策A决策。
信息需求	信息需求表示输入数据或决策输出被用作决策的决策。
知识需求	知识需求表示对业务知识模型或决策服务A调用。
权威需求	权威需求表示一个 DRG元素对另一个 DRG元素的依赖性, 该 DRG 元素源指导或知识的来源。

决策表达式编辑器

DMN 表达式编辑器是您将在其中定义、审阅和更新模型中大多数不同类型 DMN 元素的详细信息的窗口。首先，它用于编辑决策元素和业务知识模型 (BKM) 元素的价值表达。

A 决策元素和决策元素使用的四种类型的值表达中的每一种，都会显示不同版本的 DMN 表达式编辑器。对于 BKM 元素，还提供了第二个窗口选项卡，用于定义调用 BKM 时使用的输入和输出参数。

还存在两个附加版本的 DMN 表达式编辑器以支持编辑 ItemDefinition 和 InputData 元素。

显示的工具栏和窗口内容的布局取决于当前选择的 DMN 元素的类型，以及在适用的情况下定义的值表达式的类型。

此图显示了用于定义决策表的 DMN 表达式编辑器的版本。在这种情况下，底层元素是一个 BusinessKnowledgeModel，因此决策逻辑由其他元素“调用”，输入和输出通过参数传递。

Cabin Upgrade Policy		Input Parameter Values for Simulation	
(Input 1, Input 2)			
U	Input 1	Input 2	Output 1
1	-	-	-
2	-	-	-
3	-	-	-

DMN 表达式编辑器对每个元素和表达式类型的特征的详细解释在本主题的子帮助主题中提供。

访问

图表	<p>双击图表上的 DMN 元素。</p> <p>显示元素及其表达式类型对应的 DMN 表达式编辑器窗口。</p>
----	---

值表达式

本表总结了四种不同类型的价值表达，并参考了详细说明每种类型的帮助主题。

类型和图标	描述
决策表	决策表是 A 组相关输入和输出表达式的表格表示，组织成规则，指示哪个输出条目适用于一组特定的输入条目。
文字表达	文字表达式 A 决策逻辑指定为描述输出值如何从其输入值派生的文本表达式。为了支持模拟和执行，文字表达式可以使用 JavaScript 函数。
盒装上下文	盒装上下文是上下文条目 A 集合，由 (名称, 值) 对组成，每对都有一个结果值。 上下文条目提供了一种将复杂的表达式分解为一系列简单表达式的方法，提供可以在后续上下文条目中使用的中间结果。
调用	调用调用另一个模型元素 (业务知识模型或决策服务) 以提供决策结果。调用定义了传递给“被调用”元素的参数，为上下文其决策逻辑提供时间。然后

	将决策结果传递回“调用”元素。
--	-----------------

ItemDefinition 和 InputData 元素

元素	描述
项目定义	<p>ItemDefinition 元素用于定义数据结构，并可选择限制数据允许值的范围。</p> <p>ItemDefinitions 的范围可以从简单的单一类型到复杂的结构化类型。</p> <p>ItemDefinitions 用于指定 InputData 元素的类型以及输入参数。</p>
输入数据	<p>决策元素用于为决策元素提供输入。</p> <p>InputData 元素的数据类型是使用 ItemDefinition 元素定义的。数据集也可以定义为 ItemDefinition 的一部分，然后 InputData 元素可以指定运行模拟时要使用的数据集。</p>

关系

DMN决策关系元素提供了A方便的简写方法，用于在 DMN 图中定义相关值的列表。决策关系就像A有列和行的关系表。网格的标题显示每列的名称。每行显示对应列的值集。

访问

工具箱	<p>创建决策关系：</p> <ul style="list-style-type: none"> • 确保蓝图设置为：需求>决策建模 • 从工具箱的图表Components 页面，将决策元素或 BKM元素拖到决策图上 • 从弹出的表达式菜单中选择“关系” • 双击 DMN元素以显示DMN 表达式窗口。
-----	---

或者，您可以将现有的决策决策或 BKM元素更改为决策关系类型。去做这个：

- 右键单击图表上的 DMN决策或 BKM元素，然后选择“属性|属性”菜单选项
- 在“常规”页面上，选择“标签”选项卡，在“表达式类型”值字段中单击下拉箭头并选择“关系”；点击确定按钮

概述

DMN关系决策类型是包含值的垂直列表。使用决策关系的A关键是使用`For` 循环遍历值行的方法。例如，可以在相关的文字表达式元素决策中将决策循环定义为处理决策关系中的行的公式。

编辑 DMN 关系

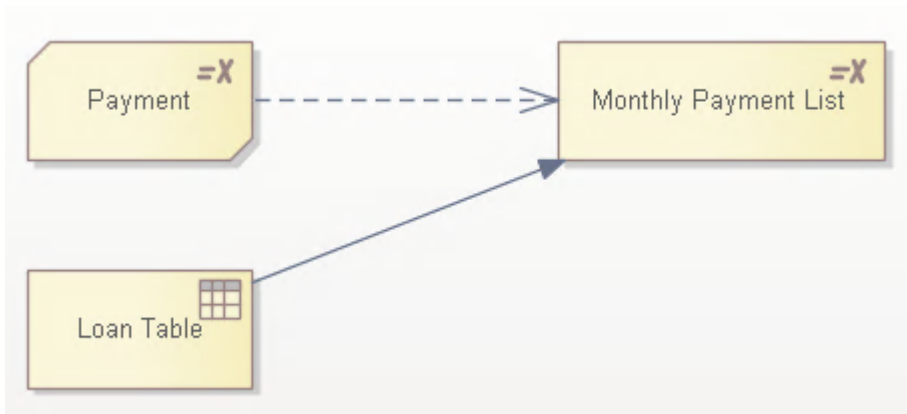
DMN 表达式窗口有一个工具栏，提供添加新行、删除现有行以及向上或向下移动选定行的选项。

你也可以：

- 拖动网格标题以重新定位列。
- 右键单击单元以显示上下文菜单选项，用于将列的类型设置为：'string'、'number'、'boolean'或'object'。

示例-贷款库表

在这个例子中，我们有两个决策——“贷款库表”和“每月付款清单”——以及一个业务知识模型——“付款”。



贷款库表”是一个以关系实施的决策，有四列：贷款”、原则”、团队”和 年利率”。

DMN Expression

Loan Table

	Loan : string	Principle : number	Term : number	Annual Rate : number
	Loan 1	100000	360	0.02
	Loan 2	100000	360	0.05
	Loan 3	100000	360	0.10

每月付款清单”是一种决策，以带有for循环的文字表达式实现：

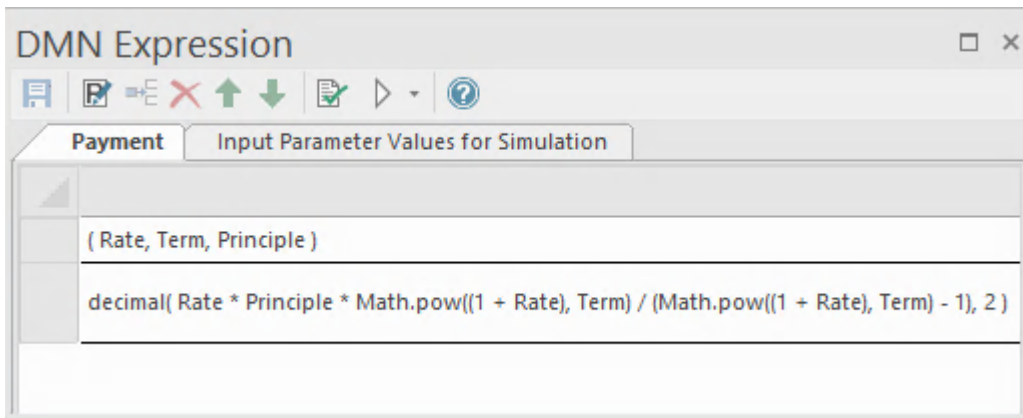
DMN Expression

Monthly Payment List

```
for x in Loan Table return [x.Loan, Payment(x["Annual Rate"] / 12, x.Term, x.Principle)]
```

for循环将遍历“贷款库表”：

- 每个项目 'x' 是表的一行，表示为一个列表
- 对每一行项目 'x'，调用业务知识模型“支付”与行列表中的项目
- 可以通过两种不同的方式访问列表中的每个项目：
 - (1) 直接访问Relation的列，如x.Loan,团队, x.Principle
 - (2) 列名有空格的地方，使用string访问：x["Annual Rate"]



仿真时，运行时值显示在仿真窗口和元素旁边的图表中；您可以单击一个步骤来查看模拟过程。

The screenshot displays the DMN Simulation interface. At the top, a decision diagram shows a "Loan Table" decision relation feeding into a "Monthly Payment List" decision relation. The "Loan Table" is defined with three rows of data:

```

[
  {
    "Annual Rate": 0.02,
    "Loan": "Loan 1",
    "Principle": 100000,
    "Term": 360
  },
  {
    "Annual Rate": 0.050000000000000003,
    "Loan": "Loan 2",
    "Principle": 100000,
    "Term": 360
  },
  {
    "Annual Rate": 0.100000000000000001,
    "Loan": "Loan 3",
    "Principle": 100000,
    "Term": 360
  }
]
    
```

The "Monthly Payment List" decision relation is defined with a loop:

```

[
  {
    "Loan 1",
    {
      "value": "369.62"
    }
  },
  {
    "Loan 2",
    {
      "value": "536.82"
    }
  },
  {
    "Loan 3",
    {
      "value": "877.57"
    }
  }
]
    
```

A callout box explains the loop: "Monthly Payment List contains: for x in Loan Table return [x.Loan, Payment(x['Annual Rate'] / 12, x.Term, x.Principle)]". It notes that the loop iterates through the Loan Table and that each item 'x' is a row of the table. It also provides two ways to access the data: (1) Directly Access the Relation's Column, like x.Loan, x.Term, x.Principle; (2) Where the Column name has a space, use string access: x['Annual Rate'].

Below the diagram is the "DMN Simulation" window, which shows a table of simulation results:

Element	Value
Loan Table	[{"Annual Rate": 0.02, "Loan": "Loan 1", "Principle": 100000, "Term": 360}, {"Annual Rate": 0.050000000000000003, "Loan": "Loan 2", "Principle": 100000, "Term": 360 ...
Payment	"369.62"
Payment	"536.82"
Payment	"877.57"
Monthly Payment List	[["Loan 1", {"value": "369.62"}], ["Loan 2", {"value": "536.82"}], ["Loan 3", {"value": "877.57"}]]

盒装清单

DMN 盒装列表是A决策元素，其中包含盒装表达式列表。这些项目在DMN 表达式窗口中以垂直列表的形式排列。

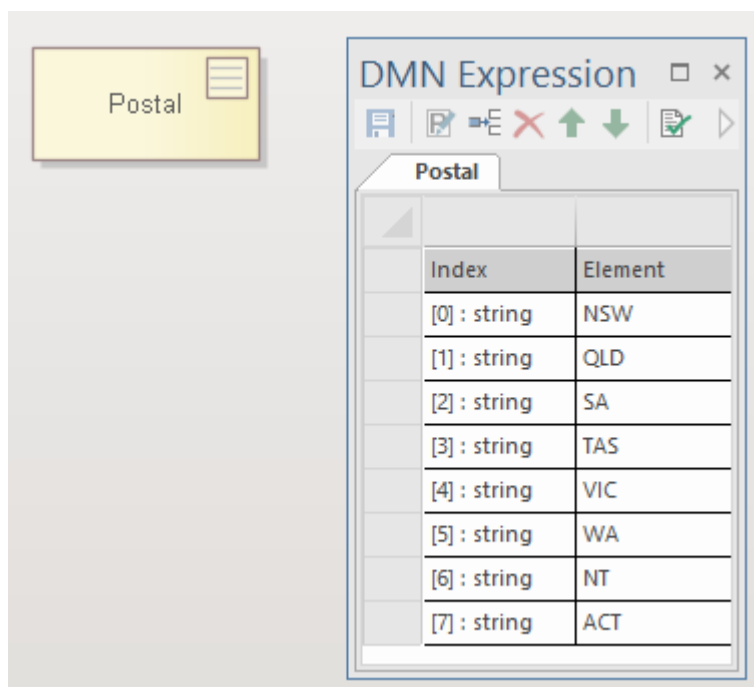
盒装A通常与包含在相关决策元素中的*for* 循环表达式结合使用。*for* 循环表达式用于遍历盒装列表中的每一行，将列表的元素字段绑定到相应的变量，并在范围内评估表达式。*for* 循环的输出是一个列表，其中包含每个单独迭代的表达式评估。

访问

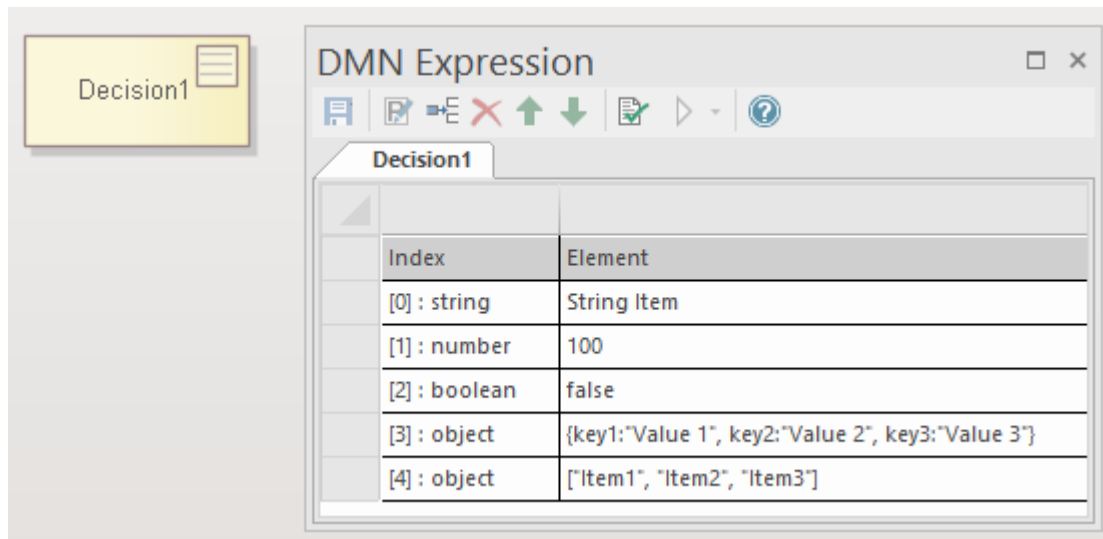
图表工具箱	将工具箱中的工具箱元素或 BKM元素拖到决策图上，然后从弹出的表达式菜单中选择“列表”。 双击DMN元素；显示DMN 表达式窗口，显示所选元素的详细信息。
属性	右键单击图表上的 DMN决策或 BKM元素，然后选择“属性 属性”菜单选项。 在“常规”页面上，选择“标签”选项卡，然后在“表达式类型”值字段中单击下拉箭头并选择“列表”。点击确定按钮。 双击DMN元素；显示DMN 表达式窗口，显示所选元素的详细信息。

概述

盒装列表通常用作枚举，其中所有列表项都属于同一类型。



将盒装列表用作数据集合也很常见，其中每个项目可能具有不同的类型。

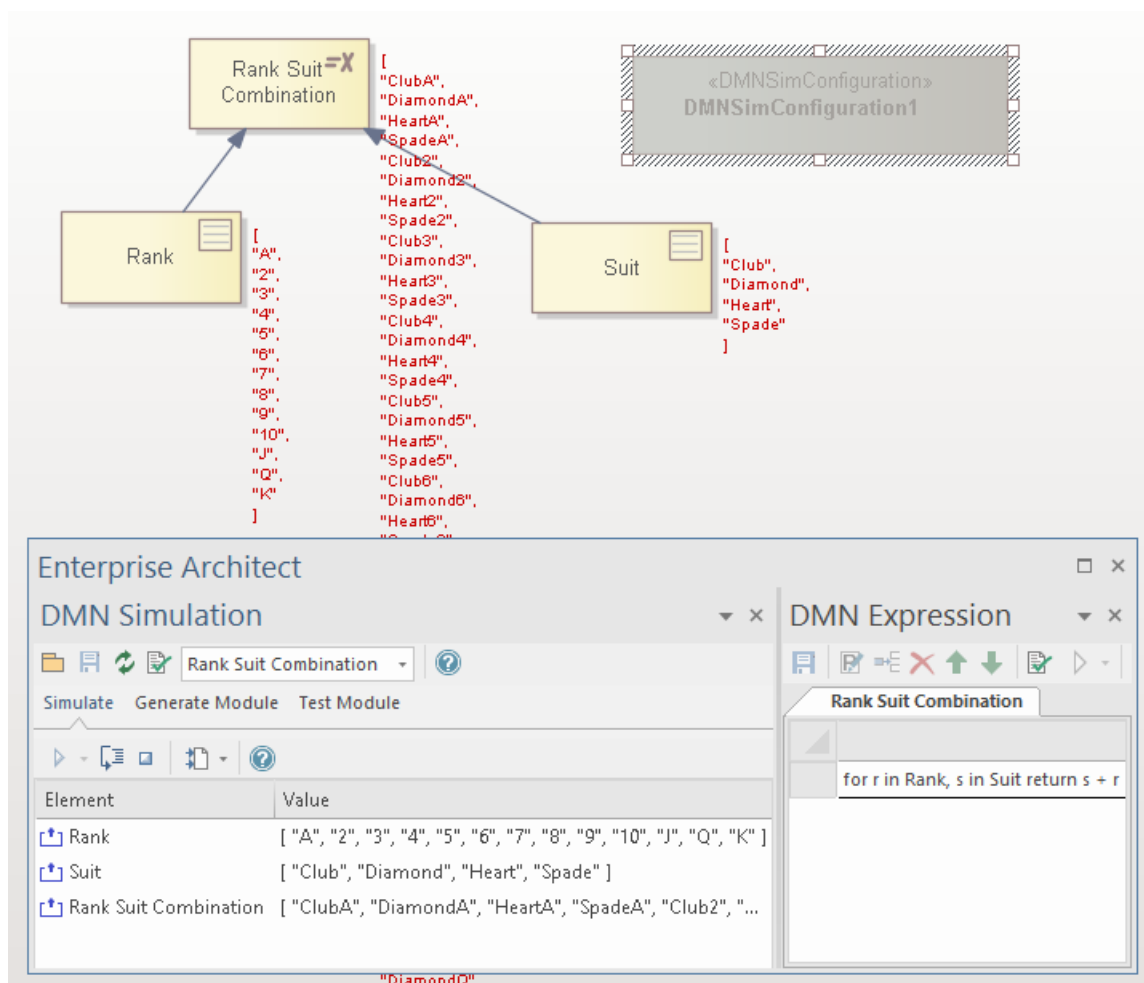


编辑盒装列表

DMN 表达式窗口有一个工具栏，提供“加新列表项”、“删除现有列表项”和“向上或向下移动项”选项。右键单击列表项以显示上下文菜单选项，用于设置列表项的类型 - string、数字、布尔值或object。

示例- 扑克等级和花色

在这个例子中，我们三个决策：Rank、Suit 和 Rank Suit Combination



- 决策表示为一个项列表，包含从“A”到“K”的13个项目
- 决策表示为带有4项的盒装列表：“俱乐部”、“钻石”、“心”和“黑桃”
- 决策Suit Combination表示为带有for循环的文字表达式：*for r in Rank, s in Suit return s + r*

当在同一个for循环表达式中定义多个迭代上下文时，生成的迭代是迭代上下文元素的叉积。迭代顺序是从内迭代上下文到外迭代上下文。

在此示例中，Rank（13项）和Suit（4项）的叉积是 $13 * 4 = 52$ 个元素的列表。

编辑DMN 表达式对话框

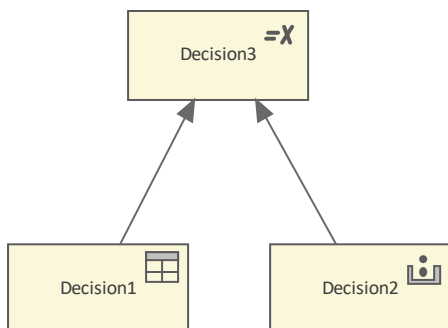
元素编辑DMN 表达式”对话框用于设置盒装内容、调用和文字表达式类型中的表达式。它提供智能感知支持，用于构建基于 FEEL 语法的表达式以及可用于模型生成的代码语言。

DMN 表达式表达编辑和智能感知支持

为了帮助您减少输入和错误，Enterprise Architect提供了智能感知表达式编辑支持来编辑表达式。注册根据 FEEL 语言规范，参数和上下文条目变量名称可以包含空格。此特征旨在使每个表达式易于阅读。

