



ENTERPRISE ARCHITECT

用户指南系列

决策建模指南

Author: Sparx Systems

Date: 20/06/2023

Version: 16.1

创建于  **ENTERPRISE
ARCHITECT**

目录

决策建模指南	4
建模决策	5
开始	8
设置模型结构	10
定制应用程序	11
创建图表元素和关系	15
窗口和工具	20
决策模型和符号概述	24
什么是决策模型和表示法	25
A一个示例	27
用途等级	29
为什么要使用决策模型和符号	30
何时使用决策模型和符号	31
示例决策模型	34
决策模型和符号的好处	36
可视化决策和规则	37
促进协作	38
简化业务流程图表	39
交流共同的理解	40
发现自动化机会	41
并入架构模型	42
为决策创建单一真理源	43
仿真决策模型	44
测试决策模型	45
生成编程代码	46
选择一致性和正确性	47
发现规格差距	48
与其他模型一起表达	49
可视化输入和分析	50
符号简介	51
决策需求图表	54
表达式决策表达式类型	62
决策表	63
文字表达	64
盒装上下文	65
调用	66
表达语言	68
表达式编辑器	69
足够友好表达语言(FEEL)	71
决策表解释	74
库表方向	76
允许的值字段	77
输入输出子句的数据类型	78
规则和输入输出	79
命中策略	80
合并和取消合并单元格	83
验证决策模型	85

规则间隙	86
重叠规则	88
决策模型和符号的上下文	89
仿真决策模型	96
搭建仿真	97
运行仿真	100
仿真决策服务	104
从决策模型生成代码	106

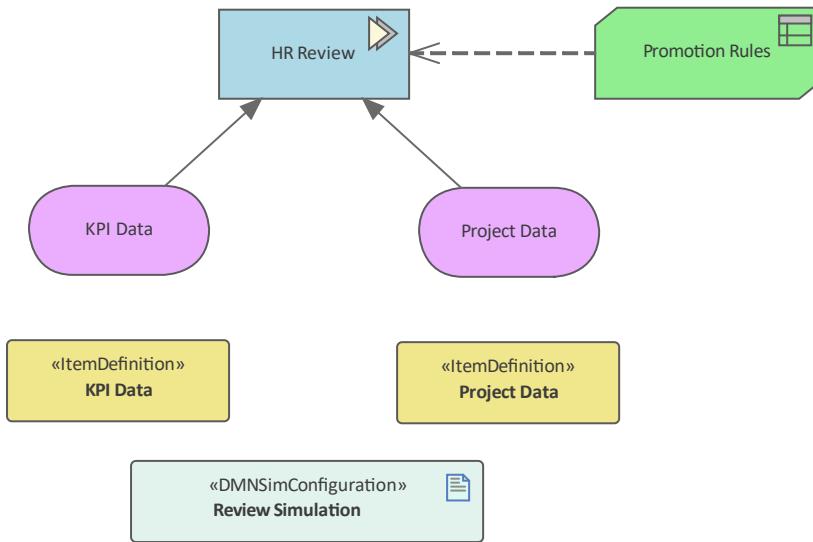
决策建模指南

现在都会发布一个新标准，该标准注定要从根本上改变我们的工作方式以及支撑我们生活几乎方方面面的业务和技术系统的质量。决策模型和表示法就是这样一个标准，它在世界领先的企业建模工具中的实施为以软件为中心的企业、业务、技术、工程和科学系统的交付（无论大小）的复兴铺平了道路。

该标准有助于消除业务和技术人员之间由来已久的沟通裂缝，并允许企业在战略或战术层面定义和管理影响组织及其客户和供应商成果的决策。它是一个适用于所有季节和学科的工具，从定义问题的策略师和他们希望系统运行的一组答案开始，到只需按下按钮即可实施决策的 Web 服务设计师。价值主张是，决策可以用图表和简单的规则库表建模，并用简单的表达方式，例如“顾客的每月可支配收入是否 > 2000 美元”。一旦定义了规则，企业就可以使用自己的数据对其进行测试，而无需技术社区的任何参与。如果业务人员可以创建一个简单的电子表格，他们会发现 DMN 非常易于使用，并且能够创建和验证他们自己的决策模型。平台内置的通信特征有助于与其他业务和技术利益相关者进行即时讨论。

建模决策

现在，在一个被创新的业务和技术理念以及颠覆性的工作方式所颠覆的世界中，组织比以往任何时候都更需要清楚地了解其选择和做出的决策。不受管理的复杂性是敌人，它的反面——敏捷性——是预示业务成功并使组织能够快速响应其业务环境变化的朋友。如果没有清晰且可交流的模型，组织几乎不可能接受数字世界中每天面临的变化。决策的描述和实施，莫名其妙地成为几乎所有其他学科的一部分，现在已经被综合成一门严格而正式的学科，以一种新的方式来建模和描述决策、输入、结果、规则、业务、权限等。确实，一旦您看到了决策模型和符号的实际应用，并了解了它带来的无数好处，您将无法回到旧的和晦涩难懂的工作方式。



Enterprise Architect因其灵活、可扩展、基于标准和实用的复杂系统建模方法而成为许多业务和技术领导者的首选工具。作为一个协作平台，它是所有学科的工具，允许创建、集成、管理、记录、模拟和生成决策模型以编程代码。模型与其他模型集成集成，包括业务流程图表、用例用户案例、测试案例模型、案例执行工件、用户数据库和编程代码可视化列出主要模型。

它将如何帮助您

大多数读者通常对组织中的决策有一定的了解，但每个读者很可能有不同的经历和不同的定义、管理和处理决策的方式。决策出现在组织描述和实施的整个结构中。将决策提炼成一个单独但清晰的模型将提供极大的清晰度和价值。读者将受益于了解企业架构师的特征以及可用于开发和管理决策模型的工具，这反过来又将使他们作为个人和团队成员更有效率。然而，最终价值将在于他们工作的组织，这将获得有效和敏捷地响应变化的能力，使他们能够在数字时代复杂且不断变化的商业环境中导航。

谁将受益

任何参与决策制定、管理或实施的人，无论是在战略层面、商业价值层面还是技术层面，都将从阅读本指南中受益。这包括广泛的角色，其工作和决策最终将由模型指导和促进，但具体而言，使用决策模型和 Notation 对四个群体具有重要价值：

- **战略思想家**负责引导企业渡过汹涌的变革浪潮、设定目标并了解驱动因素，他们将受益于能够可视化他们做出的决策并知道他们正在根据自己的意图实施决策。
- 目前将业务规则和决策以复杂的级联网集的形式构建到进程中的**业务和进程分析**人员将受益，因为他们将能够提取这些规则，将它们放入一个清晰的决策模型中。其结果将是复杂性的降低、更易于理解且对变化更具弹性的进程模拟模型，以及可以进行、测试和实施的简单决策。
- 完全了解将业务规则与代码主体隔离并允许对其进行配置的好处的**软件工程师**将受益，因为他们不仅将拥有清晰且经过测试的规则模型，而且还能够自动生成包含规则的编程代码，消除解释或翻译错误的可能性。

- **系统工程师**，他们习惯于在从太空探索到海洋学的广泛学科和背景中处理复杂且通常看似棘手的问题。系统设计为在需要做出决策的时间范围内上下文，例如生产线机器人、行星漫游车、运输控制系统、机械厂的安全控制系统等等。这些系统通常严重依赖决策来维持运营效率、安全性并能够对环境中的发生变化的变量做出响应。

你会学到什么

本主题将教您如何使用Enterprise Architect的众多特征，使用新的Decision Model and Notation (DMN)决策和管理决策模型，将决策模型连接到其他类型的企业模型，例如业务流程、状态机、用例和参数，模拟它们，自动生成编程代码，创建文档并作为团队成员协作工作。

您将了解哪些工具可用，哪些工具应该用于特定技术，以及如何使用它们。例如，该主题将教您如何使用图表和决策表将复杂且看似难以处理的决策规则分解为简单易懂的模型。这些可以手动模拟和实现，或者使用Enterprise Architect中提供的工具和功能以各种语言生成高质量的编程代码。

指南概述

本指南分为多个主题，将从多个角度向您介绍决策建模，确保一旦您通读该文档，您将对决策建模的原因、内容和方式有充分的了解。该指南以一个开始主题开始，该主题将介绍具有广泛笔触的概念，然后是概述。接下来是一个好处列表和一系列开发细节的主题。概念、符号和工具用法均已介绍，为您提供理论和实践知识，帮助您开始使用自己的决策模型，并从将这种方法应用于建模决策中获益。

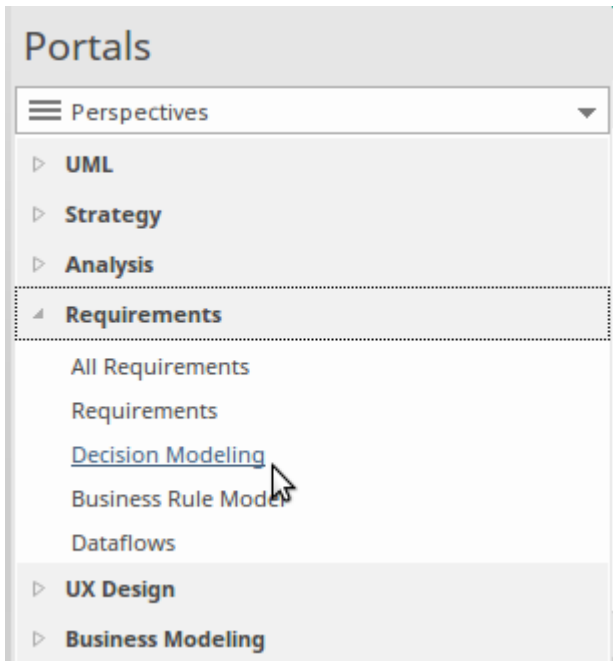
开始	开始主题为您提供了足够的信息来开始设置您自己的模型，从设置模型结构、定制应用程序、创建您的第一个图表以及学习如何使用Enterprise Architect中建模决策中使用的窗口和工具开始。
决策模型和符号概述	决策模型和符号概述主题介绍了 DMN 标准并提供了一个简单的示例。该主题继续介绍使用级别的概念，这将帮助您了解它在组织中的实际工作方式。该主题上下文提供其使用的时间，以及何时以及为什么应该使用它。
决策模型和符号的好处	决策模型和符号的好处主题几乎是使用这种方法的一个商业案例，并举例说明了为什么组织应该使用Enterprise Architect来进行模型决策。完成本主题后，您应该能够了解如何以及为什么可以使用它来帮助您的组织更正式、更严格地建模决策，以及采用这种方法可以获得的好处，包括验证和实现代码的生成。
决策模型和符号的上下文	决策模型和符号的上下文主题将帮助决策者、需求分析师和其他利益相关者了解可以使用建模模型的学科的情况。它介绍了商业、工程和科学学科的广泛背景。它讨论了一些典型示例，例如将决策模型与业务流程图一起使用，并探讨了Enterprise Architect支持的其他有趣的应用程序。
示例决策模型	示例决策模型主题介绍了一个完成示例，该示例足够简单易懂，但也具有足够的复杂性来演示标准的某些表达方面和Enterprise Architect中可用的功能。该示例将为后面的一些主题提供有用的背景，这些主题将介绍更多更丰富的语言和工具特征。
符号简介	符号简介主题向您介绍了决策模型和符号标准。这包括将放置在图表上的视觉元素，包括元素、关系和工件，以及它们的含义和用法。
决策需求图表	决策需求图表主题介绍了用于构建决策模型的主图，并教您如何使用Enterprise Architect中可用的许多工具来创建、修改和使用元素。您将在本主题中学习的许多内容也可以应用于其他图表类型，因此通过最后一个主题，您将很好地了解如何使用该工具。

表达式决策表表达式类型	决策决策类型主题是我们探索如何定义决策逻辑的地方。介绍了可用于描述逻辑的表达式类型，以及可用于管理这些表达式的表达式编辑器。
决策表解释	决策表解释主题深入到最常见的表达类型——决策决策决策表——的细节，并解释了命中策略、允许值、值和表达类型等。这将成为新手和经验丰富的决策建模者的重要参考。
验证决策模型	验证决策模型主题介绍了验证工具，可用于检查模型的一致性、正确性和完整性。这是Enterprise Architect的一个特征的内置特性，可防止由于表达式中的间隙和重叠而导致的错误。验证应作为模拟的先导，以确保模型是健全的、富有表现力的，并在逻辑上将商业决策背后的意图和推理编成代码。
仿真决策模型	仿真决策模型主题介绍了允许决策模型像在生产系统中就地执行一样执行的特征。Enterprise Architect允许建模人员（业务或技术人员）无需配置即可运行模拟。需求图中的任何决策都可以选择进行模拟，包括最高级别的决策。您可以选择任何预定义的输入数据集并运行仿真，以查看具有不同数据作为输入的模型输出。
从决策模型生成代码	来自决策模型的代码生成主题在Enterprise Architect中引入了一种生产力工具，它允许直接从模型中自动生成实现（编程）代码。该功能是在运行时引擎中成功实施规则的关键，因为程序员或技术人员无需解释模型或参与业务用户的会议——代码直接从模型生成。此功能将消除困扰行业的与商业意图误解相关的问题。工作流程很简单——指定决策、定义表达式、验证表达式并生成代码。

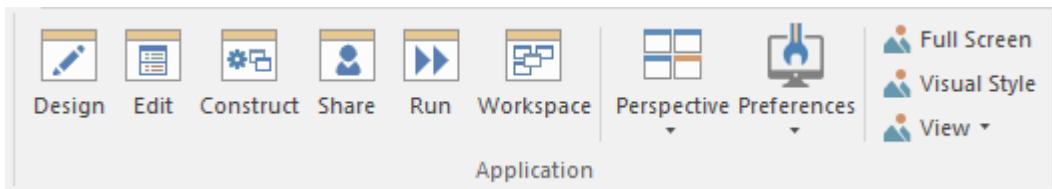
开始

入门通常是企业架构师面临的最困难挑战之一，是Enterprise Architect提供的一种最简单的辅助业务分析师或技术分析师，它为新人提供了许多功能。Enterprise Architect是一个大型且具有多种功能的应用程序，其覆盖范围的广度和深度可能会让刚接触该程序的人不知所措，但幸运的是，设计中已经内置了一个解决方案。

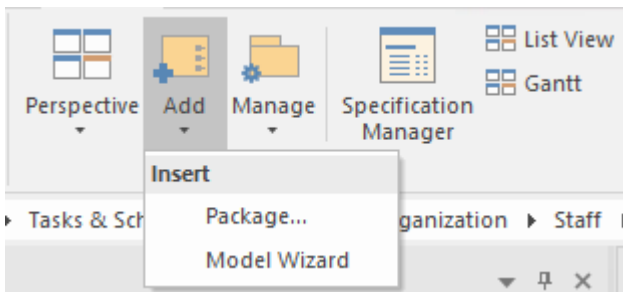
蓝图可用于将功能限制为分析员建模需求决策利益相关者更容易上手。您仍然可以使用其他可能有用的功能，例如策略建模、需求、业务流程建模、思维导图等，只需更改蓝图。所有这些都无需打开其他工具。值得注意的是，蓝图存在于Enterprise Architect支持的广泛建模学科中。更多信息请参见[Model Perspectives](#)帮助。



您还可以通过设置首选项和选择工作区和视觉样式来定制自己的环境和用户界面，具有极大的灵活性。有关更多信息，请参阅[Advanced Customization](#)帮助主题。



使用模型向导模式（附带文档）可以简单地设置新项目，该模型可用于自动创建 DMN 项目结构以帮助您入门。随着模型的开发以及问题和解决方案空间的充实，该向导可用于创建任意数量的决策需求图。



所有这些功能都让新手可以轻松上手，让您成为团队的高效成员，并迅速开始为模型做出贡献，而不会有任何延迟。如果您是新手需求分析师，与使用基于文本或其他更基本的建模工具相比，您会惊讶于自己的生产力。当您将自己和工具推向新的极限时，一路上会遇到挑战，但方便的帮助系统、庞大的用户社区、综合论坛、社区站点和一流的支持服务将使旅程变得轻松且内容丰富。此图显示了用于决策建模视角的内置模型之一，其中包含一系列可以自动注入到您自己的模型中的有用模式。每个模式都附带说明如何使用该模式并帮助确保您遵循最佳实践的随附文档。

Open Project Create from Pattern Add Diagram Guidance

Decision Modeling

- DMN Starter Patterns
 - Single Decision Table Unique Hit Policy
 - Single Decision Table Any Hit Policy
 - Single Decision Table with Business Knowledge...
 - Single Decision Table with Knowledge Source
 - Simple Decision and Business Process Diagram
 - Simple Decision and Information Model
 - Three Decision Two Level Hierarchy
 - Three Decisions with Literal Expression
 - Simple Decision Service
- DMN Business: Knowledge Model Examples
- DMN Decision Examples
- DMN Complete Example

Single Decision Table with Business Knowledge Model

The *Single Decision with Business Knowledge Model* pattern is useful for modeling reusable decisions, it can act as a starting point for more complex decision graphs. The *Invocation Expression Type* is utilized for the decision which does not do the work of deciding but invokes the Business Knowledge Model (BKM) to provide the result. The BKM publishes a set of parameters that are passed in from the decision and when the result is determined this is passed back to the Decision. It allows business architects, business analysts and other stakeholders to input and refine the rules that make up the decision. The Hit Policy of *Unique* means that only a single rule is allowed to match and there can be no overlapping rules, formally these are known as disjoint rules.

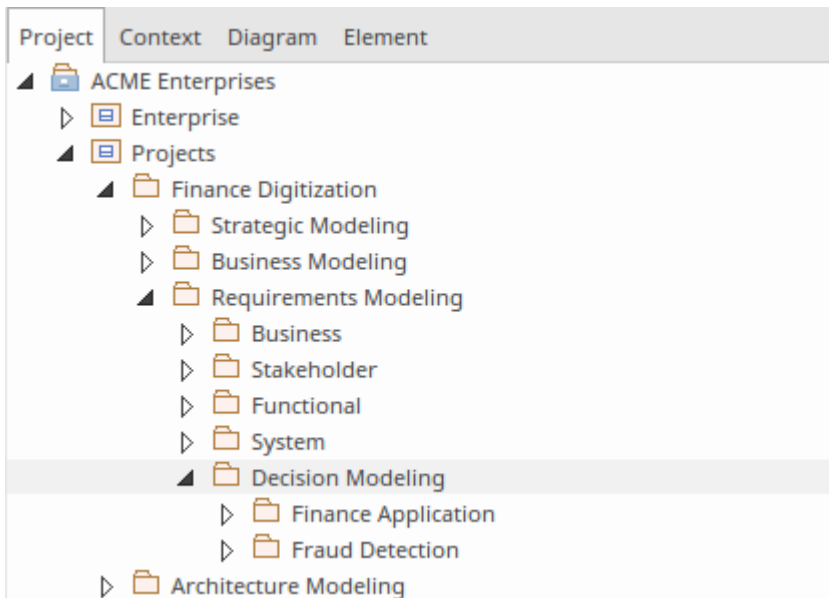
The diagram illustrates the relationship between a decision and a business knowledge model. On the left, a rectangular box labeled "Decision A" has a right-pointing arrow on its right side. On the right, a rounded rectangular box labeled "Business Knowledge Model A" has a left-pointing arrow on its left side. A dashed line connects the two arrows, indicating a bidirectional flow of information between the decision and the knowledge model.

Create Model(s) Add To: Help System Customize Pattern on import Combine with selected Package

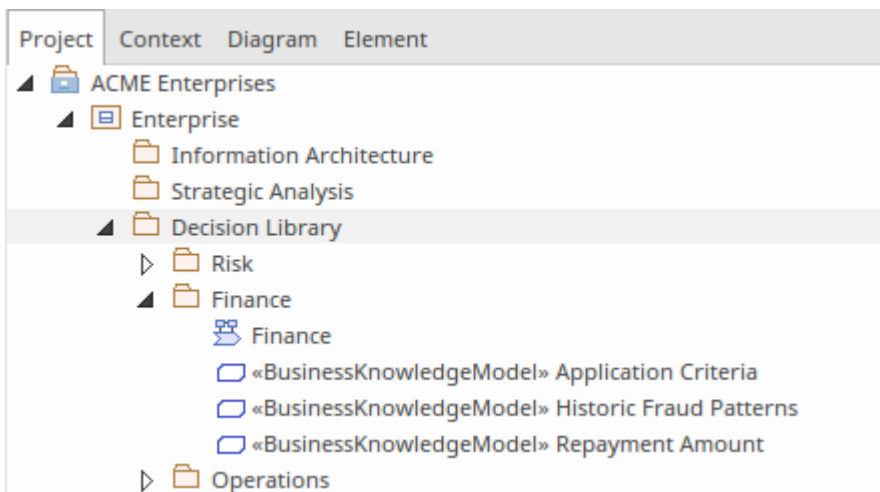
设置模型结构

Enterprise Architect从一开始就被设计为一种生产力工具，并包含一个内置模型向导，使设置模型结构变得非常简单。这些直观的工具为有经验的用户节省了宝贵的时间，并为初学者消除了困难。使用决策建模，您创建的元素通常会构成项目或计划的一部分，并与其他模型相关，例如由业务分析师、系统工程师、软件工程师和其他人创建的业务流程图和状态机。

此图显示了一个示例存储库结构，它指示了决策建模元素在项目的需求建模分支中的位置。决策模型可以位于项目中的任何位置，Enterprise Architect提供了一种功能，可以在浏览器窗口中将它们（或其包含的包）从一个位置拖放到另一个位置。



下图显示了被设计为可重用决策组件的业务模型如何包含在库中，并且可以被任何项目或计划重用。该库可以存储在企业模型级别，并且在需要时，可以通过将适当的业务模型从浏览器窗口拖到所选项目的决策需求图 (DRD) 上，将它们包含在相应的项目中。



业务模型通过允许将决策信息作为参数传入并将输出传递回调用决策来提供重用机制。

存储库的整体结构超出了本指南的范围，但它是设置存储库的一个关键方面，您的图书馆员或系统管理员将执行此任务或能够为您提供帮助。稍后我们将了解到，包是组织和维护模型库的重要单元。有关更多信息，请参阅[The Model Wizard](#)帮助。