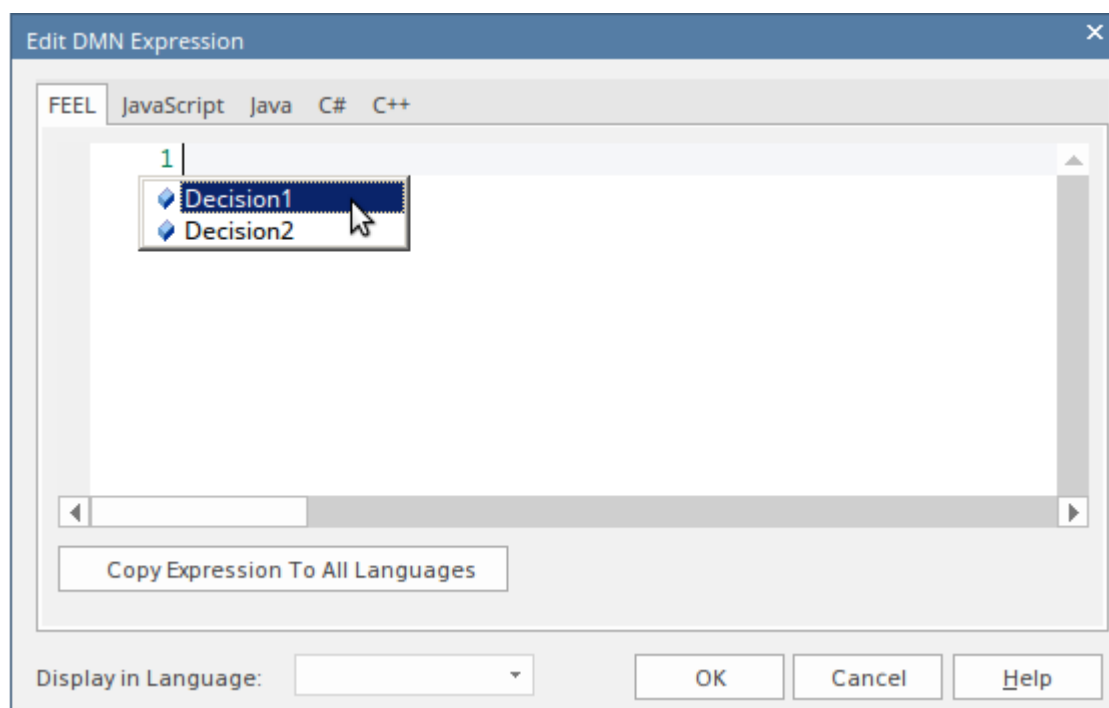
例子

鉴于此决策层次结构，“Decision3”中的表达 能够使用来自两个引用决策的输出。



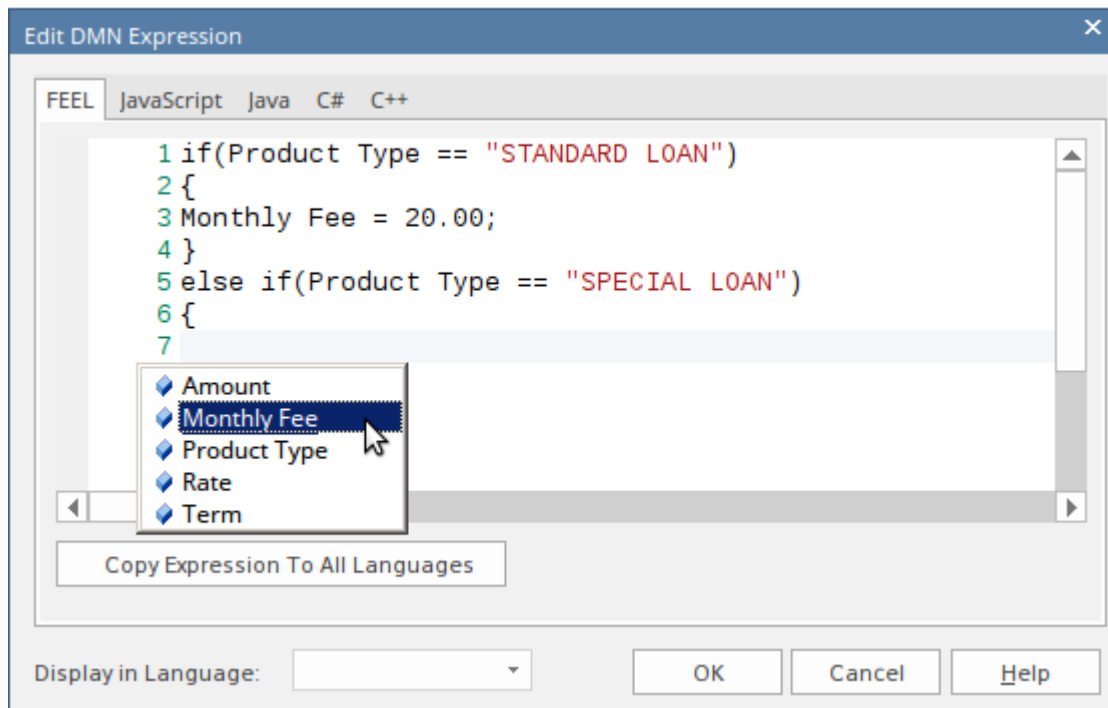
要打开 编辑DMN 表达式”对话框：

1. 双击图中的决策元素，显示DMN 表达式窗口。
2. 右键单击表达式行并选择菜单选项 编辑表达式”。将显示 编辑DMN 表达式”对话框。
3. 单击 a 并按 Ctrl+空格键以显示智能感知菜单：



- 对于 BusinessKnowledgeModel 表达式，将包含所有参数
- 对于决策表达，将包括所有必需的决策
- 将包括早于当前变量的所有上下文条目变量（不包括晚于当前变量的上下文条目变量）

在此示例中，编辑 BKM 盒装上下文表达式，输入参数显示在智能感知菜单中：




语言选择

DMN模型可以生成JavaScript、Java、C#或C++的源代码。由于语言之间的语法不同，Enterprise Architect为每种语言提供了语言覆盖页面。如果没有为语言指定覆盖代码，则将使用为FEEL语言定义的表达式。

注记：在生成的代码中，变量名中的空格会被替换为下划线。

文字表达

Literal Expression 是 DMN 表达式A最简单形式；它通常定义为单行语句或 if-else 条件块。Literal Expression 是一种在决策元素和业务知识模型(决策)元素中使用的价值表达。随着表达式变得越来越复杂，您可能更喜欢 Boxed Context，或者为了提高可读性，您可以将一些逻辑封装为 DMN 库中的函数。

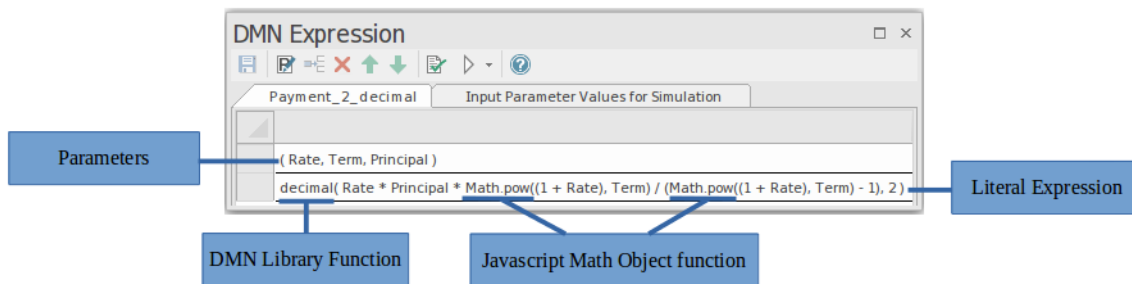
决策或 BKM元素右上角的  图标表示它是作为 *Literal Expression* 实现的。

访问

图表	在图表上，双击决策元素或 BusinessKnowledgeModel元素。 DMN 表达式编辑器窗口显示所选元素的详细信息。
----	--

概述

此图像显示DMN 表达式编辑器窗口，它显示为文字表达式。



文字表达式是决策逻辑的文本表示。它描述了如何使用数学和逻辑运算从输入值导出输出值。

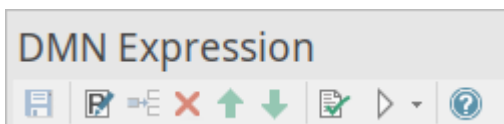
表达式编辑器窗口将 Literal Expression 显示为一个表，其中包含两个关键行：

- 参数：定义表达式中使用的输入参数
- 文字表达式：定义表达式的公式的地方 - 这定义了决策的输出

为了支持模拟和执行，文字表达式可以使用JavaScript全局函数或JavaScript object函数。用户还可以创建 DMN 库函数以在表达式中使用。

文字表达式编辑器的工具栏

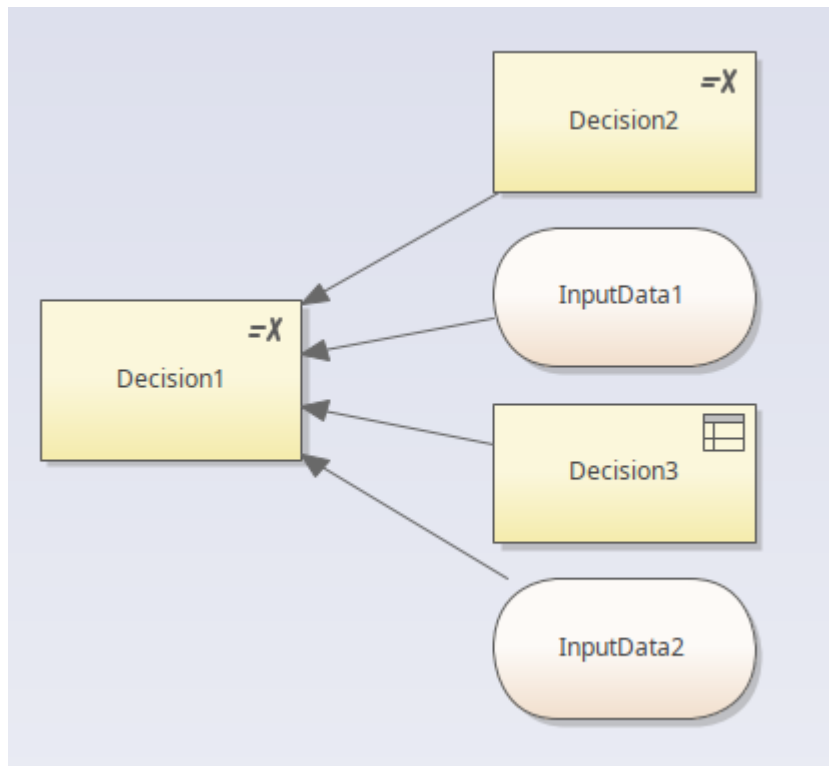
选择文字表达式后，DMN 表达式窗口中可访问的布局特征为：



更多详情请参考帮助主题 [Toolbar for帮助Expression Editor](#)。

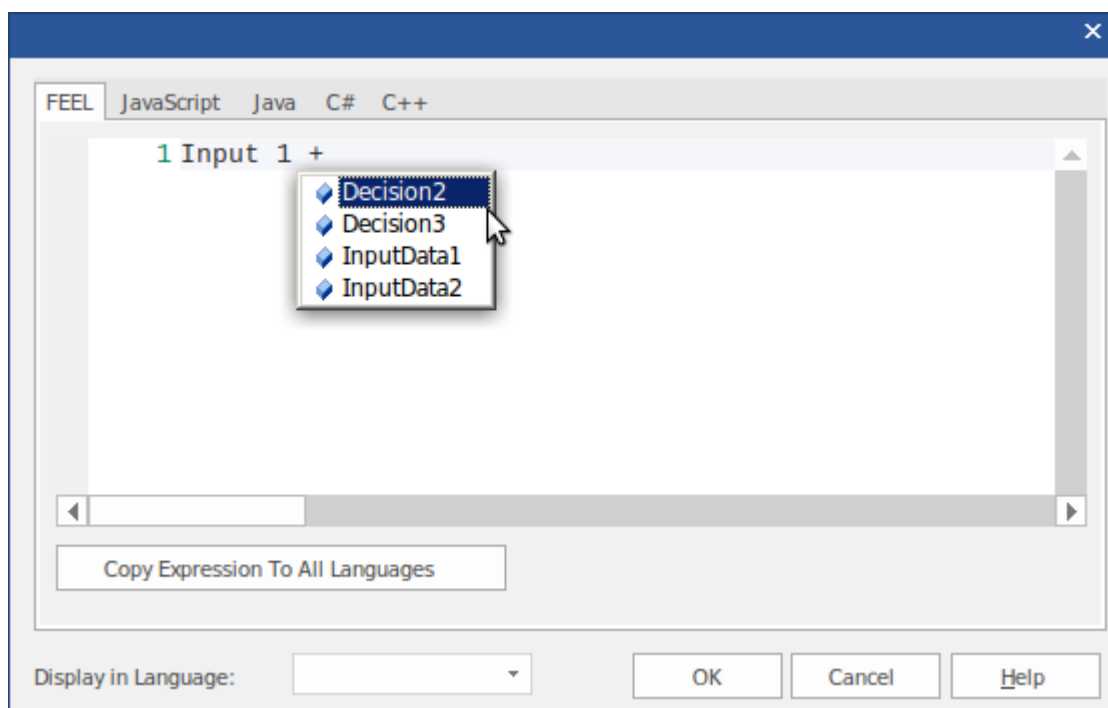
表情编辑器和智能感知支持

根据《足够友好表达语言》(FEEL)语言规范，参数名称可以包含空格，这样表达起来更容易阅读。Enterprise Architect还提供智能感知。支持编辑表达式，减少输入和错误。



给定一个如图所示的决策层次结构，当编辑“Decision1”的表达式时，“Decision1”的输入——即“Decision2”、“Decision3”、“InputData1”和“InputData2”——将通过智能感知获得编辑。

通过右键单击DMN表达式窗口的“表达式”行，然后选择菜单项“编辑表达式...”，将显示表达式代码编辑器对话框。按 Ctrl+Space 显示智能感知菜单：



- 对于“决策”元素，将显示决策的所有输入
- 对于业务知识模型 (BKM) 元素，将显示所有输入参数

DMN模型可以生成JavaScript、Java、C#或C++的源代码；由于某些语言可能对某些表达式有不同的语法，Enterprise Architect为每种语言提供了语言覆盖页面。如果没有为语言指定覆盖代码，则将使用为FEEL语言定义的表达式。

在生成的代码中，变量名中的空格将被替换为下划线。

文字表达式编辑器的工具栏

选择文字表达式后，DMN 表达式窗口会显示特定于该表达式类型的工具栏。

工具栏选项

此表提供了在选择文字表达式时可从DMN 表达式窗口中的工具栏访问的特征描述。

选项	描述
	单击此按钮可将配置保存到当前决策或 BusinessKnowledgeModel。
	单击此按钮可编辑业务知识模型的参数。
	对文字表达式禁用此选项。
	对文字表达式禁用此选项。
	对文字表达式禁用此选项。
	对文字表达式禁用此选项。
	单击此按钮以执行文字表达式的验证。Enterprise Architect将执行一系列验证以帮助您定位表达式中的任何错误。
	当为 BusinessKnowledgeModel元素定义文字表达式时，将启用此按钮。

示例-偿还贷款

该业务知识模型 (BKM) 支付实现为文字表达。

DMN Expression

Payment Input Parameter Values for Simulation

(Rate, Term, Principle)

decimal(Rate * Principle * Math.pow((1 + Rate), Term) / (Math.pow((1 + Rate), Term) - 1), 2)

- BKM 定义了三个参数：Rate、团队和 Principle

设置输入参数的值并评估模型：

Payment Input Parameter Values for Simulation

Payment	
Rate : number	0.06 / 12
Term : number	30 * 12
Principle : number	300000

- 将显示运行时参数值；例如，速率 = 00.005
- BKM 的结果将通过字面表达式求值，并在声明行显示值；例如，返回 = 1798.65

Payment Input Parameter Values for Simulation

(Rate = 0.005, Term = 360, Principle = 300000)return = "1798.65"

decimal(Rate * Principle * Math.pow((1 + Rate), Term) / (Math.pow((1 + Rate), Term) - 1), 2) (evaluation result = "1798.65")

虽然这个公式可以写成一行，但是相当复杂。我们可以使用内置函数和盒装上下文重构此模型以提高可读性：

Payment Input Parameter Values for Simulation

(Rate = 0.005, Term = 360, Principle = 300000)return = "1798.65"

pmt = 1798.6515754582765	PMT(Rate, Term, Principle)
pmt 2 decimals = "1798.65"	decimal(pmt,2)
pmt 2 decimals (evaluation result = "1798.65")	

- Boxed Context 定义了两个变量表达式配对条目；这些变量用作“局部变量”，可以在以后的表达式中使用
- 返回值：表达式可以使用‘局部变量’的值
- Boxed Context 中的任何表达式都可以使用在可定制模板 - DMN 库中定义的内置函数；例如，此示例中使用了函数 PMT(...) 和 decimal(...)

模拟结果与 Literal Expression 完全相同：

Payment		Input Parameter Values for Simulation	
	(Rate = 0.005, Term = 360, Principle = 300000)return = "1798.65"		
	pmt = 1798.6515754582765		PMT(Rate, Term, Principle)
	pmt 2 decimals = "1798.65"		decimal(pmt,2)
	pmt 2 decimals (evaluation result = "1798.65")		

调用

调用是为评估上下文业务知识模型主体提供时间的参数绑定的容器。调用有两种常见用例：

- 将输入数据绑定到业务知识模型
- 将参数或上下文入口变量绑定到业务知识模型

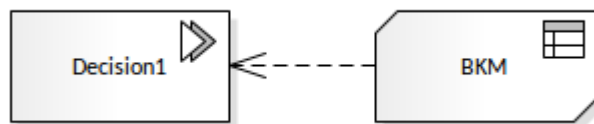
本帮助主题的子主题中提供了每个示例。

访问

图表	双击适当的决策元素或 BKM 元素。 显示 DMN 表达式窗口，显示所选元素的详细信息。
----	---

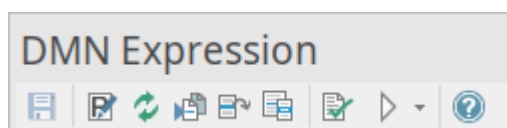
概述

调用是一种适用于决策要素和决策业务知识模型要素的价值表达类型。它是一个表格表示，一个决策或另一个业务知识模型调用一个可调元素（一个决策业务知识模型或一个决策服务）中定义的决策逻辑。