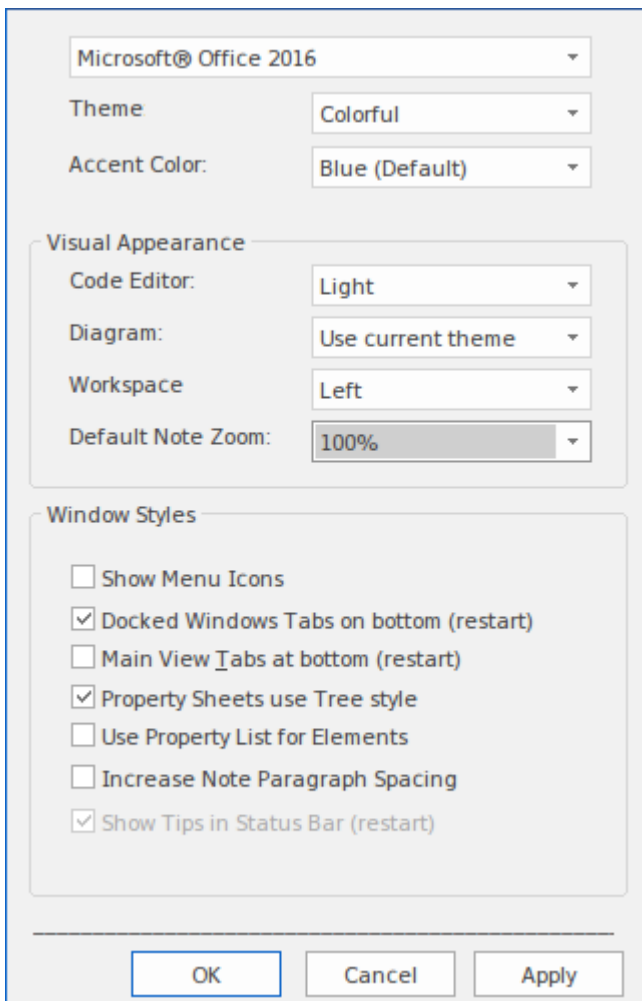
定制应用程序

Enterprise Architect是一种具有大量功能的工具，这也是它作为任何类型的系统建模工具如此受欢迎的原因之一。为了确保该工具为组织、团队、项目或个人提供最大的利益，对界面进行一些剪裁以适应建模意图将确保所有各方都能获得最佳结果。大多数设置都可以通过单击一个按钮来更改，从而将工具转换为适合目的，在我们的案例中，这是在需要对决策进行建模、表达和模拟的项目上进行协作。

在这些部分中，我们将研究一些可以将应用程序从通用建模工具更改为需求建模工具的聚焦，并特别关注决策模型的创建、管理和模拟。

选择视觉风格

每个建模者都会对用户界面的配色方案和样式有自己的偏好，Enterprise Architect允许为每个用户设置和保存这些偏好，从而使应用程序更具吸引力并符合个人用户的工作方式。例如，一些建模者会想要深色方案，而其他人会喜欢浅色或彩色方案。

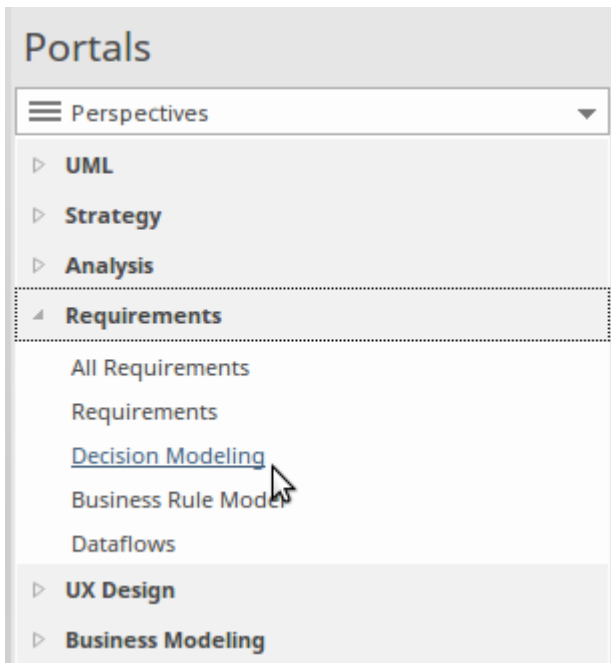


The screenshot shows the 'Visual Appearance' and 'Window Styles' sections of the Enterprise Architect settings dialog. The 'Visual Appearance' section includes dropdown menus for 'Theme' (Colorful), 'Accent Color' (Blue (Default)), 'Code Editor' (Light), 'Diagram' (Use current theme), 'Workspace' (Left), and 'Default Note Zoom' (100%). The 'Window Styles' section includes checkboxes for 'Show Menu Icons', 'Docked Windows Tabs on bottom (restart)', 'Main View Tabs at bottom (restart)', 'Property Sheets use Tree style', 'Use Property List for Elements', 'Increase Note Paragraph Spacing', and 'Show Tips in Status Bar (restart)'. At the bottom, there are 'OK', 'Cancel', and 'Apply' buttons.

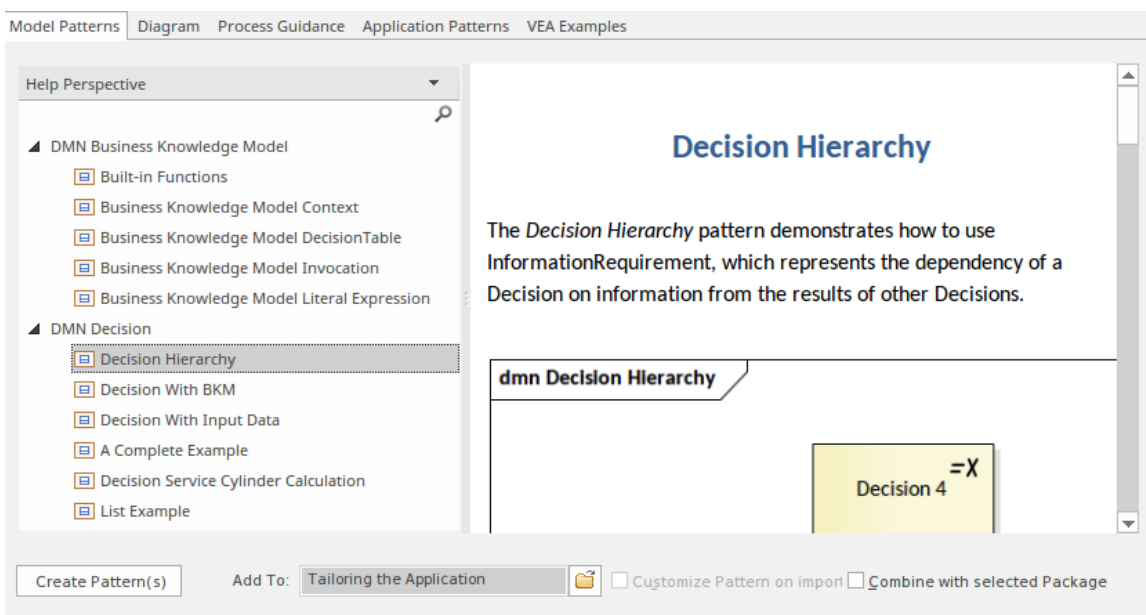
这里有一系列选项，包括设置主窗口选项卡的位置和注记窗口中的文本大小等等。设置视觉风格将有助于个性化建模环境，并使单个建模者感到舒适，同时保持一致和严格的模型。有关更多信息，请参阅[Visual Styles](#)帮助主题。

选择蓝图

Enterprise Architect是一个包含特征特性的工具，适用于广泛的学科、方法、语言和框架。蓝图为用户提供了一种选择工具方面的方法，使他们能够聚焦于工具特征和功能的特定子集。决策建模蓝图为专注于建模决策的用户提供了一个自然的起点，但在任何时候，如果您决定使用工具中的其他功能，您可以简单地更改蓝图，该工具将重塑自身以提供对选定区域的聚焦。



选择决策蓝图会将工具更改为聚焦到模型建模功能，例如，应用程序将显示一系列模型模式，通过能够加载标准模型片段或图表的模式，为用户提供快速启动。新图表对话框也将只显示决策需求图表类型。

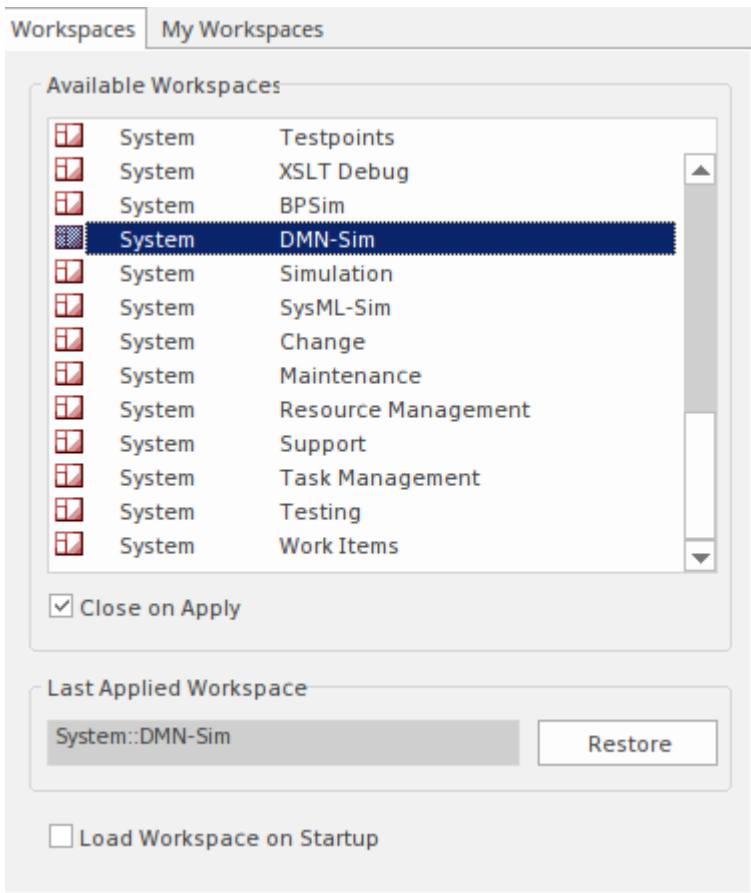


用户还可以功能地创建任意数量的自己的蓝图，为每个蓝图添加技术集。这允许主要关注于决策建模图的建模者添加其他功能，例如战略建模、需求建模、看板图和许多其他有用的图表和建模机制。有关详细信息，请参

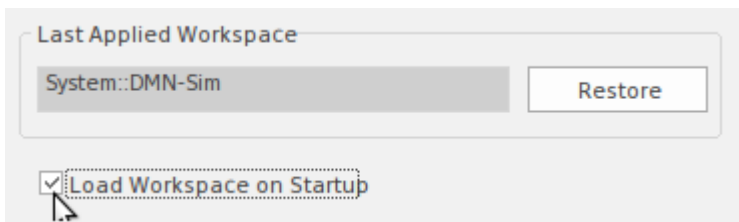
阅 [Model Perspectives](#) 主题。

选择工作空间

Enterprise Architect提供了一种快速更改用户接口布局的便捷方式，以促进特定任务或工作方式。这是通过简单地选择将更改可见窗口和工具的工作区来实现的，以提供适合任务的最有效的工作方式。例如，为需求建模定义了一个工作区，一个用于使用用例建模，另一个用于测试。我们将选择 DMN仿真工作空间，因为我们将使用 Enterprise Architect特征的仿真特性来运行我们在本指南中创建的模型的执行测试。您还可以定义任意数量的您认为有用的工作区布局，方法是打开窗口和工具并将它们放置在有助于处理特定任务或一组任务的排列中，然后保存它们。工作空间布局窗口可以通过从开始功能区的应用面板中选择工作空间项来获得。如需更多信息，请参阅 [Workspace Layouts](#) 帮助。



该对话框还提供了在启动应用程序时应用此工作区的选项，这使得主要从事决策建模工作的建模者可以轻松使用所有工具，而无需手动设置工作区。



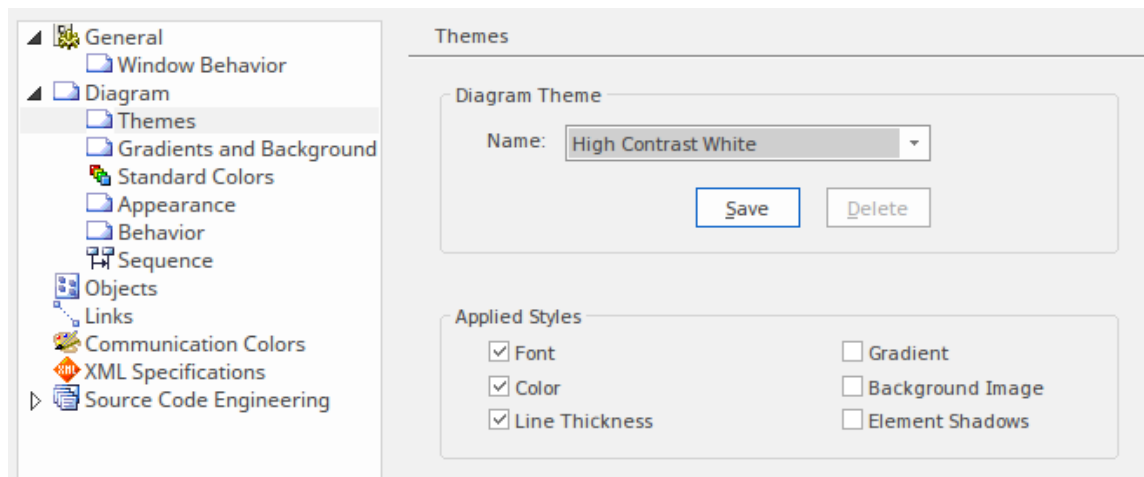
设置首选项

Enterprise Architect有一组强大的首选项，其中一些可以为整个存储库设置，另一些可以为每个用户设置。这些允许定制应用程序以适合单个建模者或整个团队。首选项包括以下选项：

- 窗口-样式、选项卡、标题和状态栏文本。
- 图表包括主题、网格、渐变、背景、样式和颜色。
- 元素——颜色、行为、显示选项。
- 连接器 -样式、线宽、方向。
- 源代码工程 -一般和特定语言的选项。

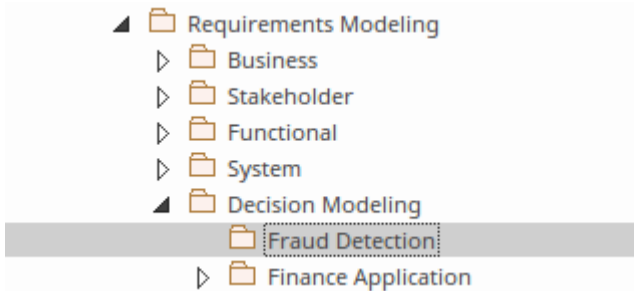
有关详细信息，请参阅[User Preferences](#)主题。

此插图显示了如何设置图表主题以及如何指定样式元素，包括字体、颜色、线条粗细和渐变。



创建图表元素和关系

一旦建立模型结构并根据您的需要定制应用程序，包括选择 DMN 蓝图和适当的工作区，您就可以开始创建图表和元素了。虽然可以在不首先创建图表的情况下创建元素，但通常的做法是从创建图表作为元素如何可视化的画布开始。您需要做的第一件事是在浏览器窗口中选择图表的位置。例如，您可能正在从事一个项目，并在“决策建模包”下定义了一个名为“欺诈检测”的包。通过选择“欺诈检测”包，您是在告诉 Enterprise Architect，您希望在此处插入新的决策需求图。

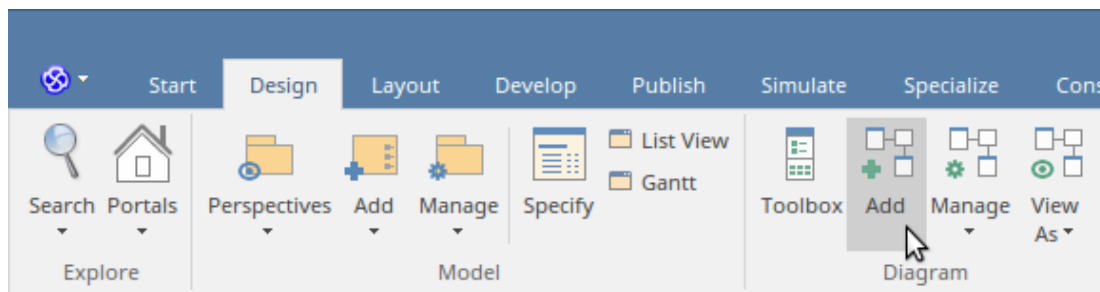


在接下来的几节中，我们将学习如何创建图表并添加元素、关系（需求）和工件，这些元素将定义描述重要决策的模型、它们的输入数据和知识源。

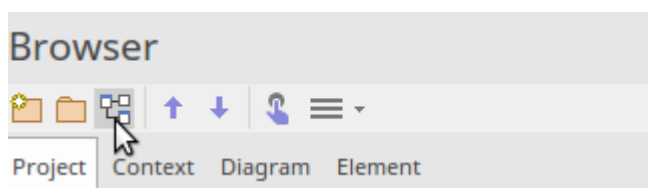
创建和查看图表

决策建模表示法实际上是一种可视化语法，虽然存在不可见的句法项，但大部分语法都是用图表直观地表达和查看的。因此，能够创建和查看图表非常重要，因为这些图表将有效地定义表达决策内容的句子。Enterprise Architect 是一个灵活的工具，提供了多种打开现有图表和创建新图表的方法，包括：

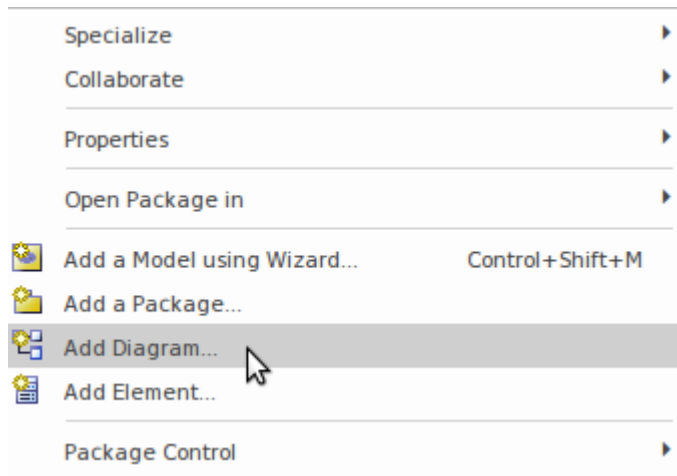
- 从丝带中选择：



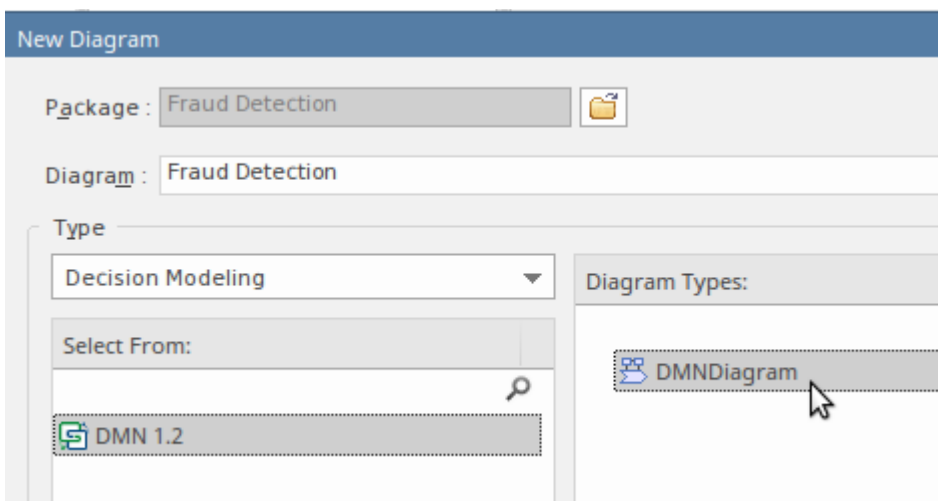
- 从浏览器窗口标题栏中选择：



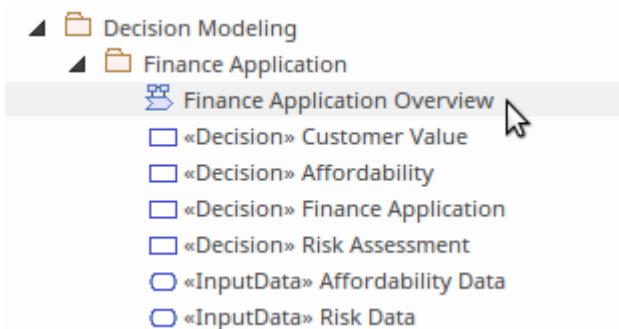
- 从浏览器窗口上下文（右键单击）菜单中选择：



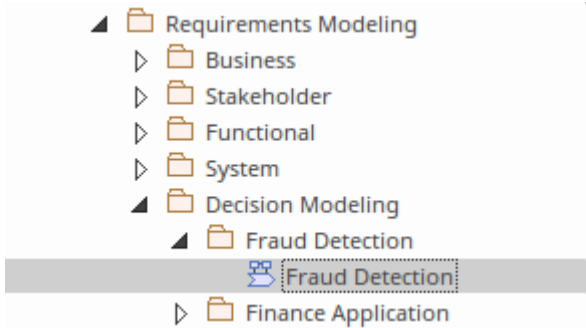
无论您选择哪种方法，您都可以从“新建图表”对话框的“图表类型”面板中选择 DMN 图表类型。



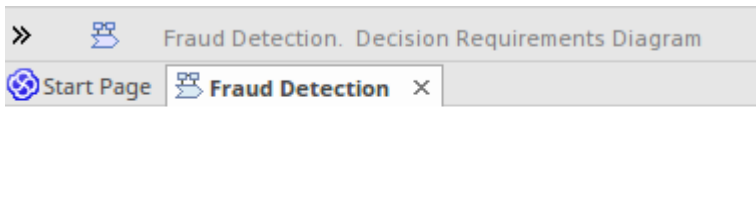
要打开现有图表，请在浏览器窗口中找到它，然后双击，或右键单击并从图表的上下文菜单中选择打开。



让我们继续创建一个决策需求图 (DRD) 来表示“欺诈检测”决策模型。选择决策需求图作为图表类型并输入适当的名称（注记将默认使用包的名称）。单击确定按钮后，将创建并打开一个新的（空白）DRD 图，并显示 DMN 工具箱，供您或您的团队成员创建元素和关系。如图所示，一个新节点将添加到浏览器窗口中，位于所选包的下方，代表图表。可以随时重命名图表，或将其移动到其他位置，并且可以添加或更新图表的属性。



Enterprise Architect将创建一个图表画布，其中包含一个表示图表边界的不可见框架。标题信息包含在画布的顶部，并显示图表名称和图表类型。图表框架可以根据需要显示在文档和其他输出中