调用编辑器工具栏

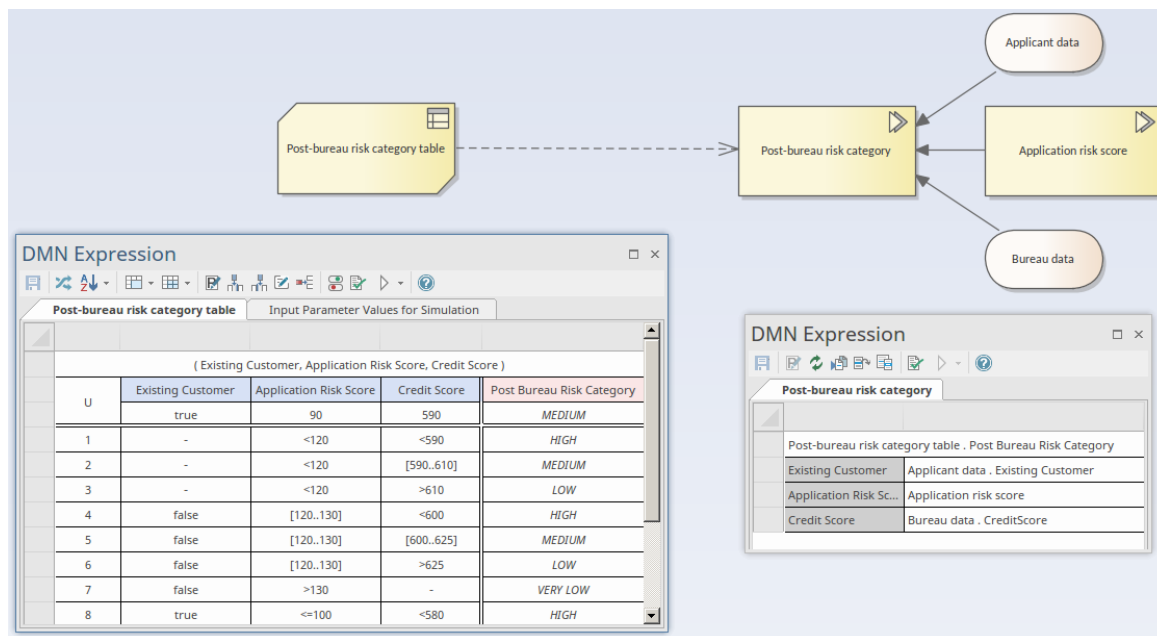
选择一个调用后，可以从功能 DMN 表达式窗口的工具栏访问许多处理它的功能：



更多详细信息请参考帮助主题“调用编辑器工具栏”。

绑定

调用的参数上下文提供了评估可调元素主体的时间。

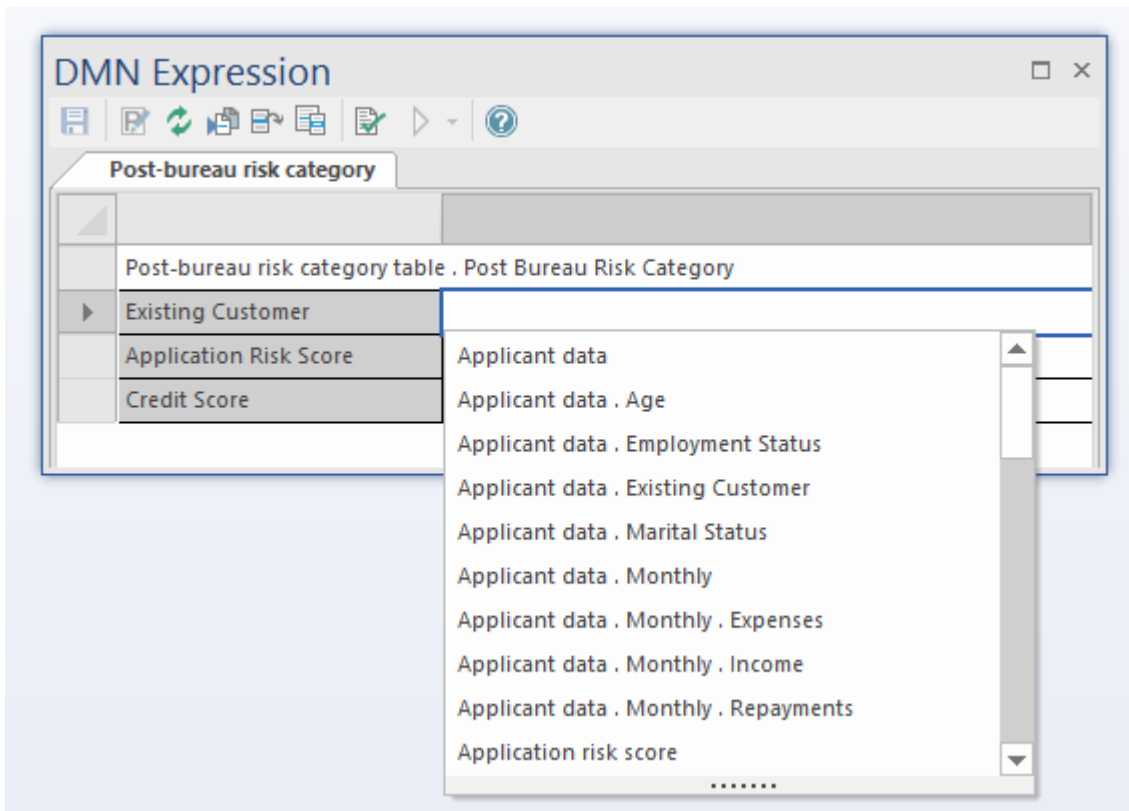


在这个例子中：

- 将'Post-bureau risk category'表示为与决策业务知识模型“局后风险类别表”相关联的调用，以决策决策表的形式实现
- 决策决策'Post-bureau risk category'是来自两个输入数据元素和一个决策要素的三个信息需求连接器的目标
- 绑定列表将输入值绑定到业务知识模型的参数
- 调用还指定了请求的“OutputClause”；在决策表定义了多个输出子句的情况下，调用必须明确请求输出子句作为表达式的结果

输入

可以通过在字段中按空格键来设置来自其他决策和 InputData 元素的输入：



输出

由于Invocation只能调用一个业务知识模型，输出由业务知识模型输出定义。

调用编辑器工具栏

选择调用表达式时，DMN 表达式窗口工具栏提供特定于该表达式类型的选项。

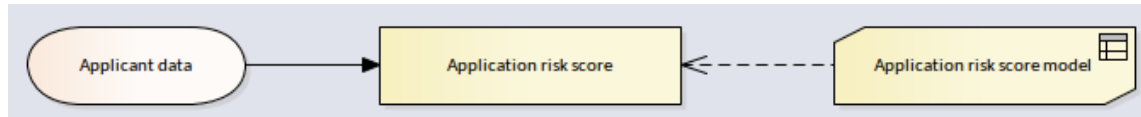
工具栏选项

此表提供了在选择调用时在DMN 表达式窗口中可访问的特征的描述。

选项	描述
	单击此按钮可将配置保存到当前决策或 BusinessKnowledgeModel。
	单击此按钮可编辑业务知识模型的参数。
	适用于调用值表达式，适用于决策元素和业务知识模型(BKM) 元素。 单击此按钮可与调用的 BKM 同步。例如，如果 BKM 更改名称、参数、输出或类型，请单击此按钮以同步这些更改。
	适用于调用值表达式，适用于决策元素和业务知识模型(BKM) 元素。 单击此按钮可设置或更改 BKM 作为调用。
	适用于调用值表达式，适用于决策元素和业务知识模型(BKM) 元素。 单击此按钮可在DMN 表达式窗口中打开调用的 BKM。
	适用于调用值表达式，适用于决策元素和业务知识模型(BKM) 元素。 当一个BKM被实现为决策表时，它可以定义多个输出子句；对此 BKM 的调用可能必须指定请求的输出。 单击此按钮可在上下文菜单中列出所有可用的输出；检查当前配置的输出。
	执行调用的验证。Enterprise Architect将执行一系列验证以帮助定位调用定义中的任何错误。
	此按钮在为业务知识模型定义调用时启用。 选择“用于仿真的输入参数值”选项卡，完成字段并单击此按钮。测试结果将显示在决策表上，显示输入和输出的运行时值，并突出显示有效规则。 您可以使用此功能对业务知识模型进行单元测试，而无需知道上下文和稍后由决策或其他业务知识模型调用。 此工具栏按钮可使用菜单选项。有关详细信息，请参阅仿真DMN模型帮助主题。

示例1 - 将输入数据绑定到业务知识模型

可以在A模型模式创建完整示例（在功能区中，选择 仿真>决策分析>决策>应用蓝图>决策决策>决策与决策：创建模型”）。



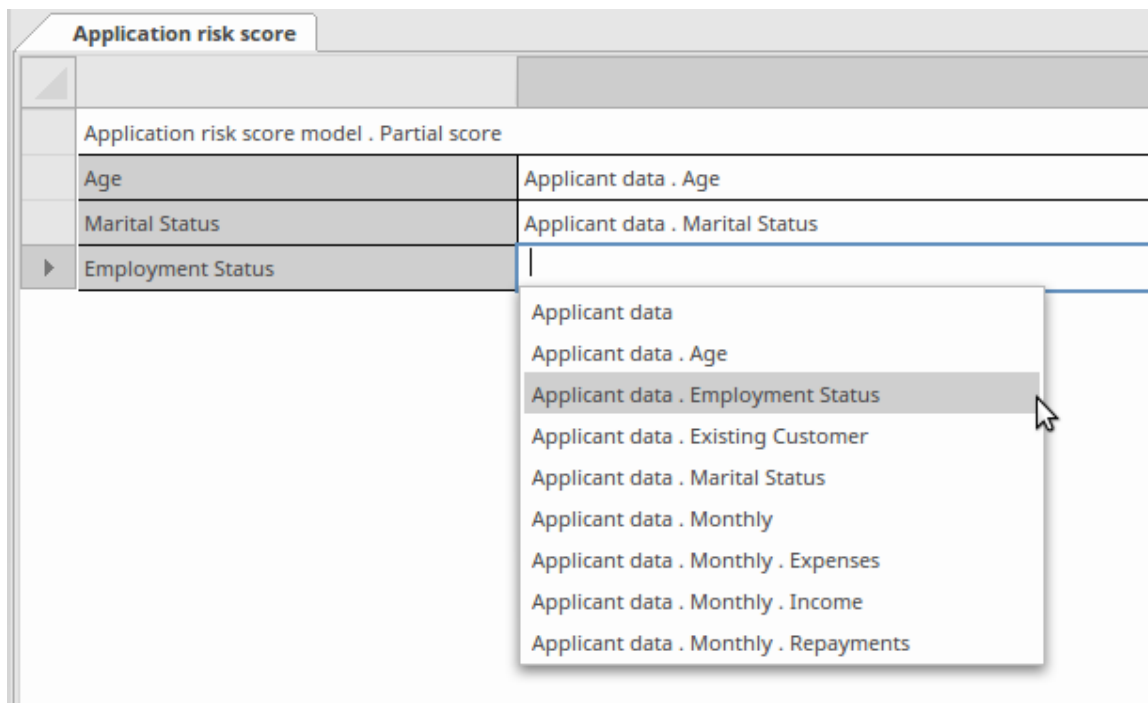
在此示例中，输入数据申请人数据被键入到申请人数据定义中，该定义具有三个组件。

Applicant data : Applicant data Definition		
Applicant data Definition	Age : number	40
	Employment Status : string	"EMPLOYED"
	Marital Status : string	"M"

业务知识模型应用风险评分模型实现为三进一出的决策决策表。

Application risk score model		Input Parameter Values for Simulation		
(Age, Marital Status, Employment Status)				
C+	Age	Marital Status	Employment Status	Partial score
	[18..120]	S,M	UNEMPLOYED,STUDENT,EMPLOYE...	
1	[18..21]	-	-	32
2	[22..25]	-	-	35
3	[26..35]	-	-	40
4	[36..49]	-	-	43
5	>=50	-	-	48
6	-	S	-	25
7	-	M	-	45
8	-	-	UNEMPLOYED	15
9	-	-	STUDENT	18
10	-	-	EMPLOYED	45
11	-	-	SELF-EMPLOYED	36

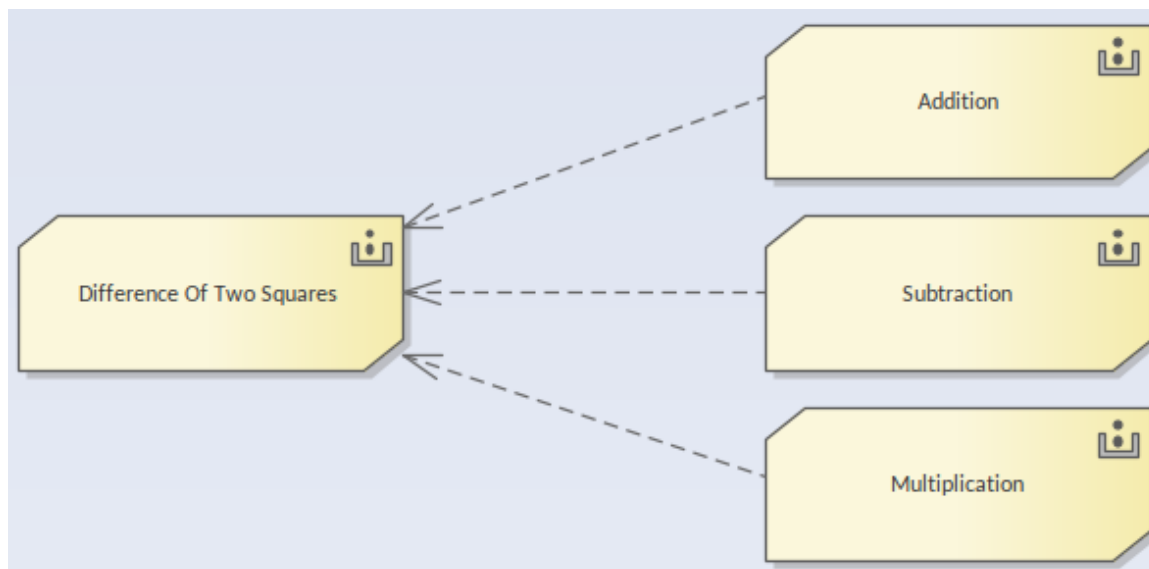
决策应用程序风险评分被实现为将输入数据的“叶”组件绑定到 BKM 参数的调用。



为了使绑定更容易，绑定表达式支持自动完成。
该模式的文档中提供了完整的建模和仿真说明。

示例2 - 将上下文条目变量绑定到业务知识模型

可以在A模型模式创建完整示例（在功能区中，选择 仿真>决策分析>决策>应用蓝图>DMN业务知识模型示例>业务知识模型调用：创建模型）。



在本例中，业务知识模型（BKM）两平方差值实现为盒装上下文：

- ab 的变量 sum 通过将参数 a 和 b 绑定到 BKM *Addition* 来实现为调用
- ab 的变量差异通过绑定参数 a 和 b 到 BKM 减法实现为调用
- 通过将局部变量 ab 的 sum 和 ab 的差值绑定到 BKM 乘法来实现变量的平方差

Difference Of Two Squares		Input Parameter Values for Simulation	
(a, b)			
sum of ab	Addition		
	addend 1	a	
	addend 2	b	
difference of ab	Subtraction		
	minuend	a	
	subtrahend	b	
difference of squares	Multiplication		
	factor 1	sum of ab	
	factor 2	difference of ab	
difference of squares			

为了使绑定更容易，绑定表达式支持自动完成。
该模式的文档中提供了完整的建模和仿真说明。

DMN 表达式自动完成

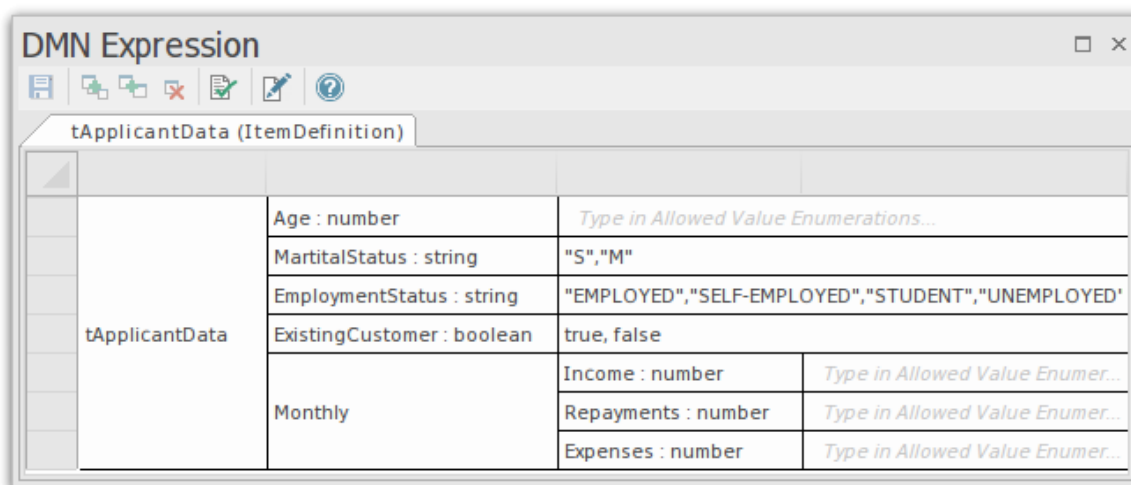
DMN 定义了很多表达式，例如 FunctionDefinition、DecisionTable、Boxed Context、Invocation 和 Literal Expression。这些表达式的参数、参数和逻辑主要由文本实现。

为了使建模简单可靠，Enterprise Architect提供了自动完成功能，帮助提供：

- ItemDefinition 的允许值
- 决策表的输入/输出条目
- 信息需求

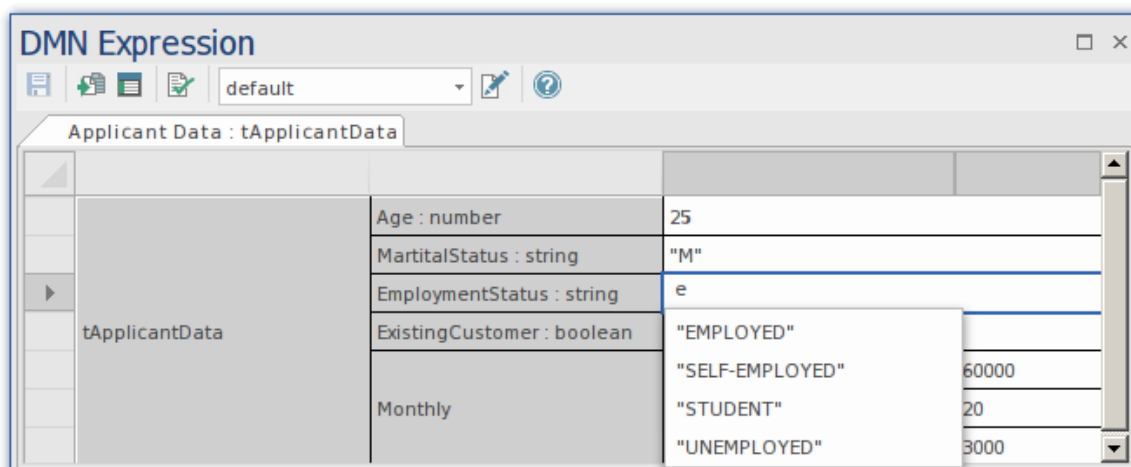
ItemDefinition 的允许值

这个想法是在 ItemDefinition 中定义允许的值枚举，然后在请求这些值时组成一个列表以供选择。在本例中，ItemDefinition '申请人数据'。就业状态'定义了允许值的枚举。



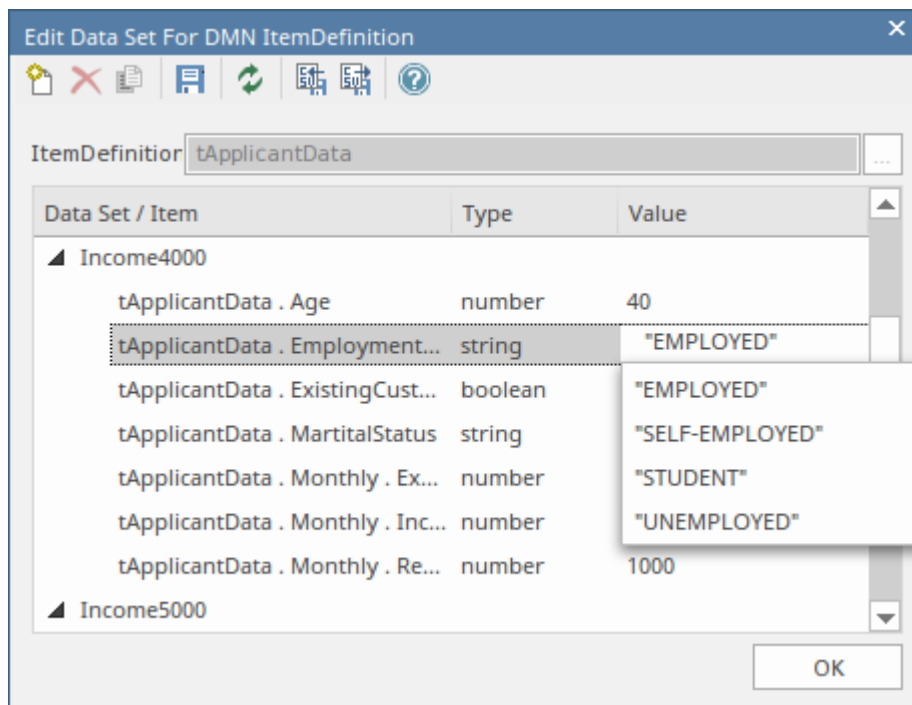
DMN Expression			
tApplicantData (ItemDefinition)			
tApplicantData	Age : number	Type in Allowed Value Enumerations...	
	MartitalStatus : string	"S","M"	
	EmploymentStatus : string	"EMPLOYED","SELF-EMPLOYED","STUDENT","UNEMPLOYED"	
	ExistingCustomer : boolean	true, false	
	Monthly	Income : number	Type in Allowed Value Enumer...
Repayments : number		Type in Allowed Value Enumer...	
Expenses : number		Type in Allowed Value Enumer...	