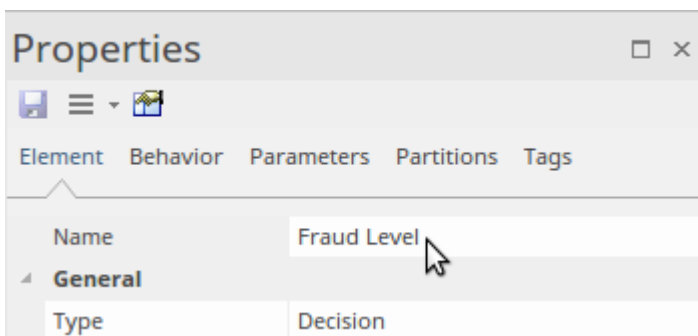
将元素添加到图表中

打开新的（或现有的）图表后，您就可以开始创建元素和关系来描述决策了。基本上有两种类型的object可以添加到图表中：

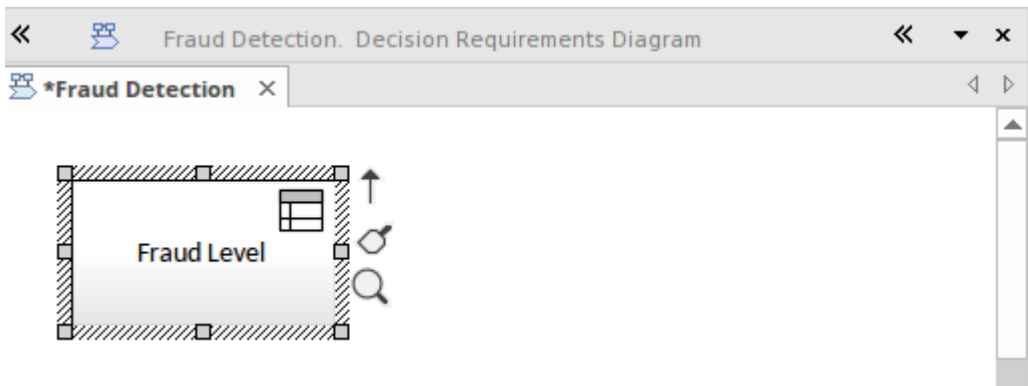
- 新元素 -通过从工具箱中拖动一个项目并将其放到图表画布上来创建
- 现有元素 -通过从浏览器窗口拖放元素放置在图表上

如果您正在开始一个新项目并且刚刚设置了存储库，则浏览器窗口中通常不会有元素，因此您将更多地使用第一个选项并从工具箱创建元素。随着项目的进展，使用第二个选项并将现有元素从浏览器窗口拖到图表上将变得越来越普遍。

我们将创建一个新的决策，因此我们将一个决策项从工具箱到图表画布上。该元素将被赋予默认名称“Decision1”。现在使用属性窗口，通常停靠在图表的一侧，通过键入默认名称“Decision1”将元素的名称更改为“欺诈级别”。



这将更改浏览器窗口和图表中元素的名称。返回图表，您将看到新添加的决策，名称为“欺诈级别”。



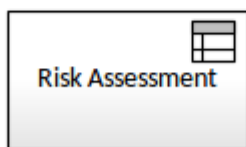
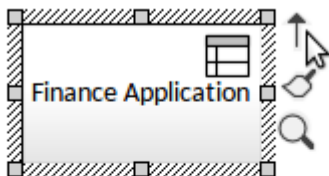
我们现在可以使用相同的方法添加任意数量的其他元素，包括其他决策、输入数据、业务模型等。这些其他元素都可以从工具箱中获得。

将关系添加到一个图表

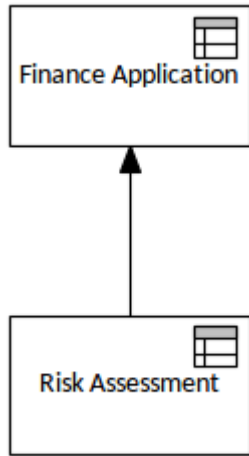
添加两个或更多元素后，您可以将它们与关系连接起来，这提供了模型中不同元素之间的语义粘合。例如，输入数据元素使用信息需求关系连接到决策元素。有两种主要方式可以将连接器添加到图表中：

1. 快速链接器 - 通过拖动快速链接器箭头（在元素的右上角）和另一个图表 *object* 之间的链接来启动直观的图表设备。
2. 工具箱项 - 可以在工具箱中选择连接器，然后在两个图表对象之间拖动。

任何一种方法都将导致在两个元素之间绘制指定的连接器。需要注意确保您在正确的方向上拖动 - DMN 信息需求连接器显示客户A需要来自供应商B的信息，因此您从A拖动到B；但信息流是从供应商B到客户A，因此生成的箭头从B指向A。相同的逻辑适用于知识需求连接器。



无论使用哪种方法，结果都将是连接两个决策的信息需求关系。可以更改连接器的方向和样式，并且可以添加任意数量的路径点，以便随着模型的开发以不同的方式对其进行布线。此图显示了添加连接器的结果，如果建模者无意中已将连接器添加到错误的方向，可以通过访问连接器上下文菜单的高级子菜单中的选项来方便地反转它。



窗口和工具

我们已经看到用户界面的某些部分可用于创建和维护决策模型，包括浏览器窗口、新图表对话框和画布，以及蓝图和工作空间功能。当您选择 DMN 透视图时，Enterprise Architect 会加载适当的内部技术，有效地将工具更改为决策建模平台，这样当您选择创建新图表时，您将只会看到创建决策需求图的选项。

Enterprise Architect 实施 DMN 标准最有用的方面之一是模拟决策模型的能力。这意味着需求分析员或其他利益利益相关者可以使用测试数据来执行模型的试运行，以模拟上下文将在决策模型时间产生的输出。因此，例如，在我们的欺诈检测模型中，可以提供许多决策模型的输入，并根据该输入和已配置的规则生成输出。

Enterprise Architect 提供了两个重要的决策窗口；我们现在将介绍这两个现在，正如您将在整个指南中看到的那样，这些窗口将被广泛使用。

- DMN 表达式窗口——用于定义决策表形式的表达式、盒装表达式和其他表达式类型
- DMN 仿真窗口 - 用于配置和运行仿真

此外，还有多种工具可用于促进使用决策模型；我们稍后将在指南中查看其中的一些内容。这些附加工具包括：

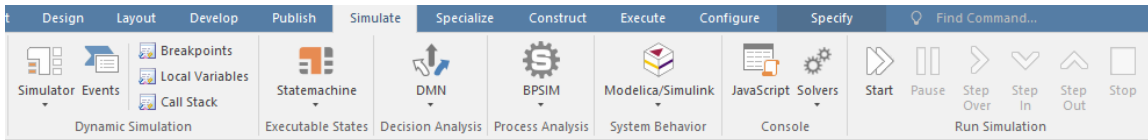
- 文档生成
- 讨论和评论
- 可追溯性
- 关系矩阵
- 图表过滤器
- 规范管理器

DMN 表达式窗口

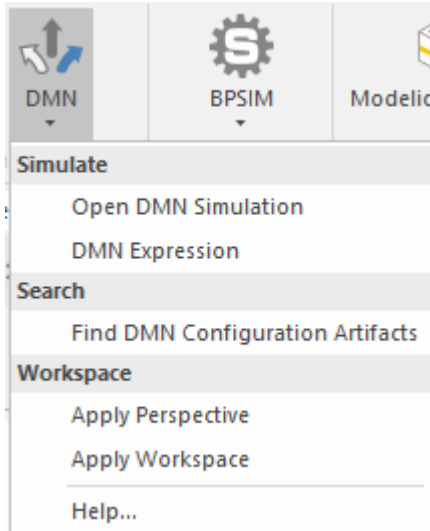
我们花了很多时间介绍上下文并描述模型的时间，包括在浏览器窗口中的位置和添加图表元素。表达式窗口是决策的机房或控制中心，它是逻辑所在的位置，根据用户定义的表达式（例如，可以是决策表中定义的规则）将输入信息有效地转换为决策输出。有一个验证功能允许测试决策表中定义的规则的正确性和覆盖率。它也是定义输入数据的地方，包括可以创建和重复使用的数据集，以使用预定义的输入数据运行模拟，从而允许以稳健和可重复的方式执行测试。

仿真功能区

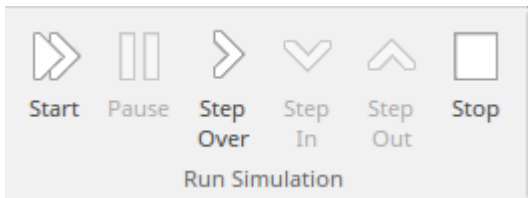
仿真功能区提供了一系列用于仿真的工具，因此我们在这里看到了建模仿真功能keep company 与许多其他仿真工具，包括业务流程仿真仿真（决策）、Modelica/Simulink（SysML 仿真）和状态机仿真功能。状态机仿真。



功能区上的功能项目有一个下拉菜单，提供了许多用于使用决策建模下拉模拟的功能。其中包括选择工作空间和透视图，并查找代表模型片段的配置项，其中包含已配置的模拟。

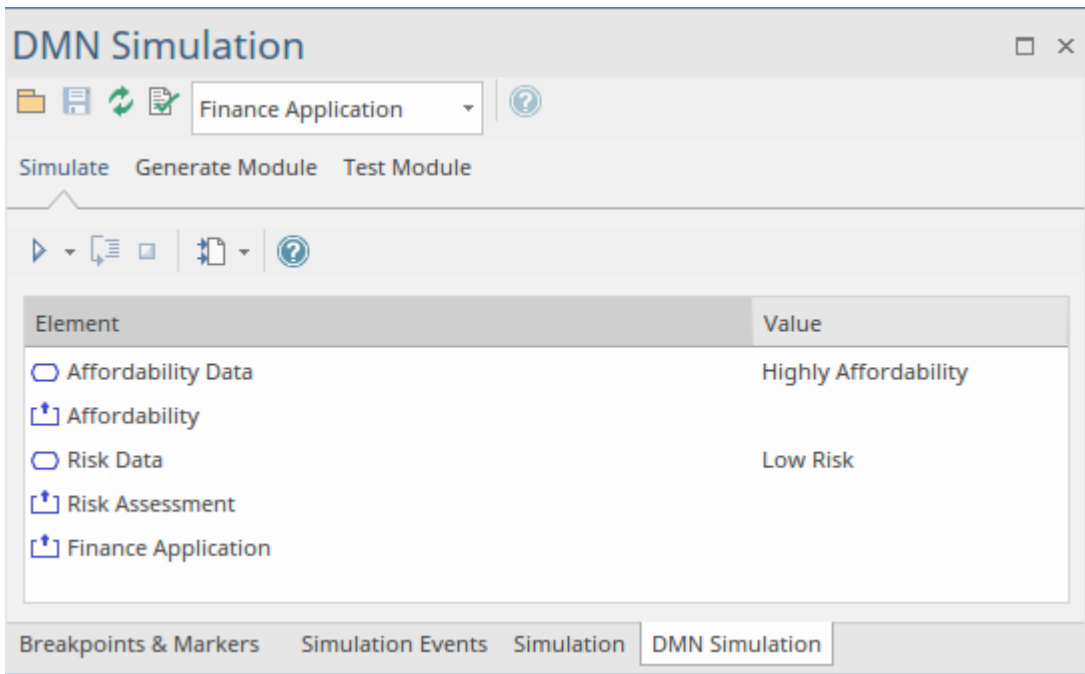


功能区提供了一个通用控制面板，用于运行任何类型的模拟；这可以在配置和加载 DMN 模拟时使用。它允许模拟运行完成、暂停、逐步执行或停止。当您开发决策模型或评估现有模型并需要了解模型的哪些部分对最终输出做出贡献时，此功能将成为您的朋友。我们将在指南的细节部分详细介绍运行模拟。