编辑为此 ItemDefinition 键入的 InputData 的值时，按键盘上的空格键以显示可供选择的值列表。



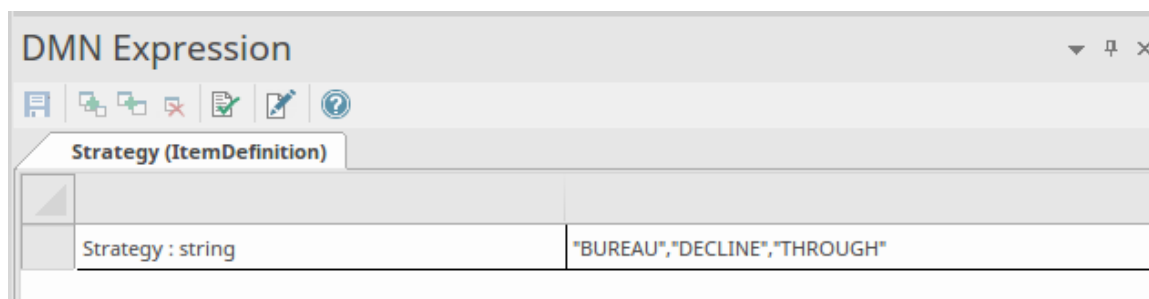
DMN Expression			
Applicant Data : tApplicantData			
tApplicantData	Age : number	25	
	MartitalStatus : string	"M"	
	EmploymentStatus : string	e	
	ExistingCustomer : boolean	"EMPLOYED"	
	Monthly	Income : number	60000
Repayments : number		20	
Expenses : number		3000	

我们还可以为 InputData 定义多个数据集，因为自动完成特征在此对话框中可用。

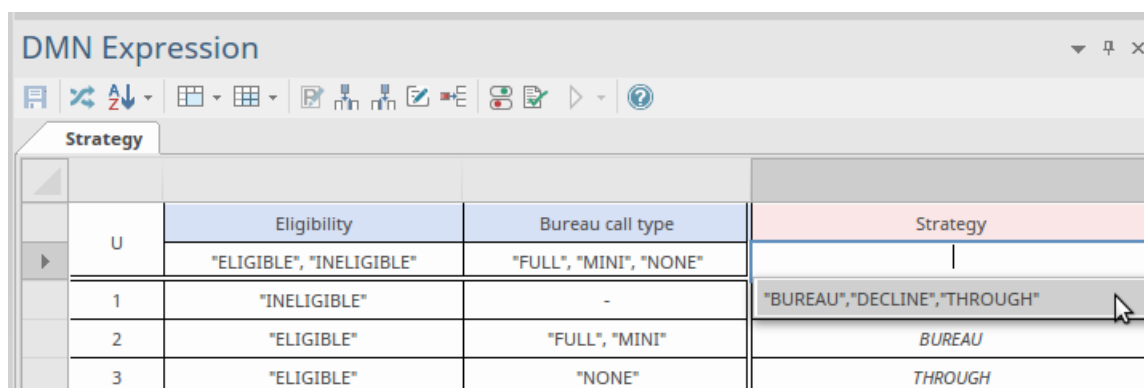


决策表的输入/输出条目

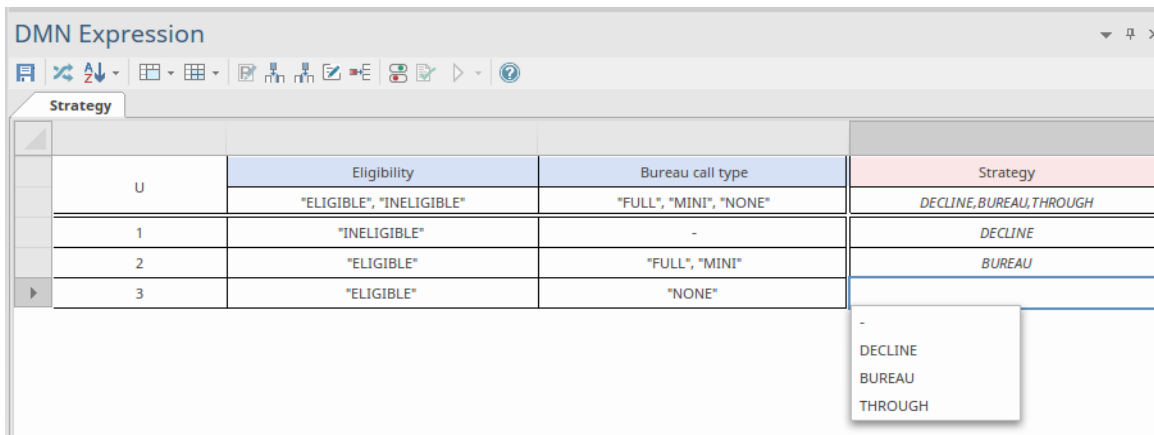
以 'Strategy' ItemDefinition 为例：



我们可以通过选择快速填写决策表的 允许值"字段：



然后我们可以通过选择快速填写决策表规则：



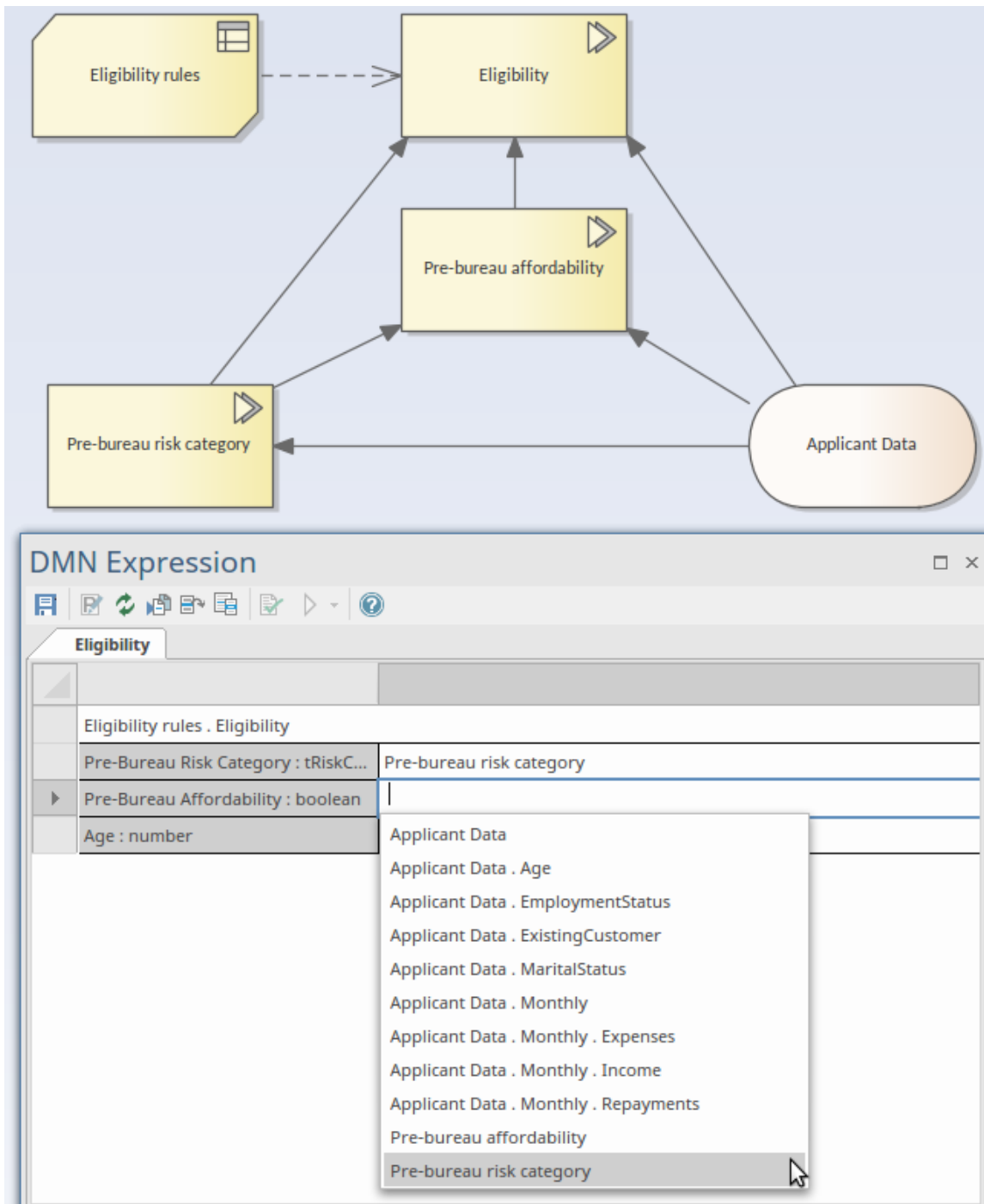
The screenshot shows the 'DMN Expression' window with a table. The table has three columns: 'Eligibility', 'Bureau call type', and 'Strategy'. The first row is a header row with 'U' in the first column, 'Eligibility' in the second, 'Bureau call type' in the third, and 'Strategy' in the fourth. The second row contains '1', '"INELIGIBLE"', '-', and 'DECLINE'. The third row contains '2', '"ELIGIBLE"', '"FULL", "MINI"', and 'BUREAU'. The fourth row contains '3', '"ELIGIBLE"', '"NONE"', and '-'. A dropdown menu is open for the '-' value in the 'Bureau call type' column, showing the options: '-', 'DECLINE', 'BUREAU', and 'THROUGH'.

	Eligibility	Bureau call type	Strategy
U	"ELIGIBLE", "INELIGIBLE"	"FULL", "MINI", "NONE"	DECLINE, BUREAU, THROUGH
1	"INELIGIBLE"	-	DECLINE
2	"ELIGIBLE"	"FULL", "MINI"	BUREAU
3	"ELIGIBLE"	"NONE"	-

注记：默认的“-”表示“未定义”。

信息需求

在决策层次结构中，决策可能会访问所需的决策和输入数据；这些必需的元素构成了可供决策使用的变量列表。



在这个例子中，决策“资格”需要两个决策——“局前风险类别”和“局前负担能力”——以及一个输入数据项“申请人数据”。

为调用的 BusinessKnowledgeModel 'Eligibility rules' 设置绑定值时，会提示选择一个自动完成列表。在此列表中，有子决策名称 - 输入数据的叶组件。使用此特征，您可以轻松设置调用。

DMN 表达式验证

DMN 定义了很多表达式，例如 `FunctionDefinition`、`DecisionTable`、`Boxed Context`、`Invocation` 和 `Literal Expression`。这些表达式的参数、参数和逻辑主要由“文本”实现。

为了使建模更容易和更可靠，Enterprise Architect提供了两个特征：自动完成和验证。

- 验证：识别由拼写错误、逻辑不完整、不一致等引起的建模错误
- 自动完成：您可以从枚举列表中选择一个文本string，而不是在本主题中，我们将向您展示如何验证DMN 表达式。

访问

DMN 表达式窗口	仿真>决策分析> DMN > DMN 表达式：验证按钮
DMN仿真窗口	仿真>决策分析>DMN>打开DMN仿真>仿真：验证图标

公共验证

变量名称Validation

The screenshot shows the 'DMN Expression' editor window. The main area displays a table with the following content:

(Input 1, Input 2)	
Local Variable 1	Input 1 - Input 2
Local Variable 2	Input 1 + ceiling(Input2)
Local Variable 1 + LocalVariable 2	

Below the editor, the 'System Output' window is open, showing the following validation results:

```
Running BKM1 Validations ...
Validating BusinessKnowledgeModel 'BKM1' ...
Warning : 'BKM1': Context Entry #1 : 'Expression can not be parsed. ' Input 1 - Input 2''
Warning : 'BKM1': Context Entry #2 : "'Input2' can not be parsed'
Warning : 'BKM1': Context Entry #3 : 'Expression can not be parsed. ' Local Variable 1 + LocalVariable 2''
BKM1 Results: (0) error(s), (3)warning(s)
```

在本例中，`Boxed Context`业务知识模型BKM1定义了两个参数 输入1 “和 输入2” ，以及两个局部变量 局部变量1 “和 局部变量2” 。表达 已经过验证，结果输出到系统输出窗口的 `DMN Validation`”选项卡。

- 上下文条目 #1 失败，因为存在印刷错误；它应该是运算符'-'，但用户在'-'中键入或复制
- Context Entry #2 失败，因为 'Input' 和数字 2 之间没有空格；注记函数'ceiling()' 是在 DMN 库中定义的，因

此可以成功解析

- 上下文条目 #3 失败，因为“本地”和“变量”之间没有空格

很难直观地识别这些类型的错误。运行验证可以帮助识别错误，然后您可以轻松地执行更正。

依赖验证

A 决策可能需要其他决策、输入数据和业务知识模型；这些关系由 InformationRequirement 和 KnowledgeRequirement 连接器标识。

当图表变得复杂时，很可能缺少某些连接器或使用了错误的连接器类型。

The screenshot displays a DMN diagram on the left and a 'DMN Expression' window on the right. The diagram shows a flow from BKM2 to Decision1, and from Decision2 and InputData1 to Decision1. Decision3 is also connected to Decision1. The 'DMN Expression' window shows the following table:

Decision1		
Local Variable 1	BKM2	
	Input 1 : number	Decision2
	Input 2 : number	Decision3
Local Variable 1 + InputData1		

The 'System Output' window shows the following validation results:

```

Running Decision1 Validations ...
Validating Decision 'Decision1' ...
Warning : 'Decision1': Context Entry #1 : 'Binding #2 : "Decision3" can not be parsed"
Error : Connector from 'BKM2' to 'Decision1' should be a 'KnowledgeRequirement'
Decision1 Results: (1) error(s), (1)warning(s)
  
```

在本例中，单击验证按钮，Enterprise Architect将显示：

- 'Decision3' 由'Decision1' 通过绑定到被调用的BKM2 的参数来使用；但是，它没有定义 - 缺少 InformationRequirement 连接器
 - “Decision1”中定义的调用无效；从 BKM2”到 Decision1”的连接类型应该是 KnowledgeRequirement
- 解决这些问题后，再次运行验证：

The screenshot displays the corrected DMN diagram on the left and the 'DMN Expression' window on the right. The diagram shows a flow from BKM2 to Decision1, and from Decision2 and InputData1 to Decision1. Decision3 is also connected to Decision1. The 'DMN Expression' window shows the same table as in the previous screenshot.

The 'System Output' window shows the following validation results:


```

Running Decision1 Validations ...
Validating Decision 'Decision1' ...
Decision1 Results: (0) error(s), (0)warning(s)
  
```

盒装上下文

上下文Context 是时间条目A集合，以表的形式呈现，后跟最终结果表达式。

这些上下文条目由一个与值表达式配对的变量组成，可以被认为是中间结果。这允许将复杂的表达式分解为一系列简单的表达式，最终结果以更简单的形式进行评估。

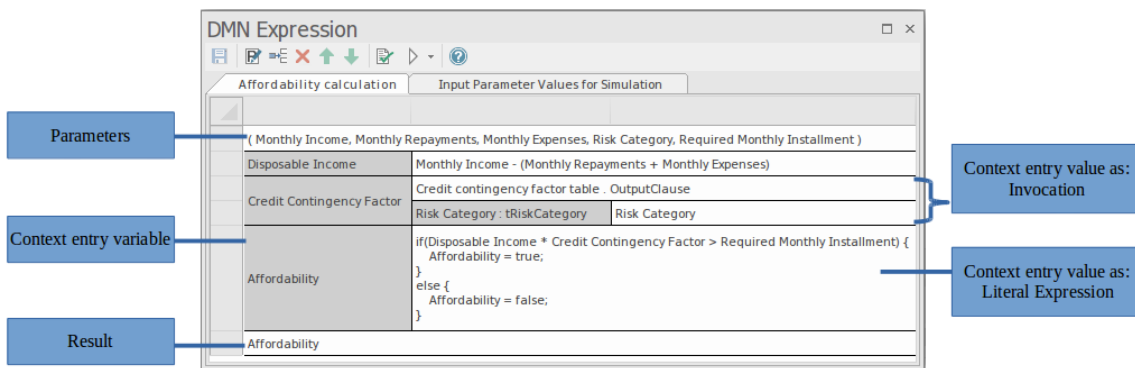
决策和决策业务知识模型元素类型都支持盒装上下文类型。它由  图标表示。

访问

图表	在图表上，双击决策元素或 BKM元素。 随即显示DMN 表达式编辑器窗口，显示所选元素的详细信息。
----	--

概述

此图像显示了为盒装上下文表达式显示的DMN 表达式编辑器窗口。

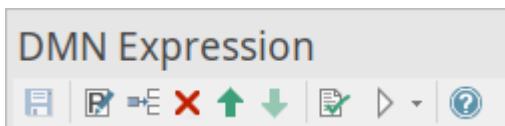


上下文Context 是时间条目A集合，以表的形式呈现，后跟最终结果表达式。每个上下文条目由一个变量和一个值表达式组成。该变量可以被认为是一个中间结果，它可以在任何后续上下文条目的值表达式中使用。上下文条目的值表达式可以是文字表达式或调用，并且可以使用任何可用的输入，例如参数（到 BKM元素）、InputData 或决策结果，以及任何先前定义上下文变量。

Boxed Context 表达式的最终结果是通过依次处理每个上下文条目、评估值表达式并将其结果分配给变量，然后最终评估结果表达式来确定的。结果表达式也可以使用任何输入或局部变量，但必须计算以提供结果。

盒装上下文编辑器的工具栏

选择布局表达式时，DMN 表达式窗口中可访问的地形特征为：

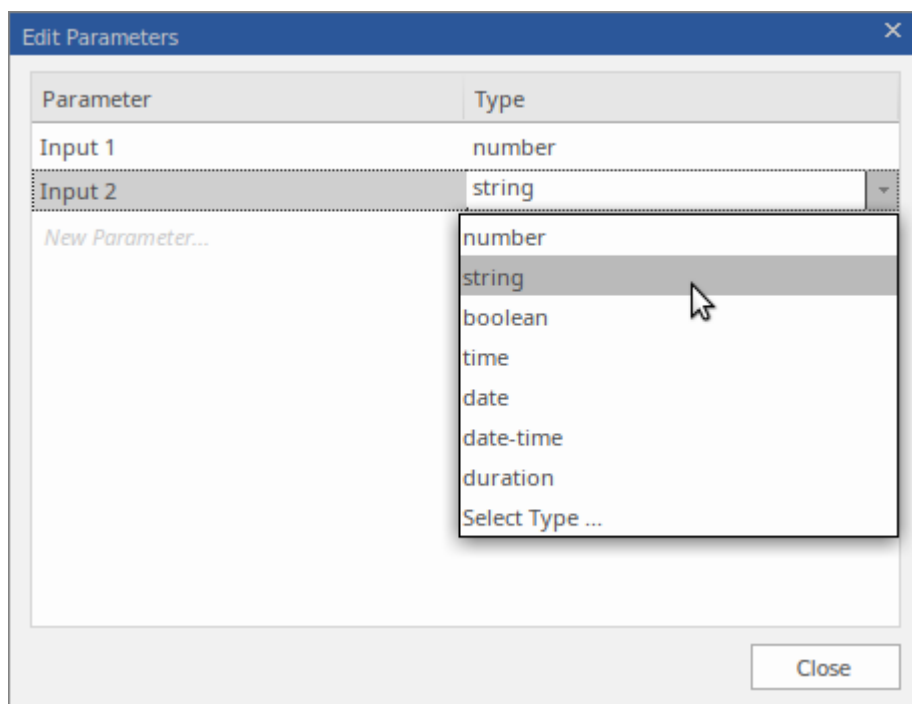


更多详情请参考帮助主题 [帮助Context Editor 的工具栏](#)。

指定参数

对于 BusinessKnowledgeModel 元素，参数用于传递调用元素提供的输入值。使用输入参数评估 BKM 的决策逻辑，并将结果返回给调用元素。默认情况下，使用两个输入参数 `Input 1` 和 `Input 2` 创建 BKM 元素。

单击 DMN 表达式窗口工具栏中的  图标，显示“编辑参数”窗口。



您可以在此处更改参数名称、设置其数据类型、创建附加参数或删除现有参数。

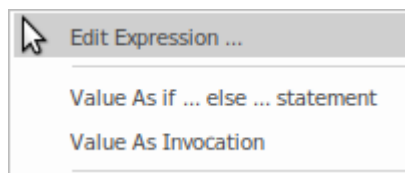
指定上下文条目

每个上下文条目都包含一个变量-表达式对。

变量名可以是您喜欢的任何文本，甚至可以包含空格。要编辑变量名称，单击单元将其选中，然后再次单击或按 F2 进入编辑模式。要退出编辑模式，请单击其他位置或按 Enter 键。

一般，没有必要为表达式或变量指定数据类型 - 类型将从值中推断出来。但是，如果您打算为 Java、C++ 或 C# 等编译语言生成代码，则必须指定所有上下文条目变量的类型。

上下文条目的值表达可以是文字表达或调用，并且可以利用任何可用的输入，例如参数（对业务知识模型元素）、输入数据或决策结果，以及任何先前定义上下文变量。右键单击表达式单元会显示一个弹出菜单，该菜单提供用于显示表达式代码编辑器或将值表达式设置为 If-Else 语句或调用的选项。



您还可以通过在表达式中直接输入文本来编辑值单元。

有关如何指定文字表达式或调用的更多信息，请参阅帮助这些主题的帮助主题。

盒装上下文编辑器的工具栏

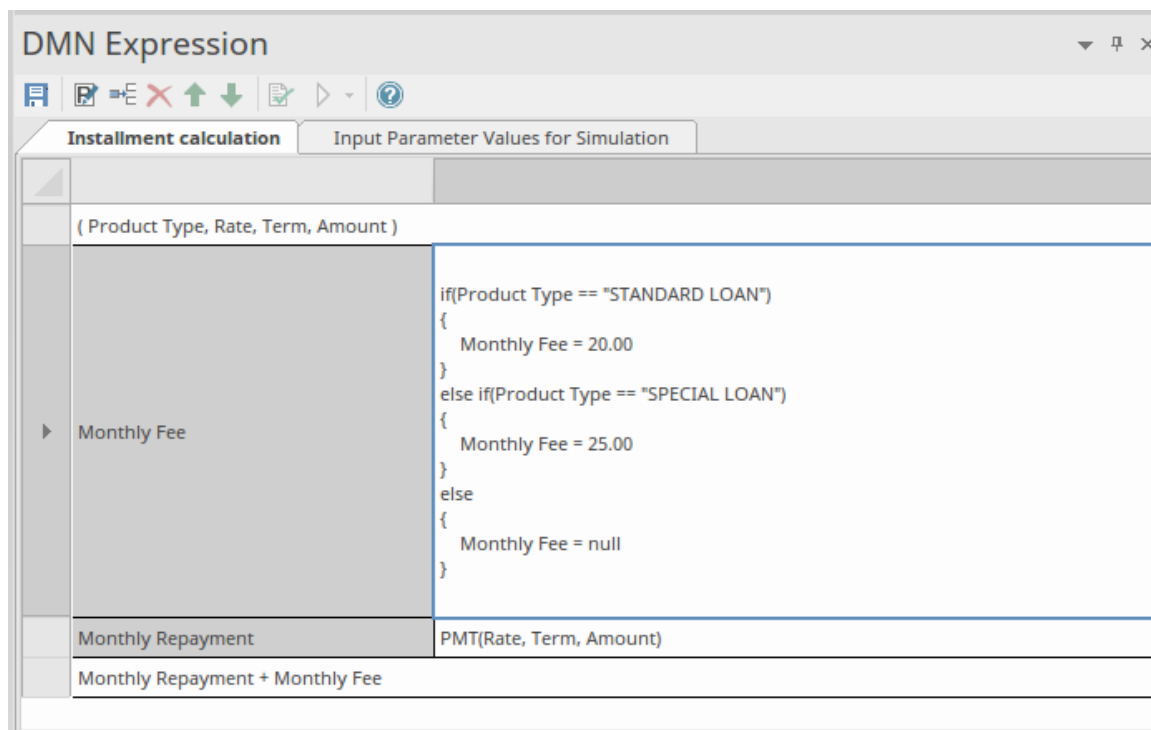
此表提供了在选择盒装上下文时在DMN 表达式窗口中可访问的特征的描述。

工具栏选项

此工具栏适用于盒装上下文。

选项	描述
	保存对当前选择的决策或 BusinessKnowledgeModel元素的更改。
	显示“编辑参数”窗口，您可以在其中指定调用 BusinessKnowledgeModel元素的决策逻辑时传递的每个参数的名称和数据类型。
	创建一个新的上下文条目并将其附加到上下文列表中。
	删除当前选定的上下文条目。
	将当前选定的上下文条目在列表中上移一位。
	将当前选定的上下文条目在列表中下移一位。
	执行BoxedContext的验证。Enterprise Architect将执行一系列验证，以帮助您发现BoxedContext定义中的任何错误。
	<p>该按钮在为知识业务知识模型元素定义业务决策决策表时启用。</p> <p>选择“参数for仿真”选项卡，完成字段，然后单击此按钮。测试结果将显示在决策表上，显示输入和输出的运行时值，并突出显示有效规则。</p> <p>您可以使用此功能对 BusinessKnowledgeModel元素进行单元测试，而无需指定其上下文。</p> <p>此工具栏按钮A许多菜单选项可用。有关详细信息，请参阅帮助主题仿真帮助仿真模型</p>

示例-贷款分期计算



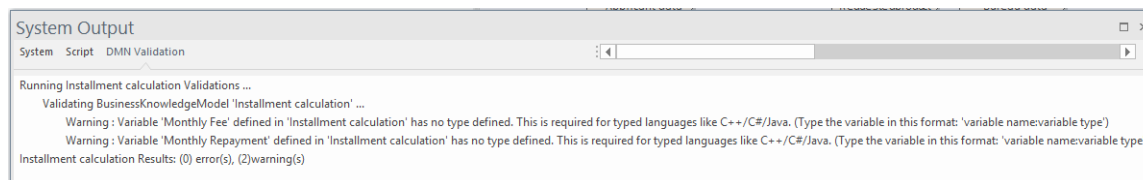
业务知识模型 (BKM) 分期计算以盒装上下文的形式实现。

- BKM 定义了四个参数：类型、Rate、团队和 Amount
- Boxed Context 定义了两个变量-表达式对条目；这些变量充当“局部变量”，可以在以后的表达式中使用
- 返回值：表达式可以使用‘局部变量’的值
- 盒装上下文中的任何表达式都可以使用内置函数，这些函数在可定制的模板 - DMN 库中定义；本例中使用了函数 PMT(...) 和 decimal(...)

为上下文条目变量指定类型

一般，表达式和变量不必指定类型，类型是从提供的值推断出来的。此特征一般由JavaScript支持，用于Enterprise Architect的DMN仿真。

但是，如果您想将DMN模型中的代码生成为Java、C++或C#等编译语言，则必须为每个上下文条目变量指定类型。否则，如果您验证模型，您将看到如下警告：



右键单击此模型中的上下文条目变量（月费、月还款）。

The screenshot shows the 'DMN Expression' window with the 'Installment calculation' tab selected. The expression is defined as follows:

```
( Product Type, Rate, Term, Amount )  
  
if(Product Type == "STANDARD LOAN")  
{  
  Monthly Fee = 20.00  
}  
else if(Product Type == "SPECIAL LOAN")  
{  
  Monthly Fee = 25.00  
}  
else  
{  
  Monthly Fee = null  
}  
  
Monthly Repayment PMT(Rate, Term, Amount)  
  
Monthly Repayment + Monthly Fee
```

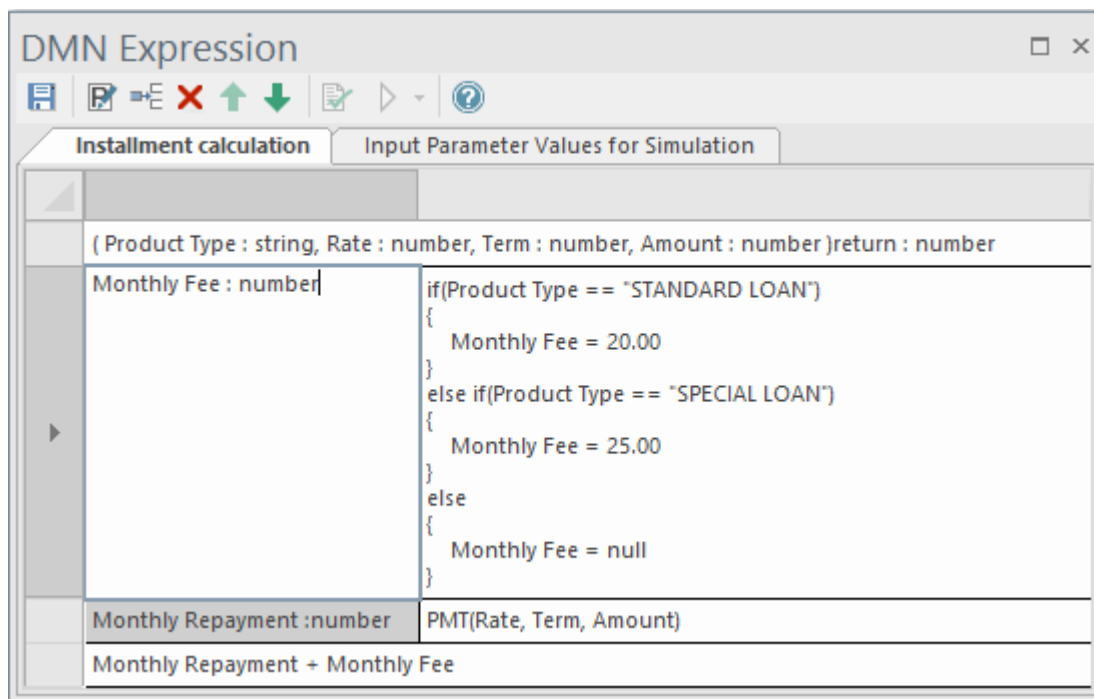
选择“显示变量类型”选项。


The screenshot shows the same DMN Expression editor, but with a context menu open over the 'Monthly Fee' variable. The menu options are:

- Delete Local Variable
- Move Up Local Variable
- Move Down Local Variable
- Show Variable Type
- Show Simulation Runtime Value

The 'Show Variable Type' option is highlighted, and a 'Show Variable Type' button is visible in the bottom right corner of the editor area.

现在输入变量类型，将其附加到变量名称并用冒号分隔，如此处所示。

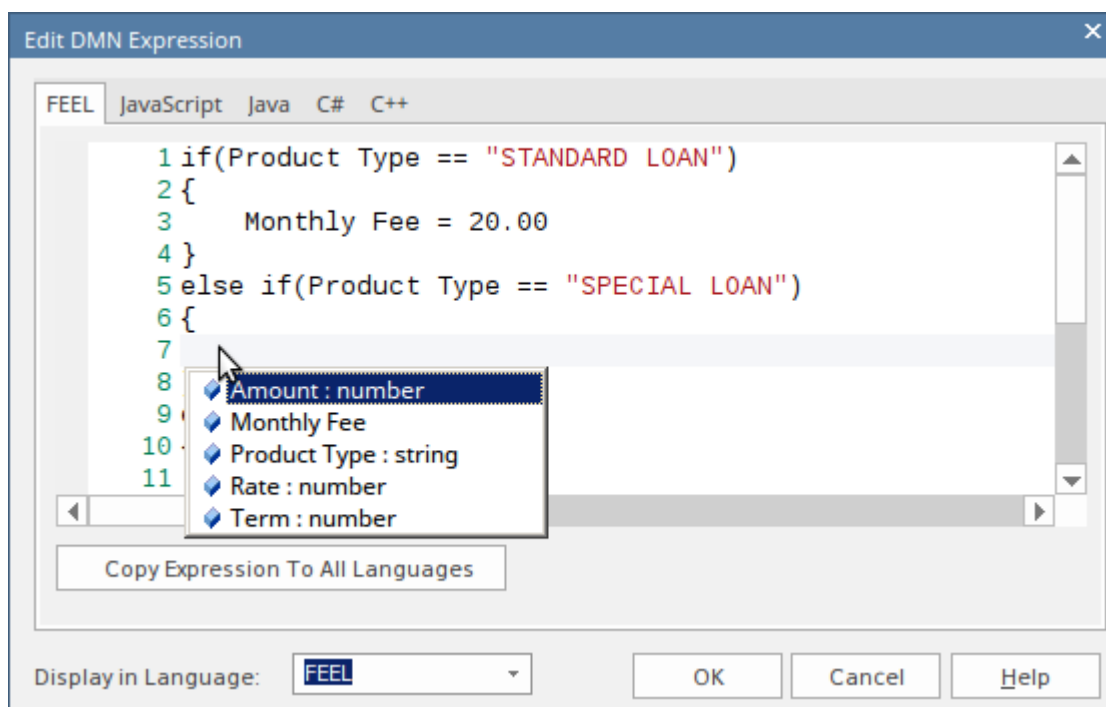


然后单击工具栏上的 Save 按钮保存表达式，然后单击  按钮再次验证模型。

表情编辑器和智能感知支持

根据 FEEL 语言规范，参数和上下文条目的变量名称可以包含空格。这个特征使表达式易于阅读。为了帮助您用更少的输入和更少的错误编辑表达式，Enterprise Architect 智能感知了对编辑表达式的支持：

要编辑表达式，请右键单击表达式（在右侧字段中）并选择 编辑表达式”菜单选项。将显示 表达式”对话框。单击并按 Ctrl+Space 以显示所需的智能感知菜单行：



- 将包含所有早于当前的上下文条目变量（不包括晚于当前的上下文条目变量）
- 对于业务知识模型（BKM），将包含所有参数

- 对于决策，将包括所有必需的决策

DMN模型可以生成JavaScript、Java、C#和C++的源代码。由于某些语言可能对某些表达式有不同的语法，Enterprise Architect为每种语言提供了语言覆盖页面。如果没有为语言指定覆盖代码，则将使用为FEEL语言定义的表达式。

在生成的代码中，变量名中的空格将被替换为下划线。

业务知识模型仿真

选择“用于仿真的输入参数值”选项卡并完成每个字段。

Installment calculation		Input Parameter Values for Simulation	
Installment calculation			
Product Type : string	"STANDARD LOAN"		
Rate : number	0.045/12		
Term : number	12*30		
Amount : number	300000		


单击保存按钮，然后单击工具栏上的仿真按钮；测试结果将显示在Boxed Context表达式中。

Installment calculation		Input Parameter Values for Simulation	
(Product Type : string = "STANDARD LOAN", Rate : number = 0.00375, Term : number = 360, Amount : number = 300000)return : number = "1540.06"			
Monthly Fee : number = 20	<pre> if(Product Type == "STANDARD LOAN") { Monthly Fee = 20.00 } else if(Product Type == "SPECIAL LOAN") { Monthly Fee = 25.00 } else { Monthly Fee = null } </pre>		
Monthly Repayment : number = 1520.0559294776567	PMT(Rate, Term, Amount)		
decimal(Monthly Repayment + Monthly Fee,2) (evaluation result = "1540.06")			

- 将显示运行时参数值；例如，速率 = 0.00375”
 - 将显示“上下文条目”变量的运行时值；例如，每月还款额 = 1520.06”
 - 业务知识模型 (BKM) 的结果将根据最后一个条目和申报行显示的值进行评估；例如，'return = 1540.06'
- 您可以使用此功能对BKM进行单元测试，而无需了解上下文，以便稍后可以由决策或另一个BKM调用。

决策表

决策表是一组相关输入和输出表达式的表格表示，组织成规则，指示哪个输出条目适用于一组特定的输入条目。

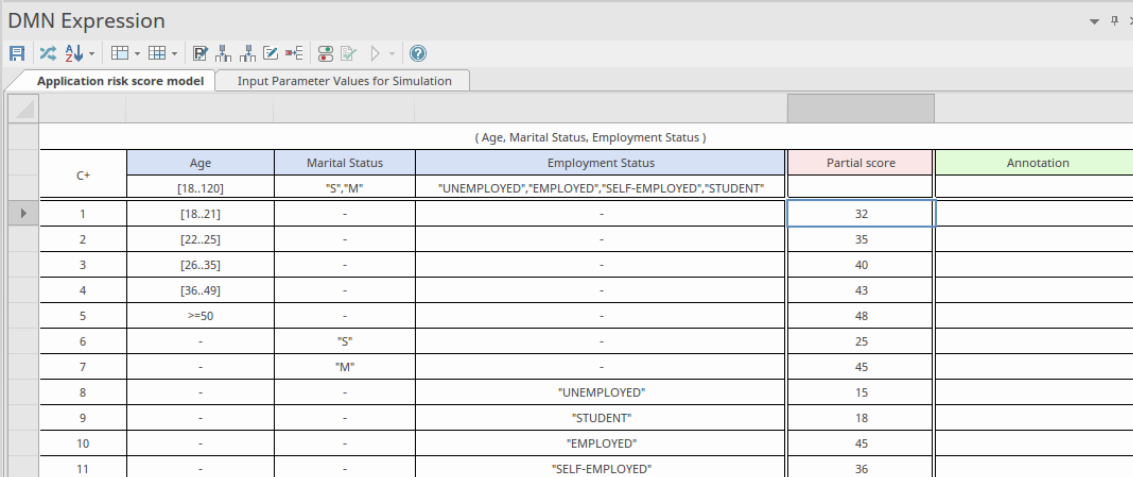
决策表由决策和决策业务知识模型元素类型提供支持。它们由图上元素右上角的  图标表示。

访问

图表	在图表上，双击决策元素或 BusinessKnowledgeModel 元素。显示 DMN 表达式窗口，显示所选元素的详细信息。
----	--

概述

此图显示了决策表的 DMN 表达式窗口。



The screenshot shows a window titled "DMN Expression" with a toolbar and a table. The table is titled "Application risk score model" and "Input Parameter Values for Simulation". It has a header row with columns: C+, Age, Marital Status, Employment Status, Partial score, and Annotation. The table contains 11 rows of data.

(Age, Marital Status, Employment Status)					
C+	Age	Marital Status	Employment Status	Partial score	Annotation
	[18..120]	"S","M"	"UNEMPLOYED","EMPLOYED","SELF-EMPLOYED","STUDENT"		
1	[18..21]	-	-	32	
2	[22..25]	-	-	35	
3	[26..35]	-	-	40	
4	[36..49]	-	-	43	
5	>=50	-	-	48	
6	-	"S"	-	25	
7	-	"M"	-	45	
8	-	-	"UNEMPLOYED"	15	
9	-	-	"STUDENT"	18	
10	-	-	"EMPLOYED"	45	
11	-	-	"SELF-EMPLOYED"	36	

决策表包括A

- 指定如何应用规则的库表命中策略 (C+、U、A、P 等)
- 规则A (1等)，其中每个规则行包含特定的输入条目和相应的输出条目
- 输入子句A (在蓝色标题下)，定义为通常涉及一个或多个输入值的表达式
- 输出子句A (在粉红色标题下)，定义对应于一组特定输入的输出
- 您可以通过窗口工具栏添加的每个规则的可选注释 (在绿色标题下)

输入子句由一个表达式和一个可选的允许值列表 (列标题下方的行) 组成。很多时候，表达式只是一个未经修改的输入值；但是，它也可能是一个涉及多个输入值的表达式，或者可能是一个条件语句，例如 **Application 风险 Score > 100**”。允许的值适用 表达式结果，而不是使用的输入值。

每个输出子句由一个标识符 (名称) 和该子句允许值的可选列表组成。

决策表应包含确定输出所需的所有输入——并且仅包含那些输入。

在确定应用哪些规则时，输入子句中定义的表达式会针对给定的输入进行评估，然后使用表达式结果来查找具有匹配输入条目的规则。

如果 DMN 表达式窗口不够宽或不够深，无法显示所有列和行，您可以使用滚动条来访问隐藏的内容，或者将边框拖出以增加每列的宽度。输入“和 输出”列宽最初是相同的，但您可以通过在表内或在选项卡名称正下方的

灰色栏中拖动列边框来独立调整每个列宽。

决策表编辑器工具栏

选择决策表后，可以通过窗口顶部的工具栏访问DMN 表达式窗口中可用的特征，如下所示：

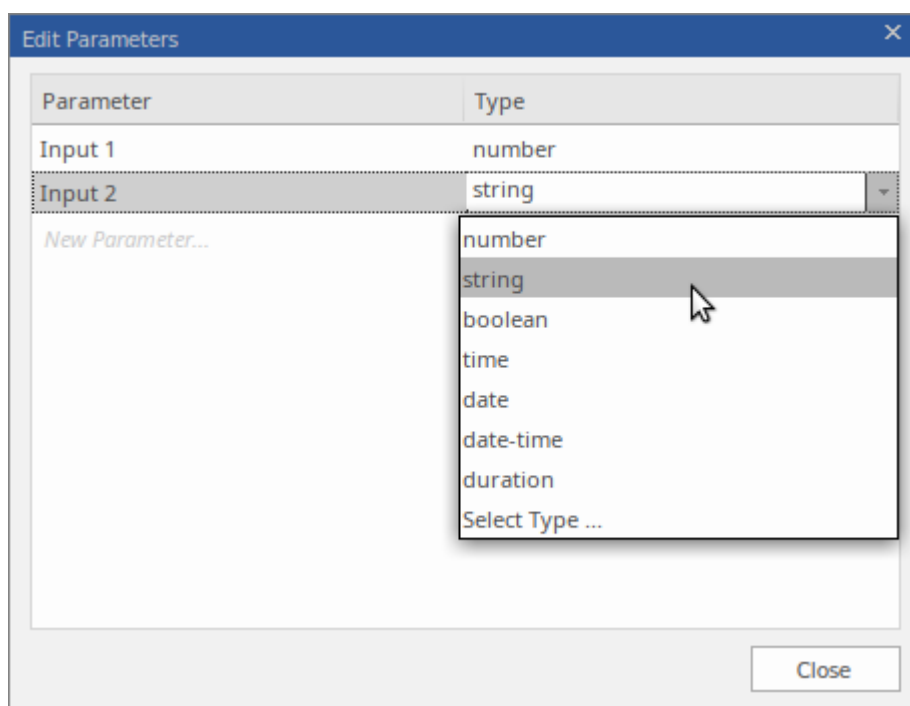


有关更多详细信息，请参阅决策表编辑器帮助主题的工具栏帮助

参数

在业务知识模型 (BKM) 元素的情况下，参数用于传递调用元素提供的输入值。使用输入参数评估 BKM 的决策逻辑，并将结果返回给调用元素。默认情况下，使用两个输入参数 `Input 1` 和 `Input 2` 创建 BKM 元素。

单击DMN 表达式窗口工具栏中的  图标以显示“编辑参数”对话框。



您可以在此处更改参数名称、重新排列序列、设置其数据类型、创建附加参数或删除现有参数。

命中策略

右键单击“命中策略指示器”，然后从弹出菜单中选择所需的命中策略目标。各种库表命中策略在细节决策表命中策略帮助中详细描述。

输入子句

决策表的输入子句被定义为一个表达式。很多时候，表达式只是一个未经修改的输入值；但是，它也可以是一个涉及多个输入值的表达式，或者可以定义为条件语句，例如“`风险 > 100`”。允许的值适用 表达式结果而不是使用的输入值，因此，值的类型应与表达式结果的类型相匹配。

决策表是用两个默认的输入子句创建的，输入1”和 输入 2”。这两个子句的数据类型都是 数字”。在DMN 表达式窗口中，输入子句在决策表上显示为列标题。要修改输入子句，请单击列标题以选择单元，然后再次单击或按 F2 进行编辑。

编辑输入子句时支持自动补全。这意味着，对于决策元素，任何连接到决策元素的输入都可以从列表中进行选择。类似地，对于业务知识模型元素，调用参数可用于从列表中选择。有关详细信息，请参阅帮助DMN 表达式自动完成帮助主题。

(Flights in the last month, Number of Pax Overbooked)			
U	Flights in the last month		
		Flights in the last month	
1	-	Number of Pax Overbooked	
2	-	-	

要向  表添加其他输入条目列，请单击DMN 表达式窗口工具栏上的决策表图标。

要从表中删除输入列，请在不需要的输入列中单击鼠标右键，然后从弹出菜单中选择 删除输入列”选项。

可以通过将列拖放到新位置来重新排列表中输入列的顺序。（将表列最顶部未标记的单元所需位置。）

允许值

在定义 输入”或 输出”列时，该列的第二行定义了允许值。这是列中的一个可选单元，但对于澄清其下方行中的条目很有用。运行验证时，将检查允许值单元的每个单元，以确保它们符合此单元中的表达式。

此单元中使用的表达式取决于 输入”或 输出”列的键入方式。例如：

- 数字 - [18 ..35]
- 字符串- 高”、 低”、 中”
- 布尔值 - 真、 false
- 不明确的 - '!