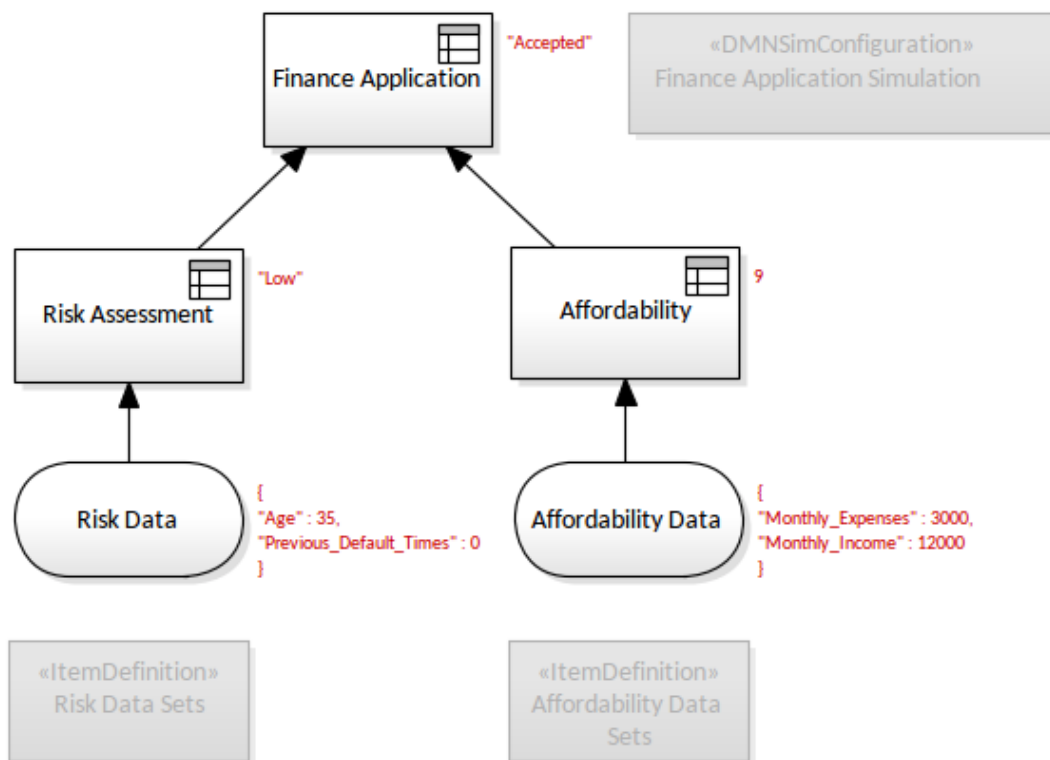


DMN仿真窗口

仿真窗口提供配置和运行仿真的功能，包括选择每个决策应使用的数据集。模拟结果将以对图表元素的注释形式显示在当前图表中。仿真也可以从运行仿真功能区的运行仿真面板进行控制，两个控制面板都允许通过显示为给定输入数据选择决策表达式（包括决策库表）中的哪些规则来逐步模拟。仿真配置加载后，模拟窗口将变为活动状态，这可以通过双击当前图表中的配置元素来实现。



第二个图演示了在模拟运行时创建的图的运行。元素具有为决策图的每个级别注释的输入和输出，并且可以看到这些从层次结构中的较低决策冒泡，直到最高决策（只有传入但没有任何传出的决策）信息要求）。



其它工具

在前面的部分中，我们向您介绍了使用决策模型时使用的主要窗口，在本节之前的主题中，我们介绍了用于决策建模的主要图表，即决策需求图。但是，在创建和管理这些模型时，还有许多其他工具非常有用，这些工具将在本文档的后面介绍。我们在讨论设置存储库结构时介绍了浏览器窗口，但我们还没有介绍存储库的其他选

项卡或视图，这对于获得决策模型部分的重点视图或“锁孔”视图很有用正在努力。

- 上下文浏览器- 提供项目浏览器中元素的重点子集
- 图表浏览器- 提供当前图表上的元素和连接器的列表
- 详情展示（检验员窗口）——提供当前元素的重要特征、属性和关系的列表

决策模型和符号概述

我们的世界瞬息万变，客户的期望越来越高，他们对不良服务的耐烦和不容忍，以及他们在竞争激烈的数字世界中拥有的无数选择，导致组织需要能够快速有效地进行变革，并了解支撑其业务模型的决策以及这些模型的实施。

决策建模并没有被广泛认为是一门独立的学科或科学，业务规则和决策以异构方式建模，组织内部或组织之间几乎没有一致性。决策以级联图表的形式从业务流程表进一步处理，使它们变得复杂且难以更改，并将它们移动为直通式的“圣杯”。这种情况使协作和团队工作几乎不可能，并导致规范定义不明确、无数的编程错误、沮丧和最终不满意的客户。

本节讨论什么是决策模型和表示法（上下文），它在业务流程建模和软件工程等其他学科的时间安排中的位置，以及一个好的模型的特征是什么。本节还讨论了使用级别以及使用 DMN 的原因和时间，并给出了第一个示例，以帮助新手了解该标准的好处。

实用的建模和可用于 Enterprise Architect 业务分析师和软件工程师和其他人的广泛的功能方法使其成为一个有用的工具，作为存储库决策和创建、管理、模拟、实施和传播决策模型的平台并作为一个团队进行协作，为组织带来真实和可重复的价值。

DMN Components

- Decision
- Business Knowledge Model
- Knowledge Source
- Input Data
- Item Definition
- Decision Service

什么是决策模型和表示法

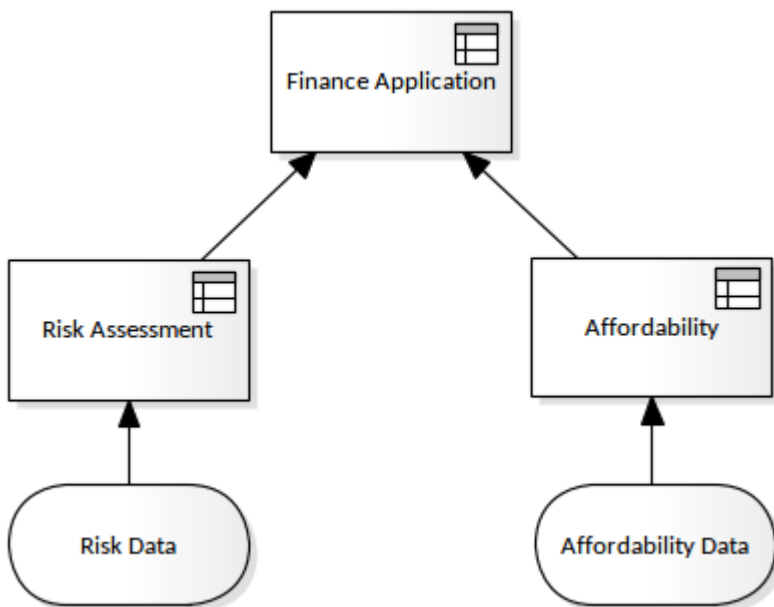
决策模型和符号 (DMN) 是由物件管理组 (OMG) 发布和管理的标准。它提供了一种标准方法，用于在组织或计划内描述、建模、模拟、重用和实施可重复的决策。它还旨在促进组织之间决策模型的共享和交换。

建模符号包含一个可视语法，允许以一种使业务和技术受众都可读的方式记录决策和业务规则，从而确保决策和规则不会被误解。由此产生的决策模型还提供了如何使用足够友好表达语言(决策) 来评估决策表中定义的决策逻辑的定义。

DMN 定义了两个级别的语言，它们与组织角色完全一致：

水平需求

与设计、分析和定义决策规则的业务人员保持一致。这为拥有业务的人提供了一种机制来模型规则，而无需了解如何实施规则的技术知识。



逻辑电平

与实施团队保持一致，实施团队扩充定义，以包括可由业务人员维护的决策表形式的决策逻辑，或由技术人员管理的表达逻辑。

	Affordability Data . Monthly Income	Affordability Data . Monthly Expenses	Affordability	Notes
U				
1	>10000	<=3000	9	Exemplary
2	[5000..10000]	<=2000	8	Good
3	<5000	<=1000	4	Requires Review

在这个例子中，我们看到主要决策有两个来自其他决策表的输入，它们首先被处理，并有助于决定是否接受应用程序的整体决策。本表为风险评估决策表。

DMN Expression

Risk Assessment

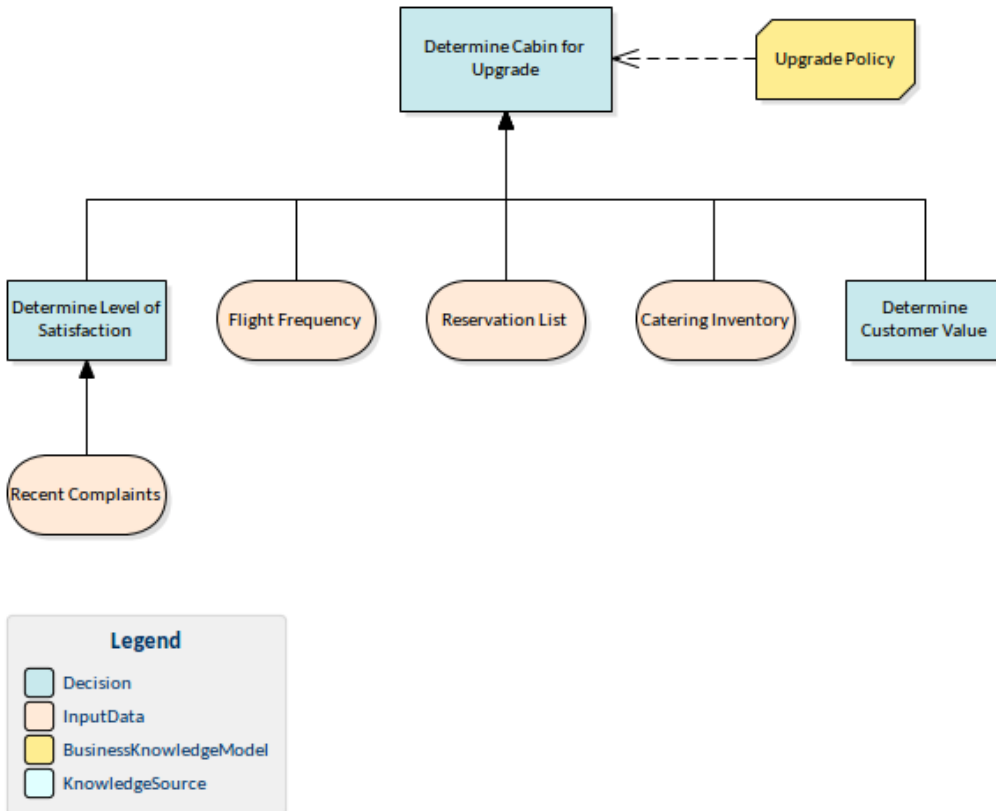
	Risk Data . Age	Risk Data . Previous Default Times	Risk Assessment	Notes
U			<i>Low, Moderate, High</i>	
1	<18	>3	<i>High</i>	<i>Exercise Discretion</i>
2	<18	[2..3]	<i>Moderate</i>	
3	<18	<2	<i>Low</i>	<i>Preferred Applicant</i>
4	[18..55]	>3	<i>High</i>	
5	[18..55]	[2..3]	<i>Low</i>	<i>Preferred Applicant</i>
6	[18..55]	<3	<i>Low</i>	<i>Preferred Applicant</i>
7	>55	-	<i>Moderate</i>	

DMN Expression DMN Simulation

A 一个示例

想象一下，您是一名航空公司预订官员，在一家繁忙的国内航空公司的值机柜台工作。准时起飞至关重要，因为延误可能导致机场管制员收取费用，并可能不得不在较低的高度飞行，从而增加燃料成本和支付其他罚款。更不用说可能错过重要会议的乘客以及下次飞行时可能会选择另一家航空公司的乘客带来的不便。