快速填充允许值

这里引用的输入/输出表达式可以是一个简单的值，也可以是一个复杂的FEEL表达式；但是，如果它与 ItemDefinition 的 允许值”字段直接相关，则按下S速度条将启用快速填充选项来设置ItemDefinition中定义的 允许值”（通常通过 InputData元素引用）。

U	Temperature
1	Hot, Warm, Frozen, Cold

快速填充行

一旦定义了 允许值”字段，以及限制在表中定义规则时可以使用的值， 允许值”字段还为用户提供了快速填充选项。这在规则单元中通过按下S速度条并选择所需项目来调用：

U	Temperature	
	Hot, Warm, Frozen, Cold	
1	-	-
2	-	-
3	-	-

Hot
 Warm
 Frozen
 Cold

更多详情请参见帮助主题帮助DMN 表达式自动完成。

输出子句

输出子句由名称、数据类型和可选的允许值列表组成。要修改输出子句，请单击列单元以选择单元，然后再单击或按 F2 进行编辑。

要向  决策表。

要从表中删除输出列，请在不需要的输出列中单击鼠标右键，然后从弹出菜单中选择选项“删除输出列”。

可以通过将列表到新位置来重新排列表中列的顺序。（将表列最顶部的未标记单元所需位置。）

输入/输出子句的数据类型

为了使模拟工作，为所有输入和输出子句设置数据类型至关重要。'number' 类型的子句支持范围、间隙和重叠验证，但如果未指定类型，则无法执行验证。C++、C# 和 Java 等类型语言的代码生成要求指定数据类型。当数据类型指定为 'string' 时，不需要将每个 string 文字括在引号中。如果类型已声明，字符串值以斜体显示。

要设置数据类型，请右键单击输入或输出列标题，然后从列表中选择所需的类型。


Credit contingency factor table		Input Parameter Values for Simulation
		(Risk Category)
U	Risk Category	<i>DECLINE, HIGH, MEDIUM, LOW, VER</i>
1		<i>HIGH, DECLINE</i>
2		<i>MEDIUM</i>
3		<i>LOW, VERY LOW</i>

type: string
 type: boolean
 type: number
 type: date
 type: time
 type: duration

定义决策表规则

决策表规则是通过在一个表行的单元内指定输入条目和相应的输出条目来定义的。对于“数字”数据类型，可以


将输入条目指定为单个值或数字范围，例如 “<10”、“>100”或 “[2..8)”。（定义数字范围时，使用round表示不包括边界数字；使用方括号表示包括边界数字。）输出条目应指定每个单元格的单个值。

通过单击工具栏中的  图标，可以将其他规则附加到规则列表中。通过右键单击规则并从弹出菜单中选择“删除规则行”选项，可以从表中删除不需要的规则。

可以通过首先选择规则来复制和粘贴现有表（使用“Ctrl+单击”从选择中添加/删除），然后使用菜单选项“复制规则到剪贴板”和“粘贴规则从剪贴板”到执行复制和粘贴。然后可以通过选择和编辑单个单元格来修改复制的规则。

如果为string或布尔表达式设置了“允许值”字段，则空格键可用于显示可供选择的值列表，如前面的“允许值 - 快速填充行”部分所示。

规则也可以在表中排序，方法是：

- 单击工具栏上的  图标，然后选择“按输入排序”或“按输出排序”，或
- 右键单击表中的单个规则，然后从弹出菜单中选择“上移规则”或“下移规则”选项

为了确定选择哪些表行进行输出，由输入子句定义的表达式针对给定的输入进行评估，然后将表达式的结果与表行的输入条目进行比较。如果表达式结果与表行的输入条目匹配，则选择该行进行输出。

决策表的“命中策略”决定了匹配行如何使用决策表来产生其输出。

规则格式

您可以选择 - 使用工具栏图标 - 以三种格式之一显示决策表，如下所示。

Rule-as-Row 格式，其中规则沿行开发，输入、输出和注释设置在列中：

(Existing Customer, Application Risk Score)				
U	Existing Customer	Application Risk Score	Pre-Bureau Risk Category	Annotations
	true,false		HIGH, MEDIUM, LOW, VERY LOW, DECLINE	
1	true	<80	DECLINE	Use standard letter DEC0004
2	true	[80..90)	HIGH	
3	true	[90..110]	MEDIUM	
4	true	>110	LOW	
5	false	<100	HIGH	
6	false	[100..120)	MEDIUM	
7	false	[120..130]	LOW	
8	false	>130	VERY LOW	Refer to Loans Officer

Rule-as-Column 格式，其中规则在列下展开，输入、输出和注释沿行设置：

(Existing Customer, Application Risk Score)									
Existing Customer	true,false								
Application Risk Score		true	true	true	true	false	false	false	false
Pre-Bureau Risk Category	HIGH, MEDIUM, LOW, VERY LOW, DECLINE	DECLINE	HIGH	MEDIUM	LOW	HIGH	MEDIUM	LOW	VERY LOW
Annotations		Use standard letter DEC0004							Refer to Loans Officer
U		1	2	3	4	5	6	7	8

Rule-as-Crosstab 格式，其中规则由定义为一组行和列组合的输入形成，输出设置在相交单元格中。（注记这种格式隐藏了“注释”字段）：

(Existing Customer, Application Risk Score)								
Pre-Bureau Risk Category	Application Risk Score							
	<80	[80..90)	[90..110]	>110	<100	[100..120)	[120..130]	>130
Existing Customer	false							
	true	DECLINE	HIGH	MEDIUM	LOW	HIGH	MEDIUM	LOW

在模拟结束时，在交叉决策表中，相关的输入条目和输出条目被突出显示。例如，在此模拟处理中，订单大小小于或等于 10 顾客适用交货的客户业务输出了 0.10 的折扣。

The screenshot shows a software interface titled "DMN Expression" with a toolbar and a decision table. The table has a header row with "Discount" and "Customer, Delivery". The "Customer, Delivery" header is split into three columns: "Business", "Private", and "Government". The "Business" column has a sub-header "sameday" and a value "slow". The "Private" column has a sub-header "slow" and a value "-". The "Government" column has a sub-header "-" and a value "-". The "Discount" column has two rows: "<10" and ">=10". The values in the "Discount" column are 0.05 and 0.10 respectively. The "OrderSize" column has two rows: "<10" and ">=10". The values in the "OrderSize" column are 0.05 and 0.10 respectively.

Discount		Customer, Delivery		
		Business	Private	Government
		-	sameday	slow
OrderSize	<10	0.05	0	0.05
	>=10	0.10	0	0.05
				0.15

交叉表设置

在 Rule-as-Crosstab 格式中，由于输入形成行和列，输出位于交叉点，因此设置值的步骤与其他两种格式略有不同。

- 要添加其他类型的输入，请右键单击输入列标题并选择“添加输入”选项。系统会提示您输入输入名称；输入被添加为当前列字段下的一组字段。
要从列中删除输入类型，请右键单击其字段集并选择“删除输入”选项。输入的名称及其字段集将从列标题中删除。
- 要添加其他类型的输出，请右键单击窗口左上角的输出块，然后选择“添加输出”选项。系统会提示您输入输出名称；名称将添加到输出块，并在窗口主体中的每个单元中添加一个新行。
要删除输出类型，请右键单击窗口左上角输出块中的类型名称，然后选择“删除输出”选项。网格中的输出名称及其字段将被删除。
- 您可以在输入类型之间旋转以选择一种作为行标题。右键单击该行，然后单击“选择输入作为行标题”选项。这将显示输入类型的列表；单击要用作行标题的类型；其他类型组合在列中。
- 要将值条目行或列添加到输入，请右键单击当前行或列，然后根据需要选择“添加输入条目行”或“添加输入条目列”选项。显示输入条目名称A提示；当您输入此信息时，相应的行或列将添加到决策表中。
要删除值条目行或列，请右键单击它并选择“删除输入条目行”或“删除输入条目列”选项。从表中删除选定的行或列。
- 在输入列中，每一行都匹配一种输入。如果要将一种输入类型的行移动到另一种类型的上方或下方，请右键单击它并选择“上移输入”或“下移输入”选项。上下文菜单仅提供可操作的选项，因此无法将最后一行向下移动，因此未列出“向下移动输入”选项。

决策表编辑器工具栏

This table provides descriptions of the features accessible in the DMN Expression window when a decision table is selected.

工具栏选项

图标	描述
	保存对当前选择的决策或 BusinessKnowledgeModel 元素的更改。
	在 Rule-as-Row、Rule-as-Column 和 Rule-as-Crosstab 之间切换决策表的视图。或者，单击下拉箭头并选择您需要的格式。
	单击“按输入排序”以按输入列对规则进行排序；单击“按输出排序”以按输出列对规则进行排序。可以拖放列来组织排序顺序。
	相邻规则的合并单元，其中输入条目的内容相同。您可以编辑合并后的内容。单元。在模拟过程中，合并的项目会突出显示。
	单元先前已合并的输入条目。
	显示“编辑参数”窗口，您可以在其中指定调用 BusinessKnowledgeModel 元素的决策逻辑时传递的参数名称和数据类型。
	将输入列附加到决策表。
	将输出列附加到决策表。
	在表格后面添加一个注释栏（带有绿色标题单元），可以在其中记录表的笔记或对规则的评论。（参见前面的 Rule-as-Row/Rule-as-Column 行中的插图。）如果需要，您可以添加多个注释列，在每个标题单元中键入适当的列标题。 要删除注释列，请右键单击它并选择“删除注释列”选项。
	将规则附加到决策表。
	显示或隐藏“输入”和“输出”列的允许值字段。 为输入或输出定义的允许值将用于验证和自动完成编辑。
	执行决策表的验证。Enterprise Architect 将执行一系列验证，以帮助您发现决策表中的任何错误。
	在为 BusinessKnowledgeModel 元素定义决策表时启用此按钮。 选择“参数 for 仿真”选项卡，完成字段，然后单击此按钮。测试结果将显示在决策表上，显示输入和输出的运行时值，并突出显示有效规则。 您可以使用此功能对 BusinessKnowledgeModel 元素进行单元测试，而无需指定其上下文。 此工具栏按钮 A 许多菜单选项可用。有关详细信息，请参阅帮助主题仿真帮助

	仿真模型
--	------

目标决策表命中策略

命中策略规定了规则重叠情况下的决策决策表结果。特定决策决策表单元中的单个字符表示表的类型，并明确地反映决策逻辑。

命中策略：

- **U**性：不可能有重叠，所有规则都是不相交的；只能匹配一条规则（这是默认设置）
- **A ny**：可能存在重叠，但所有匹配规则显示每个输出的输出条目相同，因此可以使用任何匹配
- **优先级**：可以匹配多个规则，输出条目不同；该策略返回输出优先级最高的匹配规则
- **第一**：可以匹配多个（重叠）规则，具有不同的输出条目；返回规则顺序的第一个命中

多重命中策略：

- **O**顺序：按输出优先级降序返回所有命中
- **R**顺序：按规则顺序返回所有命中
- **收集**：以任意顺序返回所有命中；可以添加一个运算符（'+', '<', '>', '#'）来对输出应用一个简单的函数

收集运算符是：

- **+** (sum)：决策表的结果是所有不同输出的总和
- **<** (min)：决策表的结果是所有输出中的最小值
- **>** (max)：决策表的结果是所有输出中的最大值
- **#** (count)：决策表的结果是不同输出的数量

示例命中策略的示例

独特”的命中策略是决策决策表中最流行的类型，所有规则都是不相交的。

Post-bureau risk category table		Input Parameter Values for Simulation		
(Existing Customer = true, Application Risk Score = 90, Credit Score = 590)				
U	Existing Customer	Application Risk Score	Credit Score	Post Bureau Risk Category
	true	90	590	MEDIUM
1	false	< 120	< 590	HIGH
2	false	< 120	[590..610]	MEDIUM
3	false	< 120	> 610	LOW
4	false	[120..130]	< 600	HIGH
5	false	[120..130]	[600..625]	MEDIUM
6	false	[120..130]	> 625	LOW
7	false	> 130	-	VERY LOW
8	true	<= 100	< 580	HIGH
9	true	<= 100	[580..600]	MEDIUM
10	true	<= 100	> 600	LOW
11	true	> 100	< 590	HIGH
12	true	> 100	[590..615]	MEDIUM
13	true	> 100	> 615	LOW

示例命中策略的示例

在带有“优先级”命中策略的表中，可以匹配多个规则，并具有不同的输出条目。该策略返回输出优先级最高的

匹配规则。

Eligibility rules		Input Parameter Values for Simulation		
(Pre-Bureau Risk Category, Pre-Bureau Affordability, Age)				
P	Pre-Bureau Risk Category	Pre-Bureau Affordability	Age	Eligibility
				INELIGIBLE, ELIGIBLE
1	DECLINE	-	-	INELIGIBLE
2	-	false	-	INELIGIBLE
3	-	-	< 18	INELIGIBLE
4	-	-	-	ELIGIBLE

注记：允许值列表用于定义输出优先级。在这里，允许的值被列为 INELIGIBLE 和 ELIGIBLE；INELIGIBLE 被定义为具有比 ELIGIBLE 更高的优先级。

一种可能的模拟结果可能类似于：

Eligibility rules		Input Parameter Values for Simulation		
(Pre-Bureau Risk Category = "HIGH", Pre-Bureau Affordability = false, Age = 25)				
P	Pre-Bureau Risk Category	Pre-Bureau Affordability	Age	Eligibility
	HIGH	false	25	INELIGIBLE
1	DECLINE	-	-	INELIGIBLE
2	-	false	-	INELIGIBLE
3	-	-	< 18	INELIGIBLE
4	-	-	-	ELIGIBLE

匹配规则被突出显示，但选择规则 2 的输出，因为 INELIGIBLE 的优先级高于 ELIGIBLE。

示例-Sum命中策略的示例

对于具有“收集和”（C+）命中策略的决策决策表，决策决策表的结果是所有不同输出的总和。

Application risk score model		Input Parameter Values for Simulation		
(Age = 40, Marital Status = "M", Employment Status = "EMPLOYED")				
C+	Age	Marital Status	Employment Status	Partial score
	40	"M"	"EMPLOYED"	133
1	[18..21]	-	-	32
2	[22..25]	-	-	35
3	[26..35]	-	-	40
4	[36..49]	-	-	43
5	>=50	-	-	48
6	-	"S"	-	25
7	-	"M"	-	45
8	-	-	"UNEMPLOYED"	15
9	-	-	"STUDENT"	18
10	-	-	"EMPLOYED"	45
11	-	-	"SELF-EMPLOYED"	36

在此示例中，输出 Partial Score 计算为 43 + 45 + 45 = 133

决策表验证

决策表是用于表达决策逻辑的最常见和最有用的 DMN 表达式之一。然而，对决策表建模也可能很复杂，特别是如果多个输入子句被组合用于许多决策表规则。如本主题所述，Enterprise Architect 提供了验证决策表的功能。

访问

DMN 表达式窗口	仿真>决策分析> DMN > DMN 表达式：验证按钮
DMN 仿真窗口	仿真>决策分析>DMN>打开DMN仿真>配置：验证按钮

条目超出范围检测

为决策表的输入子句和输出子句定义“允许值”是一种很好的做法。“允许值”列表用 对表规则的输入和输出条目值进行范围检查。

DMN Expression

Application risk score model | Input Parameter Values for Simulation

(Age, Marital Status, Employment Status)				
C+	Age	Marital Status	Employment Status	Partial score
	[20..120]	S, M	UNEMPLOYED, EMPLOYED, SELF...	
1	[18..21]	-	-	32
2	[22..25]	-	-	35
3	[26..35]	-	-	40
4	[36..49]	-	-	43
5	>=50	-	-	48
6	-	S	-	25
7	-	M	-	45
8	-	-	UNEMPLOYED	15
9	-	-	STUDENT	18
10	-	-	EMPLOYED	45
11	-	-	SELF-EMPLOYED	36

Notes | DMN Expression

System Output

System | Script | DMN Validation | Help System

Running Application risk score model Validations ...
 Validating BusinessKnowledgeModel 'Application risk score model' ...
 Warning : DecisionTable "Application risk score model" Input Violation:Input Value is not allowed for "Rule[1].Age": [18..21]
 Warning : DecisionTable "Application risk score model" Input Violation:Input Value is not allowed for "Rule[12].Marital Status": "D"
 Application risk score model Results: (0) error(s), (2)warning(s)

在这个例子中：

- 'Age' 输入子句定义了 [20..120] 的范围；但是，规则1的输入条目指定了 [18..21] 的范围；这超出了允许值的范围，因此规则1被报告为无效

- 'Marital状态' 子句将其允许值定义为 'S, M' 的枚举；规则 12 指定值 'D'，因此该规则也被报告为无效。这些问题可以通过更新“允许值”或修改无效规则的输入条目来纠正，具体取决于实际的业务规则。

完整性检测 - 报告规则中的空白

决策表规则的差距意味着，给定输入值的组合，没有规则是匹配的。这表明可能缺少某些逻辑或规则（除非定义了默认输出）。

当决策表包含许多指定数量范围的规则时，很难直观地发现差距，并且编写和运行详尽的测试用例非常耗时。

例如：

U	Existing Customer	Application Risk Score	Credit Score	OutputClause
	null	null	null	null
1	false	< 120	< 590	HIGH
2	false	< 120	[590..610]	MEDIUM
3	false	< 120	> 610	LOW
4	false	[120..130]	< 600	HIGH
5	false	[120..130]	[600..625]	MEDIUM
6	false	[120..130]	> 625	LOW
7	false	> 130	-	VERY LOW
8	true	<= 100	< 580	HIGH
9	true	<= 100	(580..600]	MEDIUM
10	true	<= 100	> 600	LOW
11	true	> 100	< 590	HIGH
12	true	> 100	[590..615]	MEDIUM
13	true	> 100	> 615	LOW

Notes: DMN Expression

System Output

Running Post-bureau risk category table Validations ...
 Validating BusinessKnowledgeModel "Post-bureau risk category table".
 Warning: DecisionTable "Post-bureau risk category table" Completeness Violation: No rule exists for Existing Customer:"true", Application Risk Score:"<= 100", Credit Score:"580"
 Post-bureau risk category table Results: (0) error(s), (1)warning(s)

验证报告规则中的差距。仔细检查发现规则 9 中有错误。输入条目 (580..600] 应该是 [580..600]。

Unique命中策略的规则重叠检测

当规则重叠时，对于给定的输入值组合，匹配多个规则。如果决策表将其命中策略指定为“唯一”，则属于违规行为。

当决策表包含许多指定数量范围的规则时，很难直观地发现差距，并且编写和运行详尽的测试用例非常耗时。

例如：

DMN Expression

Post-bureau risk category table Input Parameter Values for Simulation

(Existing Customer = null, Application Risk Score = null, Credit Score = null)

	Existing Customer	Application Risk Score	Credit Score	OutputClause
U	null	null	null	null
1	false	< 120	< 590	HIGH
2	false	< 120	[590..610]	MEDIUM
3	false	< 120	> 610	LOW
4	false	[120..130]	<610	HIGH
5	false	[120..130]	[600..625]	MEDIUM
6	false	[120..130]	> 625	LOW
7	false	> 130	-	VERY LOW
8	true	<= 100	< 580	HIGH
9	true	<= 100	[580..600]	MEDIUM
10	true	<= 100	> 600	LOW
11	true	> 100	< 590	HIGH
12	true	> 100	[580..610]	MEDIUM

Notes DMN Expression

System Output

System Script DMN Validation Help System

Running Post-bureau risk category table Validations ...
 Validating BusinessKnowledgeModel 'Post-bureau risk category table' ...
 Warning : DecisionTable "Post-bureau risk category table" ConsistencyViolation: Rules "4, 5" overlap with input "false, [120..130], [600..610]"
 Post-bureau risk category table Results: (0) error(s), (1) warning(s)

验证报告规则中有重叠，涉及规则 4 和 5。仔细检查发现重叠存在于第三个输入“信用评分”中，其中“610”“[600..625]”重叠。您可以通过将规则 4 更改为“<600”或将规则 5 更改为 “[610..625]”来纠正此问题，以反映实际的业务规则。

集成到UML类元素

创建并模拟决策模型后，您可以生成Java、JavaScript、C++或C#的DMN模块并对其进行测试。

DMN Module可以与UML类元素集成，因此从类元素生成的代码可以重用DMN Module并且结构良好。由于类元素是一个状态机，在与DMN Module集成后，可以可执行状态机模拟通用地使用DMN Module的强大功能。

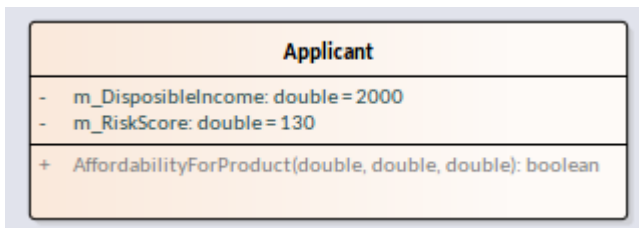
在本主题中，我们将解释将DMN模型与UML类元素集成的过程，考虑到：

- 类元素的要求
- DMN 模型
- DMN捆绑到类&智能感知
- 类元素上的代码生成

类元素的需求

假设我们有一个类*Applicant*，其操作*AffordabilityForProduct*评估申请人是否能够负担得起贷款产品。

A简化的模型类似于：



类*Applicant*包含两个属性，实际上是根据申请人的月收入、支出、现有还款、年龄和就业状态等更基础的数据计算得出的。

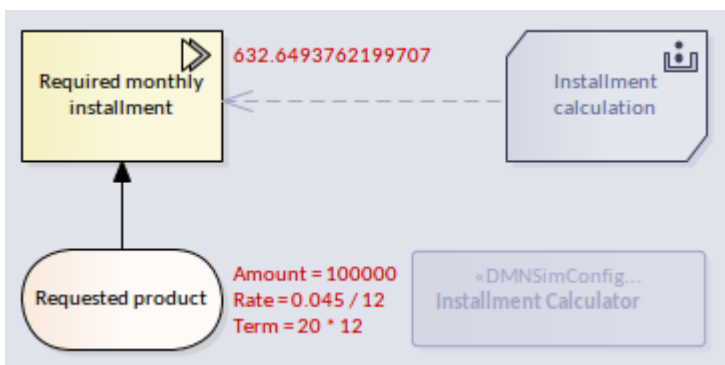
然而，在本例中，我们通过跳过这些步骤并直接提供可支配收入和风险评估来简化模型。

DMN 模型

在这个例子中，我们有两个不相交的DMN模型来展示一个UML类可以集成多个DMN模型。

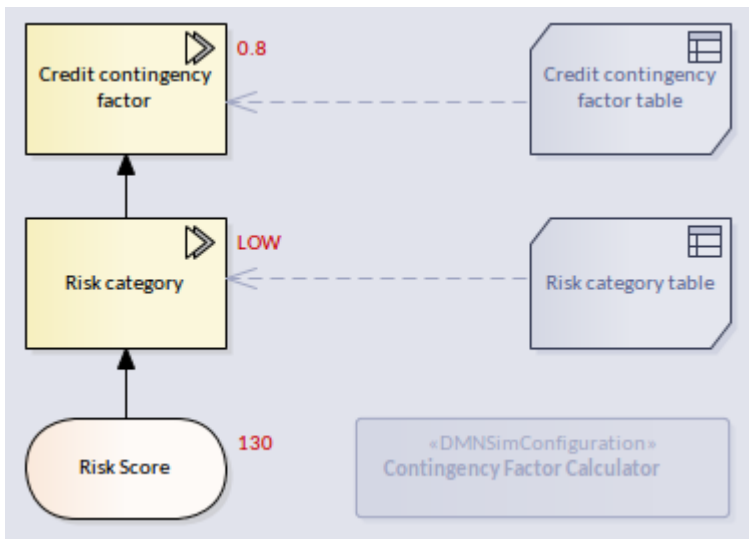
分期付款计算器

此DMN模型根据金额、费率和条款计算每月还款额。它由输入数据、决策和决策业务知识模型组成。

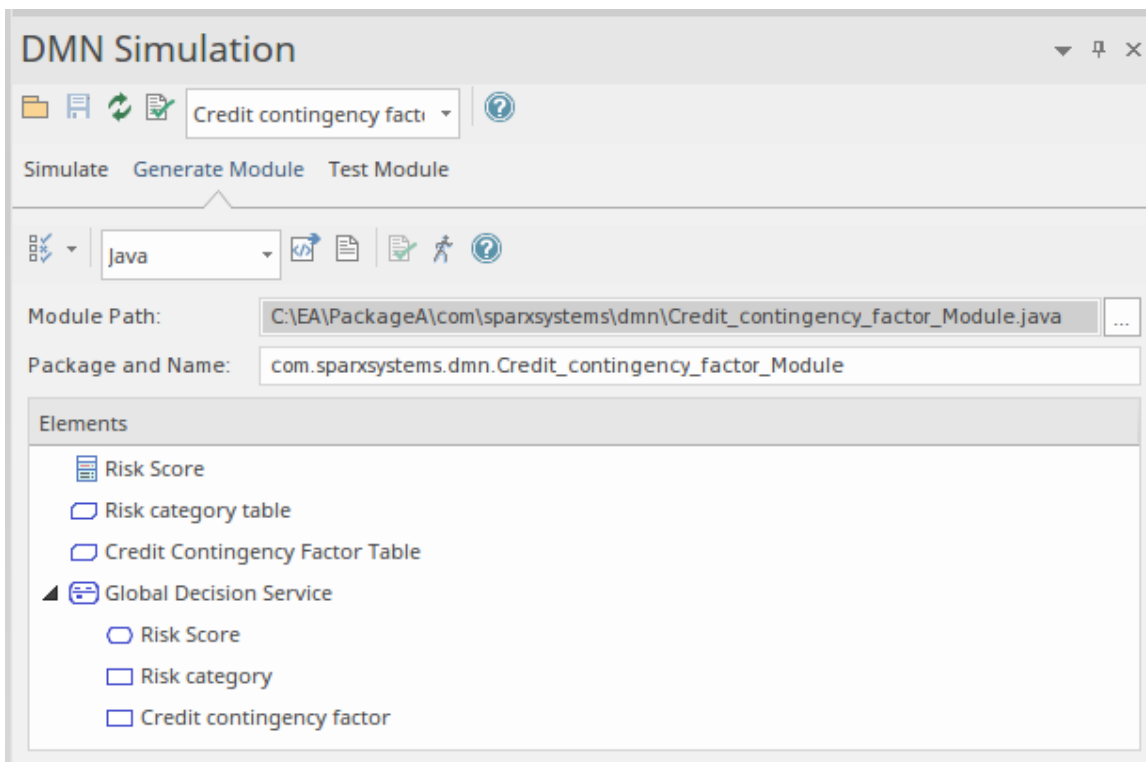


信用应急因子计算器

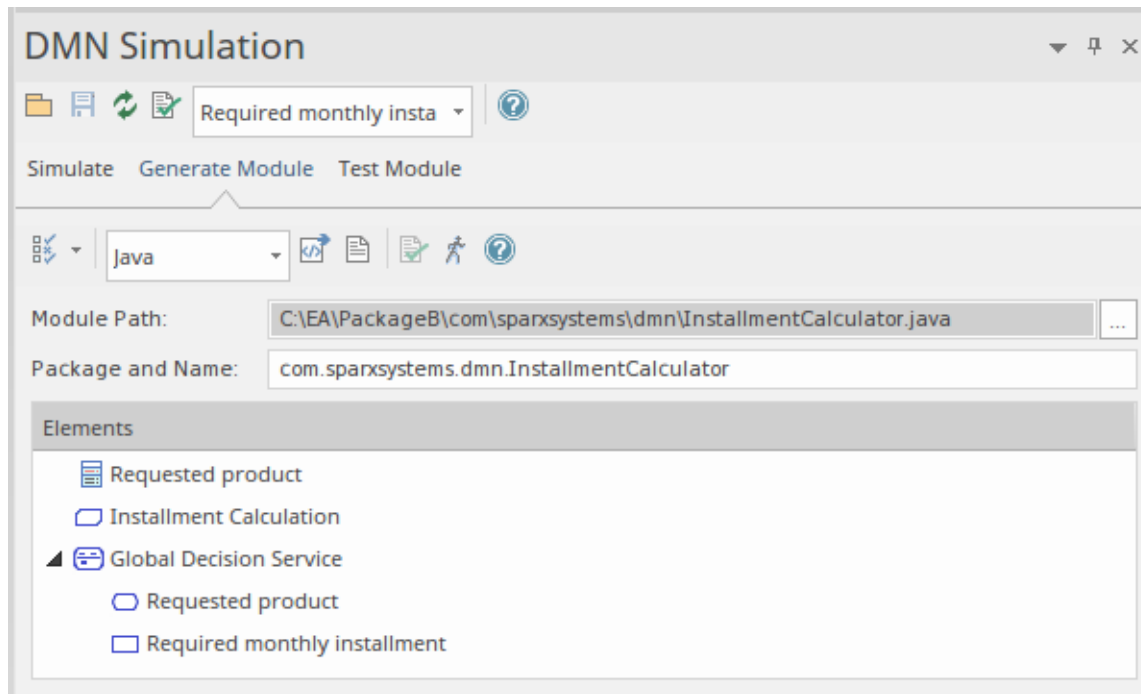
该DMN模型根据申请人的风险评估计算信用或有因素。它由一个InputData、两个Decision和两个业务模型组成。



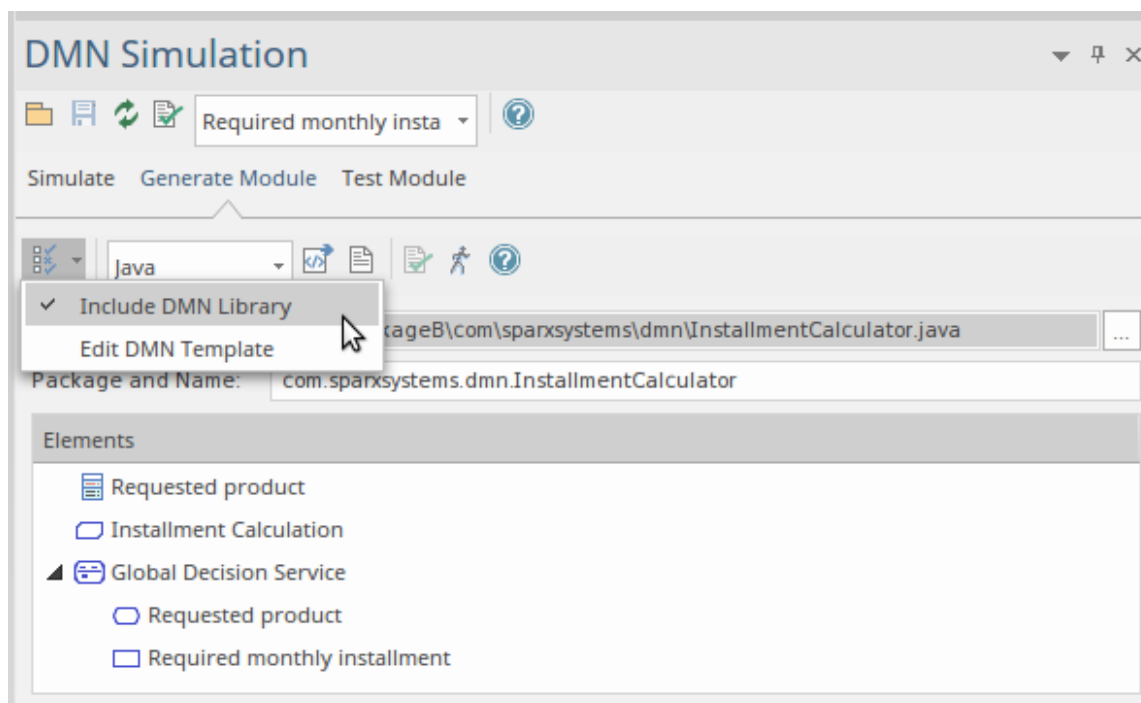
注记：在本例中，我们聚焦关注如何将 DMN 模块集成到类元素中；DMN 元素的细节在此不作描述。为两个 DMN 模型生成代码



单击生成代码图标，并检查您是否可以在系统输出窗口的“DMN”选项卡中看到此string：DMN 模块已成功编译。



注记：由于本模型使用内置函数PMT，因此必须包含 DMN 库：

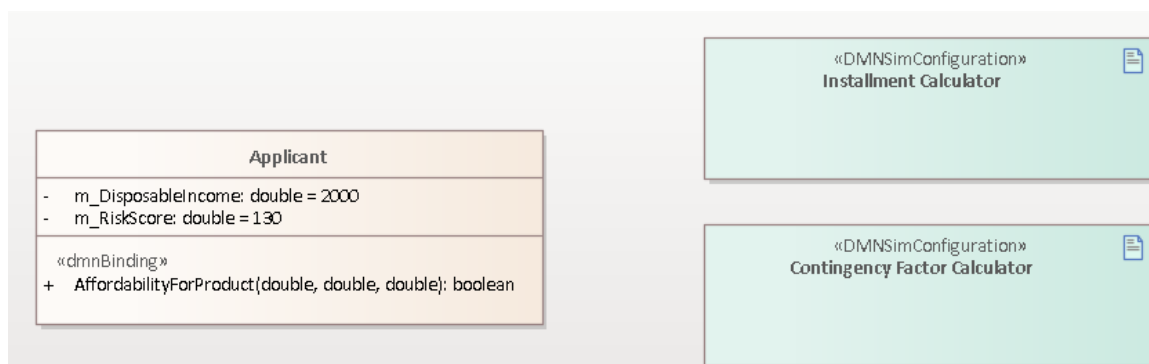


单击生成代码图标，并检查您是否可以在系统输出窗口的“DMN”页面中看到此string：

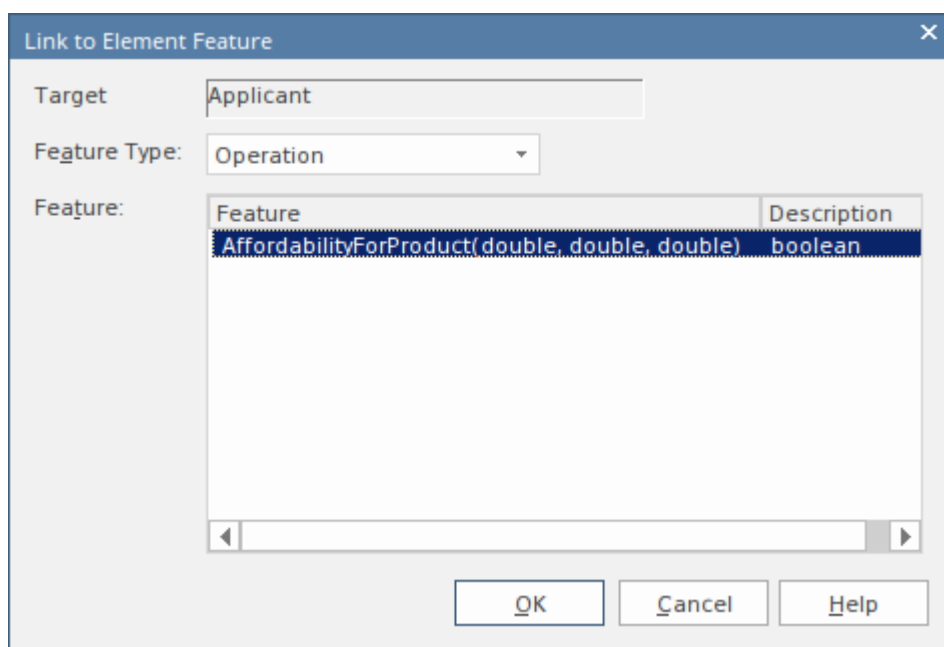
DMN 模块已成功编译。

DMN 捆绑到类&智能感知

把两个工件放在类图上。



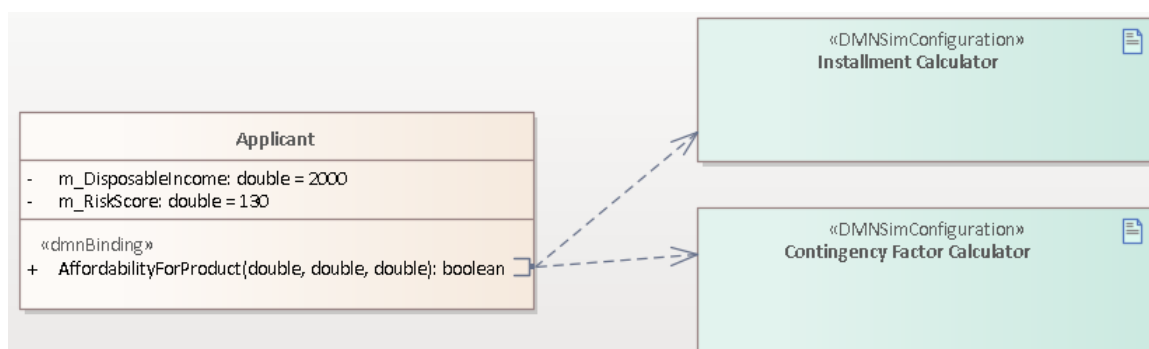
使用快速链接器创建一个从类应用程序到每个工件依赖关系的快速链接器。
 在创建连接时，会出现一个对话框，提示您选择要绑定到 DMN 模块的操作。



DMN模块绑定操作时：

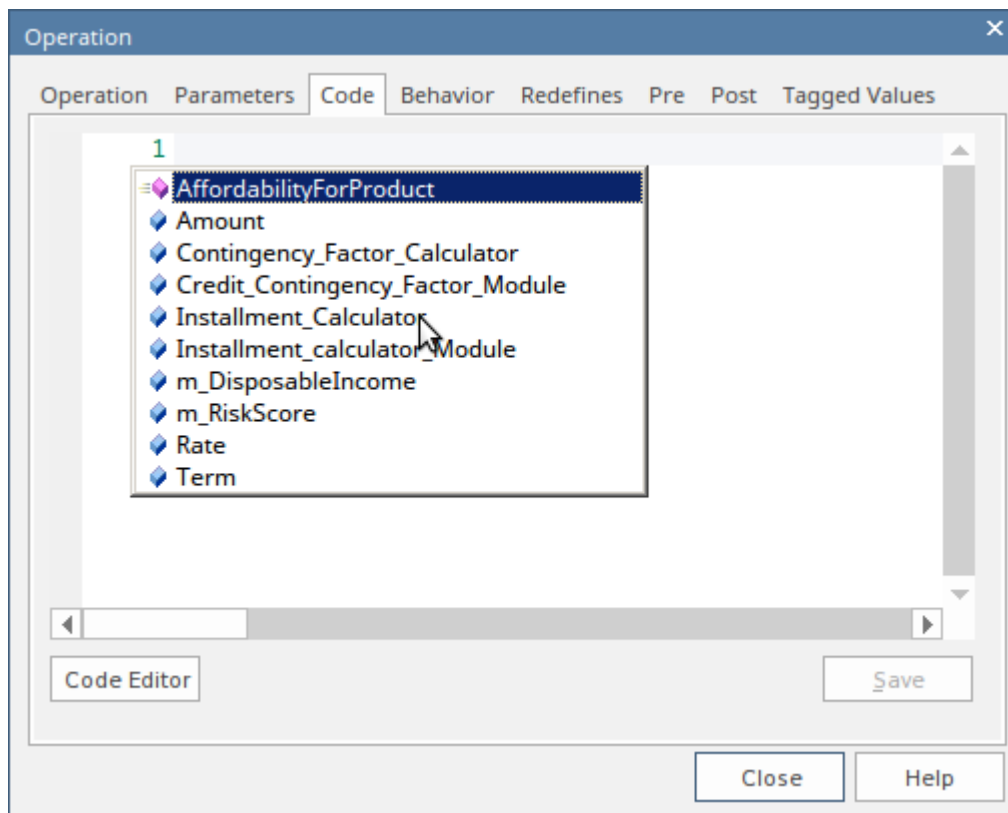
- 该操作采用构造型 <<dmnBinding>>
- 依赖连接器链接到操作

多个工件操作可以绑定到同一个操作。



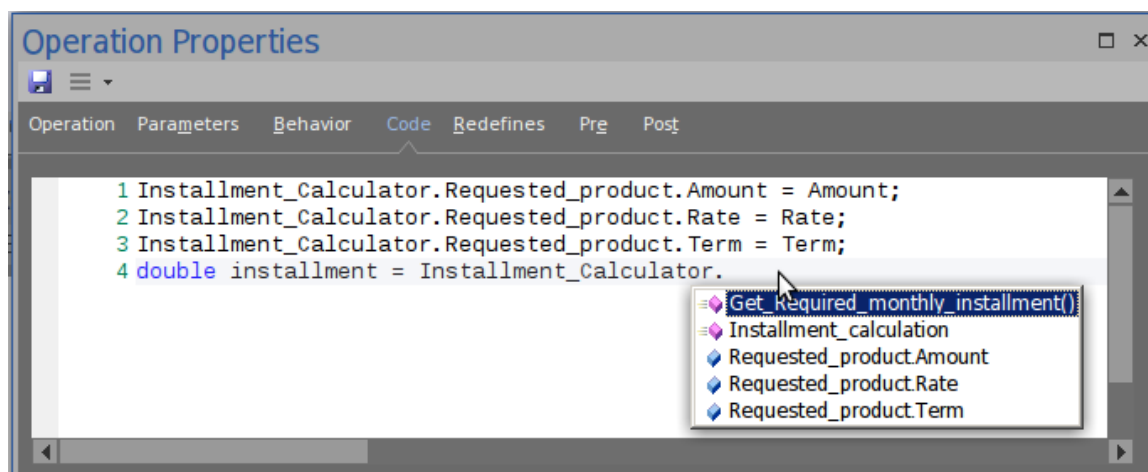
在 DMN Bindings 之后，智能感知操作的代码编辑器将支持 DMN Modules。要触发智能感知，请使用以下组合键：