想象一下，您正在繁忙的国内机场检查乘客。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；您必须将一些乘客升级为业务或类- 但应该选择哪些乘客？ A做出决定，但应该考虑哪些因素？忙碌的值机员无法做出此决定。他们可以通过查看一些他们应该考虑的因素来帮助他们，这些因素已经记录在使用决策需求图的决策模型中。



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客旅客，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”都可以记录在决策决策表中，明确哪些乘客应该升舱，哪些舱位、业务或头等舱。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。

DMN Expression

Cabin Upgrade Policy with Catering

Input Parameter Values for Simulation

(Flights in the last month, Number of Pax Overbooked)					
U	Flights in the last month	Number of Pax Overbooked	Catering Available	Upgrade Cabin	Annotation
			True, False	Business Class, First Class	
1	<=1	<=2	True	Business Class	
2	<=1	(2..8]	False	Business Class	
3	<=1	>8	True	First Class	Start Filling First Class when heavily over...
4	(1..5]	<=2	False	Business Class	

该表分为列和行。有两种类型的列：做出决策所需的输入和应用规则的结果的输出。这非常有帮助，但仍然需要忙碌的源人员能够找到在决策表中找到正确行所需的所有信息。即使所有这些信息

都可用，不恰当的决定仍然可能是由于人为错误选择表中的错误行

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。因此，我们忙碌的值机员在为乘客办理登机手续时无需做任何事情或做出任何决定；如果特定乘客有权获得升舱，则预订将自动更改，并且升舱将显示在值机屏幕上。航空公司预订官唯一需要做的就是向乘客提供受欢迎的消息——“Travelalot 先生，我们很高兴在您今天早上飞往纽约的航班上为您提供头等舱升级服务”

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。决策需求图可以更改以包括新的输入、修改、测试和模拟的决策表以及重新生成的编程代码。一旦将更改推送到机场系统，就会神奇地升级合适的乘客。签到人员仍然可以在培训和简报会期间查看决策表，以了解规则。许多实施引擎还将就决定的基础提供商业友好的解释，以便值机人员可以向乘客解释升级的原因。

用途等级

决策模型和决策的创建者打算让不同组织以及同一组织内以多种方式使用它。将模型划分为需求、逻辑和实施级别优先于以下使用场景：

- 建模Human Decision-Making
- 用于自动化决策的建模需求
- 建模自动决策

一些教科书和白皮书将这些视为决策领域的成熟度水平，但未能解释在许多情况下——即使在高度自动化的系统中——人类做出决策是重要的，甚至是必要的。最终决定，特别是在人身安全领域；这些不会自动化，但出于治理和监管原因需要进行描述。

建模Human Decision-Making

DMN 可用于模型由员工而非计算机系统执行的企业、组织或倡议级别的决策。通常，决策是在相当高的层次上描述的，规则通常用自然语言编写，但可以使用更正式的机制来描述，例如决策表。在此级别，使用业务模型来模拟知识如何与决策相关是有用的，该模型捕获业务知识的特定领域及其对一个或多个决策的适用性，例如一组标准操作过程。还可以对描述业务来源的知识源进行建模，例如标准操作过程手册。这种类型的建模既可以是描述性的，也可以是规范性的。

为自动化决策制定建模需求

当 DMN 用于对自动化决策系统的需求进行建模时，重要的是定义输入和输出，并将模型开发到自动化系统所需的保真度水平。

建模和实施自动化决策制定

Enterprise Architect是一个全生命周期工具，可用于自动决策模型系统的模型决策，并可以生成多种语言的编程代码，从而允许决策需求及其相关定义包含在实现代码中。这是一种有用的机制，可以提高生产力，并确保组织中最高级别的思考者与处理决策并确定其输出的最终自动化系统之间存在不间断的链条，所有这些都集中在一个工具中。

为什么要使用决策模型和符号

在企业或倡议级别使用 DMN 的原因有很多，这将带来业务和技术价值。下一个主题中描述了许多好处，但这里描述了最引人注目的：

降低复杂性

如前所述，当前的业务规则和决策通常位于许多地方并以多种格式存在。它们经常出现在过于复杂的业务流程流程图中，其中包含描述结果的级联网关集。这些图表通常很笨拙，几乎无法揭示输入、业务知识、权限或决策逻辑。

自动化和执行

由于业务分析师定义决策、输入和业务知识，技术人员以表达式和决策表的形式详细阐述逻辑，从而可能创建决策服务，这些模型可用于生成可以执行的编程语言代码以自动化决策并使它们可用于运行时系统。

对更改的敏捷性和响应性

由于将决策模型与业务流程模型分离，并创建了实施和自动化管道，业务和技术团队可以以创纪录的速度响应业务变化。这些模型随后成为促进敏捷性和业务响应能力的平台。

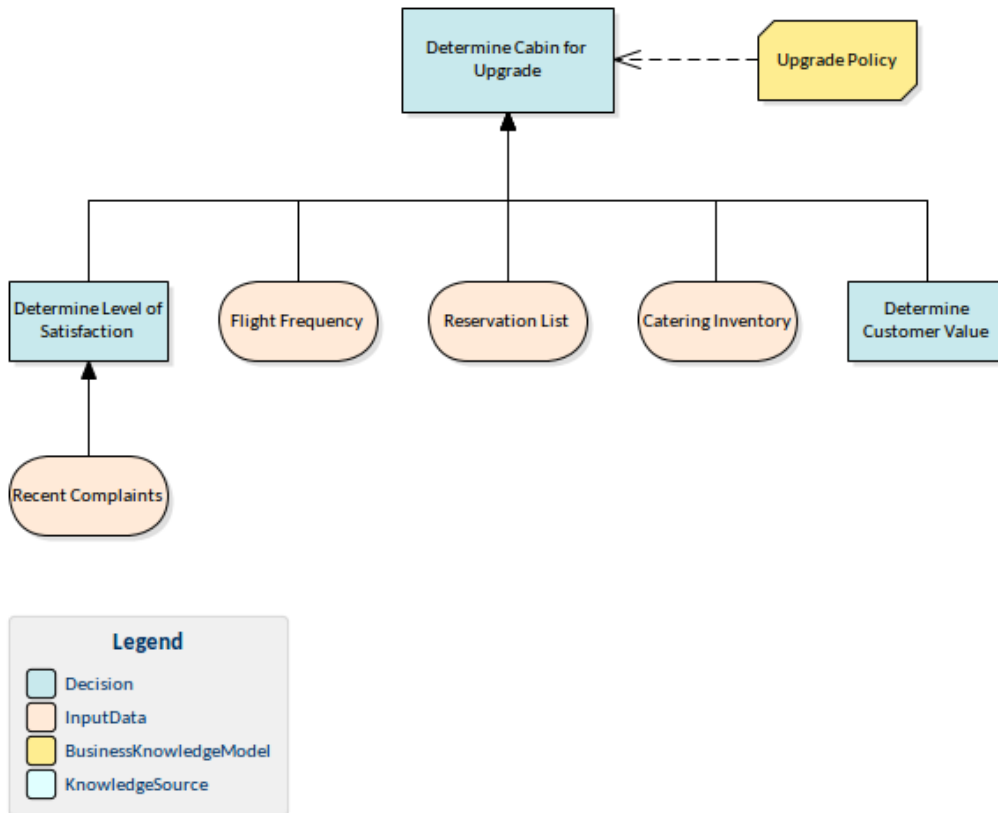
何时使用决策模型和符号

决策模型和符号 (DMN) 可用于广泛的领域，从建模高级战略决策到制定有关硬件设备的特定决策。本节描述了一些使用示例，这些示例将帮助您了解何时可以使用 DMN 以及决策模型中可用的元素。

- 确定为客户提供的折扣级别
- 决定哪些客户应该升级到类
- 决定授予新员工或承包商的系统访问级别
- 决定为给定访问者显示动态网络站点的哪些页面
- 决定何时将来袭飞机送入等待航线
- 决定何时订购库存商品，以及订购数量
- 决定何时出售股票或资产

决定客户是否应升舱至业务或头等舱

为了确保每个航段实现最佳利润水平，航空公司通常会使用严密保护的算法超额预订所有可用客舱。随着时间接近预定起飞时间，超额预订的水平会降低。这有时会导致售出的机票多于可用座位，尤其是经济舱。为了解决这个问题，航空公司通常会在办理登机手续时为经济舱乘客提供升舱服务。为乘客升舱的决定应该基于商业上的重要因素，并且是一种典型的情况，决策模型可以帮助创建可重复的、公正的和具有商业价值的结果。



Enterprise Architect 可以用来创建决策需求模型，可以添加表达式和真相表来定义逻辑，并且可以从模型中自动生成实现代码。

确定为客户提供的折扣级别

贴近机构前线的业务人员，对“顾客为王”这句话是耳熟能详的。维护现有客户并获得新客户对于企业的生存能力至关重要。向客户提供折扣是一种有效的激励机制，可以增加“销售并在竞争占主导地位的环境中提供有竞争力的价格。将这个决定留给忙碌的销售代表通常会导致在没有任何商业原因的情况下给予折扣，或者更糟糕的折扣导致销售流向竞争对手并有可能失去客户。决策模型可以A可重复的和商业上正确的选择，即哪些客户应该获得折扣以及折扣的百分比或数量。

决定授予新员工或承包商的系统访问级别

大多数组织都了解员工对其业务整体成功的重要性。他们也知道为员工提供正确的工具来履行其职责的重要性，包括对他们用于执行工作的各种软件应用程序的正确访问级别。入职人员的过程已经足够复杂，技术官员无法确定应用程序列表和所需的访问级别。

决定应订购库存商品的时间和数量

制造商在竞争激烈的环境中运营，生产高质量的产品，同时将成本保持在最低水平，这对于成功和可持续的业务至关重要。关于何时补充库存或零部件以及订购多少的决定取决于许多因素。错误和订购不足会对生产线或销售底线产生影响，订购过多和库存不足会导致成本高昂，并且会影响现金流。以生产线为例

决定为给定访问者显示动态网络站点的哪些页面

网络网站从互联网诞生之初的早期化身已经走过了漫长的道路，现在它们只是总体数字战略的一个方面。可以识别单个客户或访问者类别，并使用从先前交互到客户关系信息（例如购买记录）的大量可用数据，可以确定向他们显示的最有效的页面集并创建兴趣跟踪将引导他们购买。手动执行此操作是不可能的，这些规则通常已内置到控制网站动态方面的服务器和客户端脚本中。

创建一个决策模型，不仅描述输入和规则，而且将它们纳入总体数字战略，包括：社交媒体、业务架构和客户关系管理将为任何组织提供竞争优势。然后，该网站将成为一个动态决策驱动引擎，该引擎基于复杂的引擎创建内容，其中规则是可见的，并且是业务和技术协作产生的易于理解和清晰表达的模型的一部分。

决定何时将来袭飞机送入等待航线

航空服务是在全球和本地运输人员和货物的最常见和需求最大的形式。有许多公司竞相提供这些服务，虽然航站楼的数量和规模不断增加，但它们仍然是一种有限的资源，需要对使用它们的航班以及通过它们的人员和货物进行非常仔细的管理。在每天必须做出和协调的数百万个决定中，考虑一个：何时应指示航班在着陆前以等待模式飞行？如果当地存在危险情况，一个答案可能是立即的，但它是本地化的，预计很快就会得到解决。否则，航班将被重定向到另一个机场（在那里他们也可能被引导到等待模式，只是为了让他们适应正常的时间表），或者另一方面，如果一个小因素意味着立即着陆不方便。但是，一个多么小的因素，多么方便，可以证明这种行为是合理的呢？提出问题后，还需要做出其他决定：必须采取哪些行动来处理延迟着陆？必须权衡和接受哪些后果？考虑到最初问题的答案：何时？决策模型将是A非常有价值的工具，可以管理这种情况，更不用说仅在一个机场的日常运营中出现的所有其他情况了。