- Ctrl+空格 - 在大多数情况下
- Ctrl+Shift+Space - 当 Ctrl+Space 在括号 '(' 之后不起作用时；例如，函数的参数，或在 'If' 条件的括号内

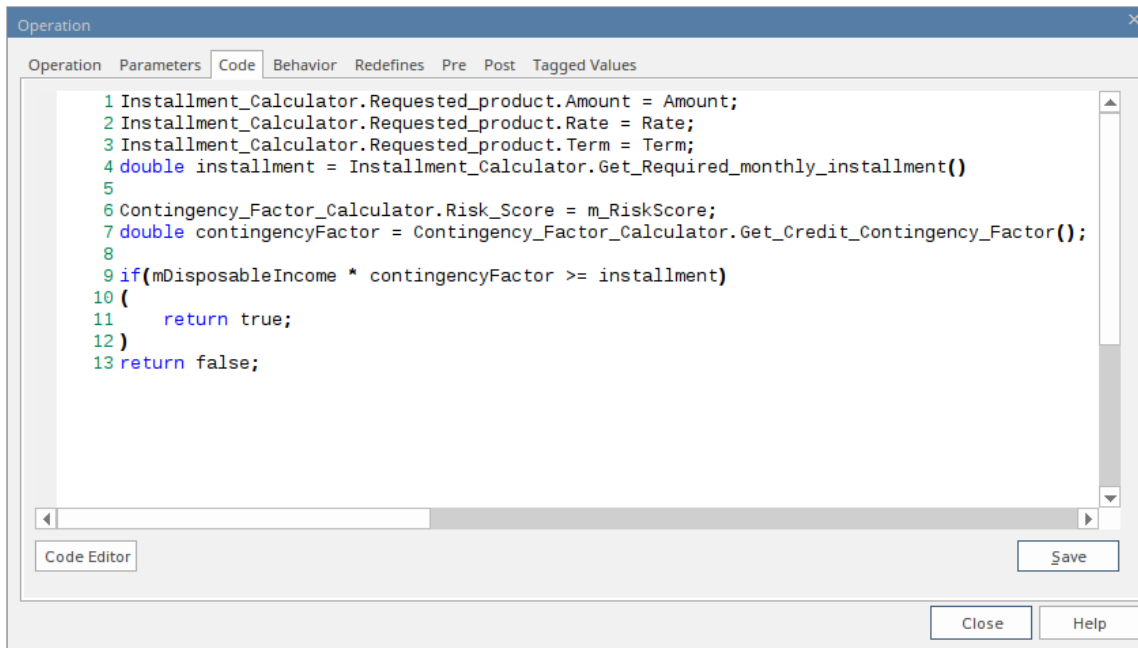


- 将列出类属性 - m_RiskScore、m_DisposableIncome
- 将列出操作参数 - 金额、费率、团队
- 操作将被列出 - AffordabilityForProduct
- 将列出所有绑定的 DMN 模块 - Contingency_Factor_Calculator、Installment_Calculator

使用智能感知支持编写代码非常容易。在访问 DMN 模块时，所有的输入数据、决策和业务模型将被列出以供选择。



此插图显示我们正在从 Installment_Calculator 中选择 “Get_Required_monthly_installment()”。这是操作的最终实现。



类的代码生成（使用 DMN 集成）

'生成代码生成' 生成类代码：

```
8 public class Applicant {
9
10     private double m_DisposableIncome = 2000;
11     private double m_RiskScore = 130;
12
13     PackageA.ContingencyFactorCalculator Contingency_Factor_Calculator = new PackageA.ContingencyFactorCalculator();
14     PackageB.InstallmentCalculator Installment_Calculator = new PackageB.InstallmentCalculator();
15
16     public boolean AffordabilityForProduct(double Amount, double Rate, double Term){
17         //WARNING: Code in this function will be overwritten when generate from EA because this operation has a flush type of stereotype
18         Installment_Calculator.Requested_product.Amount = Amount;
19         Installment_Calculator.Requested_product.Rate = Rate;
20         Installment_Calculator.Requested_product.Term = Term;
21         double installment = Installment_Calculator.Get_Required_monthly_installment();
22
23         Contingency_Factor_Calculator.Risk_Score = m_RiskScore;
24         double contingencyFactor = Contingency_Factor_Calculator.Get_Credit_contingency_factor();
25
26         if(m_DisposableIncome * contingencyFactor >= installment) {
27             return true;
28         }
29         return false;
30     }
31 } //end Applicant
```

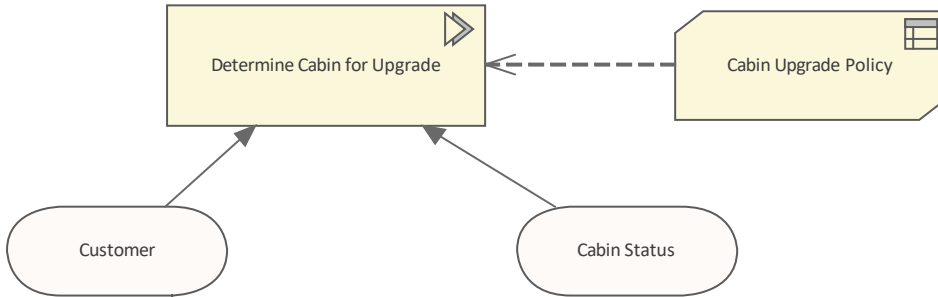
- DMN 模块作为类的属性生成
- dmnBinding 操作的代码已更新

注记：无论生成选项是 'Overwrite' 还是 'Synchronize'，如果操作的代码具有构造型 'dmnBinding'，都会更新。

示例图表

想象一下，您是一名航空公司预订人员，在一家繁忙的国内航空公司的值机柜台工作。让飞机准时起飞至关重要，因为延误可能会导致机场管制员收取费用，需要在较低的高度飞行，从而增加燃料成本以及其他处罚。

您的屏幕上会出现来自主管A消息，说明经济舱已超额预订；您需要将部分乘客升舱至业务或类——但应该选择哪些乘客，升舱至哪个舱位？A做出决定，但应该考虑哪些因素？这可以使用决策需求图记录在决策模型中。



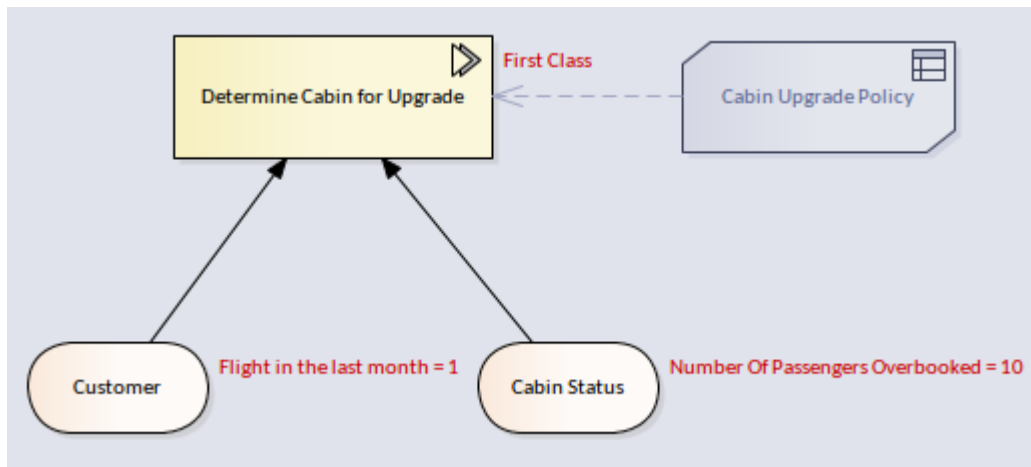
这很有帮助，但忙碌的值机员仍然需要权衡所有因素并做出公正的决定。心怀不满的乘客是否应优先于金级常旅客，或者是否应优先考虑特定乘客连接国际航班的事实。这些“规则”都可以记录在决策表中，明确哪些乘客应该升舱，哪些舱位：业务或类。这将使决策变得更加容易，并且可以在总部制定、商定和检查规则的一致性。在这个例子中，我们保持简单，并使用了两个因素：首先是乘客上个月的航班数量，其次是客舱超额预订的程度。

Cabin Upgrade Policy				
Input Parameter Values for Simulation				
(Flights in the last month, Number of Pax Overbooked)				
U	Flights in the last month	Number of Pax Overbooked	Upgrade Cabin	Annotation
			<i>Business Class, First Class</i>	
1	<=1	<=2	<i>Business Class</i>	
2	<=1	(2..8]	<i>Business Class</i>	
3	<=1	>8	<i>First Class</i>	<i>Start Filling First Class when heavily overbooked</i>
4	(1..5]	<=2	<i>Business Class</i>	
5	(1..5]	(2..8]	<i>Business Class</i>	
6	(1..5]	>8	<i>First Class</i>	
7	>5	<=2	<i>Business Class</i>	
8	>5	(2..8]	<i>Business Class</i>	
9	>5	>8	<i>First Class</i>	<i>Reward Frequent Flyers</i>

该表分为列和行。共有三种类型的列：做出决策所需的输入、应用规则的结果和注释的输出。

这再次非常有帮助，但仍然需要忙碌的源人员能够获取在决策表中找到正确行所需的所有必需信息。即使所有这些信息都可用，错误的决定仍然可能是由于人为错误选择表中的错误行而导致的。

幸运的是，决策模型可以自动生成并生成可以由应用程序执行的编程代码。所以我们忙碌的值机员不需要做任何事情或做出任何决定；当他或她正在为乘客办理登机手续时，如果特定乘客有权升级，它将在计算机屏幕上可见。在下图中，模型已被模拟，因此业务和技术人员可以同意模型已正确定义。在生成将在运行系统中运行并将结果显示给最终用户的编程代码之前，可以使用任意数量的用户定义数据集来测试模型。



在开发模型时，业务或技术用户可以逐步完成模拟，系统将向用户显示决策决策表中的哪一行被触发以确定输出。这在由多个决策组成的模型中非常有用。

Cabin Upgrade Policy		Input Parameter Values for Simulation		
(Flights in the last month = 1, Number of Pax Overbooked = 10)				
U	Flights in the last month	Number of Pax Overbooked	Upgrade Cabin	Annotation
	1	10	First Class	
1	<=1	<=2	Business Class	
2	<=1	(2..8]	Business Class	
3	<=1	>8	First Class	Start Filling First Class when heavily overboo...
4	(1..5]	<=2	Business Class	
5	(1..5]	(2..8]	Business Class	
6	(1..5]	>8	First Class	
7	>5	<=2	Business Class	
8	>5	(2..8]	Business Class	
9	>5	>8	First Class	Reward Frequent Flyers

管理升级决策的规则经常发生变化。例如，营销部门可能决定要奖励乘坐长途航班的乘客。可以更改决策需求图以包括新的输入、修改的决策表和重新生成的编程代码。一旦更改被推送到机场系统，合适的乘客将自动升级。签到人员仍然可以在培训和简报会议期间查看决策表以了解规则。

代码生成和测试模块

创建并模拟决策模型后，您可以生成Java、JavaScript、C++或C#的DMN模块。该DMN模块可以与Enterprise Architect BPSim执行引擎、可执行状态机或您自己的项目一起使用。

Enterprise Architect还提供了一个“测试模块”页面，这是一个将DMN与BPMN集成的预处理。其概念是提供一个或多个决策::DataObject元素，然后测试是否可以正确评估指定的目标决策。

如果出现任何错误或异常，您可以创建分析器来调试脚本模块和测试客户端的代码。

在这个“测试模块”过程之后，Enterprise Architect保证BPMN2.0::DataObject元素可以很好地与DMN模块一起工作。

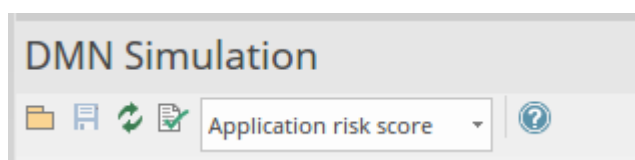
然后，您通过加载数据对象并将DMN模块决策分配给BPSim属性来配置BPSim，这将进一步用作从网关传出的序列流的条件。

访问

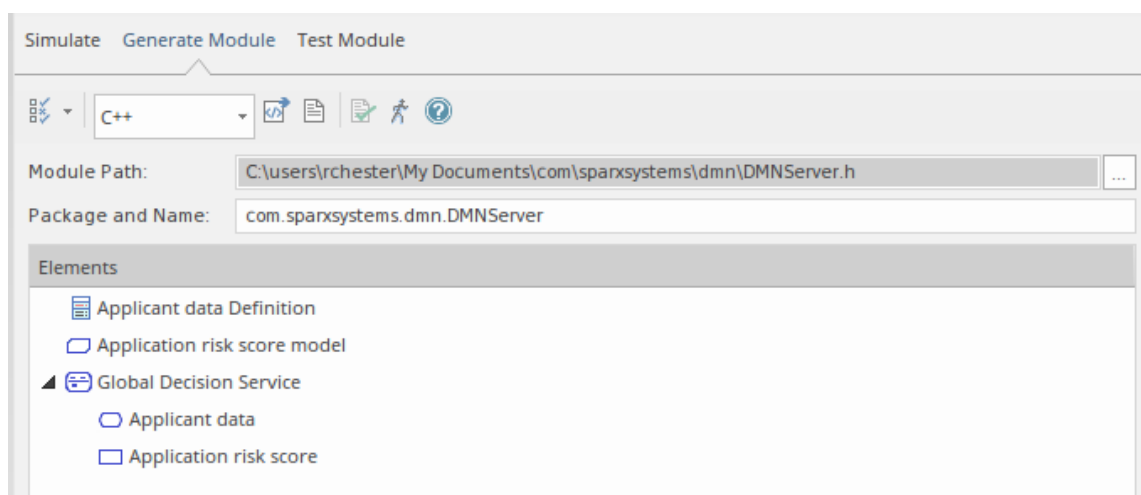
功能区	仿真>决策分析>DMN>Open DMN仿真>生成模块
-----	-----------------------------


DMN 模块：代码生成


在DMN仿真窗口上，在工具栏的数据输入字段中选择要从中生成模块的DMN结构。



单击“生成模块”选项卡，然后按住Ctrl并单击要生成到服务器的DMN元素的名称。





在选项卡工具栏的数据输入字段中，选择要生成的语言，然后在“模块路径”字段中单击图标并浏览到生成模块的路径位置（注记，对于Java，路径有以匹配包结构）。

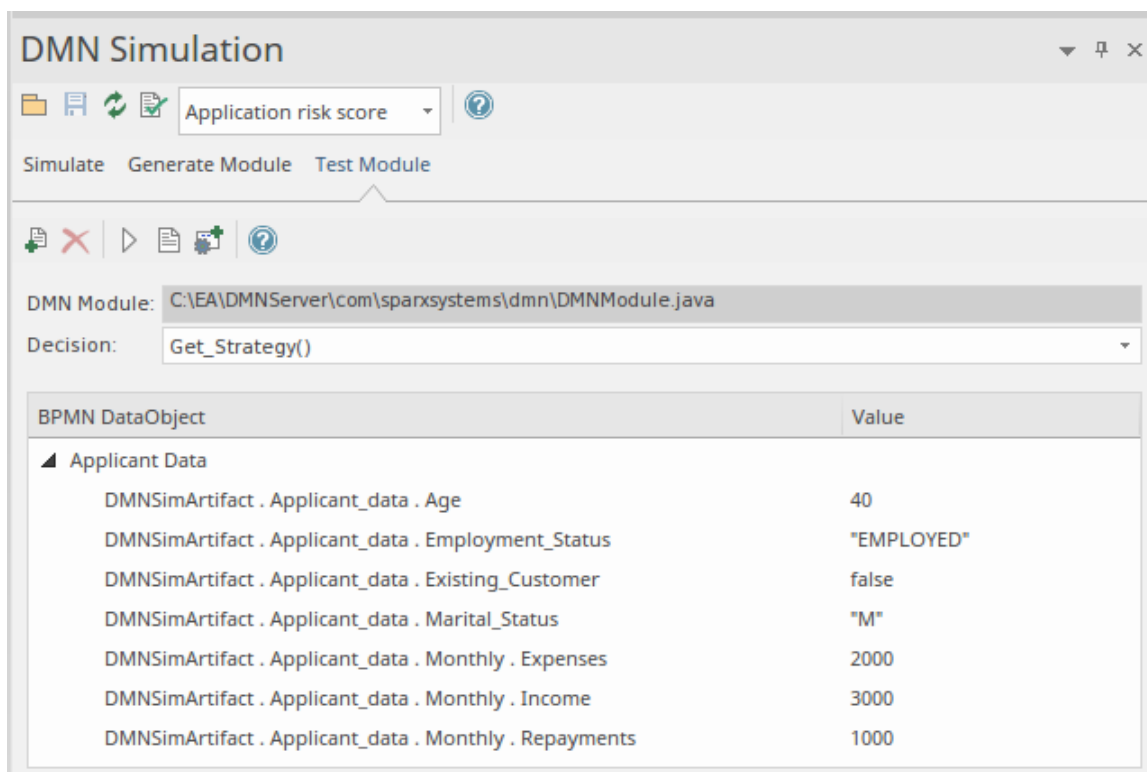
单击生成按钮生成。

生成完成后，单击完成按钮打开模块的“测试模块”选项卡。

DMN服务器：测试模块

当您使用  按钮选择“测试模块”选项卡时，“DMN 模块”字段将自动填充您最近在“生成模块”选项卡上生成的模块的生成 DMN服务器路径。如有必要，在“决策”字段中单击下拉箭头并选择所需的决策。

单击工具栏中的添加 BPMN 数据对象按钮 () 并选择一个或多个 (Ctrl+单击) BPMN2.0 数据对象以添加到主面板的列表中。



BPMN DataObject	Value
▲ Applicant Data	
DMNSimArtifact . Applicant_data . Age	40
DMNSimArtifact . Applicant_data . Employment_Status	"EMPLOYED"
DMNSimArtifact . Applicant_data . Existing_Customer	false
DMNSimArtifact . Applicant_data . Marital_Status	"M"
DMNSimArtifact . Applicant_data . Monthly . Expenses	2000
DMNSimArtifact . Applicant_data . Monthly . Income	3000
DMNSimArtifact . Applicant_data . Monthly . Repayments	1000


现在单击工具栏上的运行按钮。在系统输出窗口中，此消息表明 DMN服务器和 BPMN2.0 DataObject 可以很好地相互配合以评估所选决策：

为 DMN服务器运行测试客户端...

```
dmnServer.Application_risk_score : 133.0
```

结果：133.0

运行成功完成。

如果有错误，通过单击分析器工具栏按钮创建  脚本并使用该脚本修复问题。

重要提示：建议在将 DMNServer.java 与 Enterprise Architect BPSim 执行引擎集成之前执行此“测试模块”步骤。请参阅将 DMN 模块集成到仿真以获取仿真帮助帮助。

代码生成和连接到 BPMN

- 生成 Java、JavaScript、C++ 或 C# 中的 DMN服务器
- Java版运行服务器的运行/调试测试
- 将 DMN服务器连接到 Enterprise Architect BPSim 执行引擎

公共错误和解决方案

- 变量类型：由于 DMN 模型使用 FEEL 语言（用 JavaScript 仿真），变量类型不是强制性的；但是，当为编译的语言生成代码时，您必须键入一个变量 - 有上下文菜单选项和标记值用于设置变量的类型
- 由于 DMN 表达式允许使用空格，因此为了阐明复合输入数据，“.”之前和之 必须有一个空格。在表达式中；例如，“申请人数据。年龄”有效，而“申请人数据。年龄”无效
注记使用自动完成特征时不会出现此问题
- 运行验证将帮助您定位大部分建模问题；在模拟和代码生成之前执行此操作

注记

- 用 Java 编译需要对目标目录有完全的读写权限；如果模块路径设置为“C:”或“C:\Program Files (x86)”，编译将失败