决定何时出售股票或资产

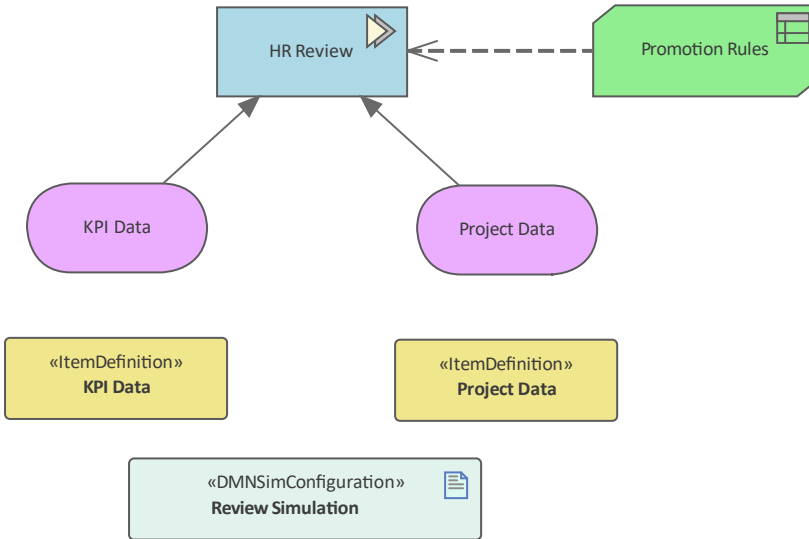
股票市场是一个不稳定的机构，买卖股票的决策似乎通常只取决于一时兴起。然而，实际上有一些好的和可靠的指标和因素可以被评估和交叉链接，以指示是否应该采取买入、持有或卖出头寸。决策模型是A很好的工具，用于权衡有关持股公司管理、损益指数或投资收益指数等重要问题的答案，无论您是在寻找较弱的股票进

行剥离以启用购买更强的股票，或仅仅寻求减少对弱势股票的敞口或利用强势股票的时机。在向投资者推荐头寸或投资组合之前，投资顾问或股票经纪人也会特别关注投资者概况以及公司业绩指标。这可能会使用两个决策模型，一个模型的输出（可能是风险因素或当前风险暴露）被输入作为另一个模型中问题的答案。

示例决策模型

这个示例决策模型使用决策模型和符号语言的基本和通用元素来演示和举例说明我们都可能在某些时候感兴趣的业务决策 - 人力资源审阅决策导致可能的加薪或晋升。

HR 审阅决策元素并不定义规则，而是属于 *Invocation* 类型，调用 *Promotion* 元素的服务。这是一个可以在不同情况下重复使用的业务知识模型 (BKM)。BKM 在内部存储的决策表中定义了提升规则，当它被调用时，它会将结果输出传递回决策元素。



决策元素有两个输入，即关键绩效指标 (KPI) 数据和项目数据——这些数据被用作决策的输入。Enterprise Architect 有一个有用的机制来定义不同的数据集，这些数据集可用于模拟决策并有效地允许业务和技术用户执行“假设”分析，作为生产前测试或生产中支持分析的一部分。决策表达式编辑器可帮助您在易于使用的表中创建和修改规则，为输入和输出列着色，并具有许多内置特征，使业务和技术用户可以轻松地处理决策。可以插入任意数量的注释列来创建额外的文档来解释表达式。

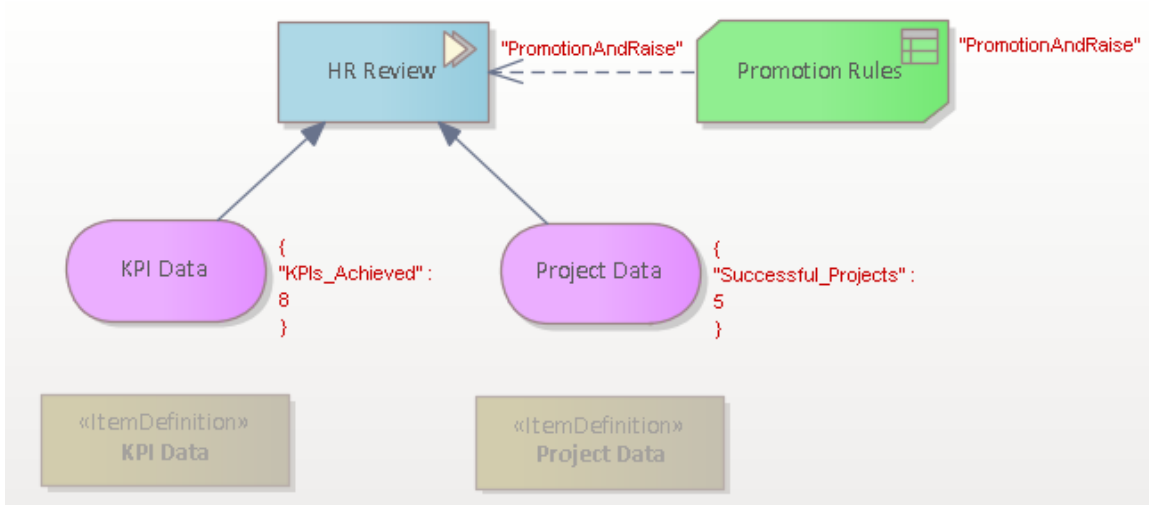
DMN Expression

Promotion Rules | Input Parameter Values for Simulation

(KPI Achieved, Successful Projects)

U	KPI Achieved	Successful Projects	HR Decision	
			PromotionAndRaise, Raise, Promotion	
1	<3	<3	Promotion	Increase in Responsibilities but no Raise
2	[3..7]	[3..7]	Raise	Increase in Wage but no promotion
3	>7	>7	PromotionAndRaise	Increase in Responsibilities and Raise
4	>7	[3..7]	PromotionAndRaise	Increase in Responsibilities and Raise

Enterprise Architect 是一个完整的生命周期工具，一旦创建并测试了决策模型，它们就可以生成为实现代码，从而确保业务定义的决策体现在代码中。作为测试过程的一部分，该工具提供了一个模拟引擎，允许模型可视化，就好像它们在生产系统中一样。可以定义任意数量的数据集并将其与仿真一起使用，以测试决策逻辑，从而避免生产系统中发生错误。



决策模型和符号的好处

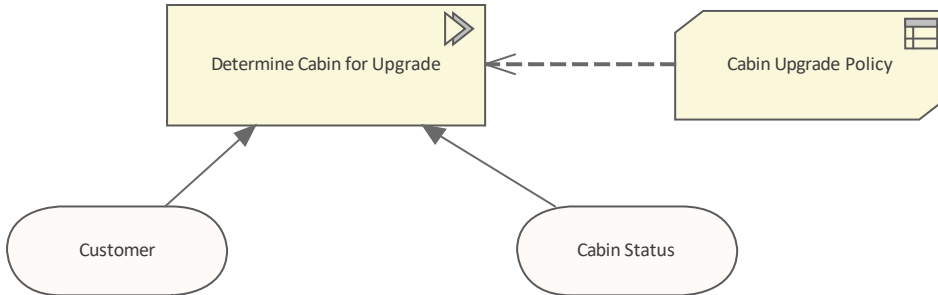
团队或组织可以从使用决策模型和相关的符号来记录计划、项目、部门或企业的重要决策中获得广泛的好处。这些好处中最引人注目的可能是能够在单个模型中收集和可视化规则，并查看它们如何相互关联以及与用于确定输出的输入、业务知识模型和知识源相关联。还有很多其他好处，包括直接从模型以编程代码的形式生成决策实施的能力。毫无疑问，每个组织都将从使用决策模型和符号中获得无数好处；本节列出了一些核心优势，包括以下能力：

- 可视化决策和规则
- 简化业务流程流程图
- 可视化输入和分析
- 交流共同的理解
- 促进协作
- 发现自动化机会
- 将信息整合到架构模型中
- 创建一个单一的“决策”真理源
- 仿真决策模型
- 生成编程代码
- 与其他模型衔接
- 选择一致性和正确性
- 找出规范中的差距和重叠
- 发现规范中的错误
- 测试模型
- 创建可重用的决策库

可视化决策和规则

业务规则和决策通常分散在计划的环境中，无论是在规范层面还是在实施层面，使得管理它们的过程非常困难 - 如果不是难以处理的话。

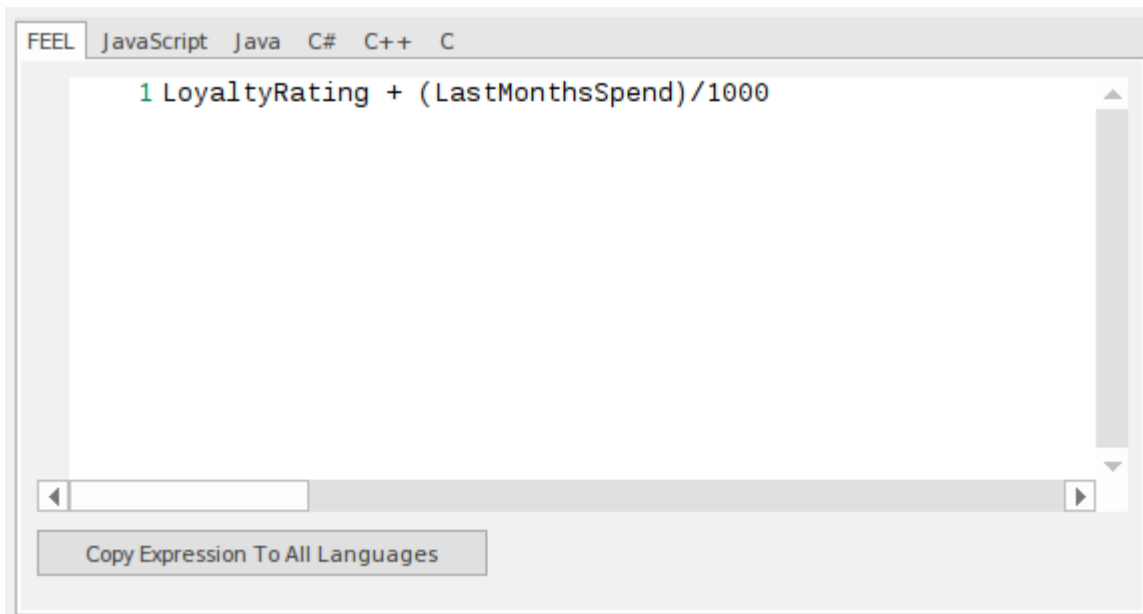
使用Enterprise Architect，可以将决策收集在一起并以通用语言和格式进行管理，并以多种方式可视化。



有许多简单的方法可以使决策的逻辑可视化，使分析人员或非技术利益相关者易于指定、阅读或更改逻辑规范。其中最引人注目的可能是决策表，它在一组相交的列和行中表达逻辑。通常，列定义输入和输出，行列出可能的输入值，并组织成将这些离散值映射到离散输出值的规则。Enterprise Architect还允许将决策表倒置，以便列阐明规则，而行列出输入和输出。

定义规则的另一种方法是使用表达式语言，这对于简单的情况会更有用。Enterprise Architect允许您以多种实现语言定义表达式，包括：

- 足够友好表达语言(FEEL)
- JavaScript
- Java
- C #
- C++
- C



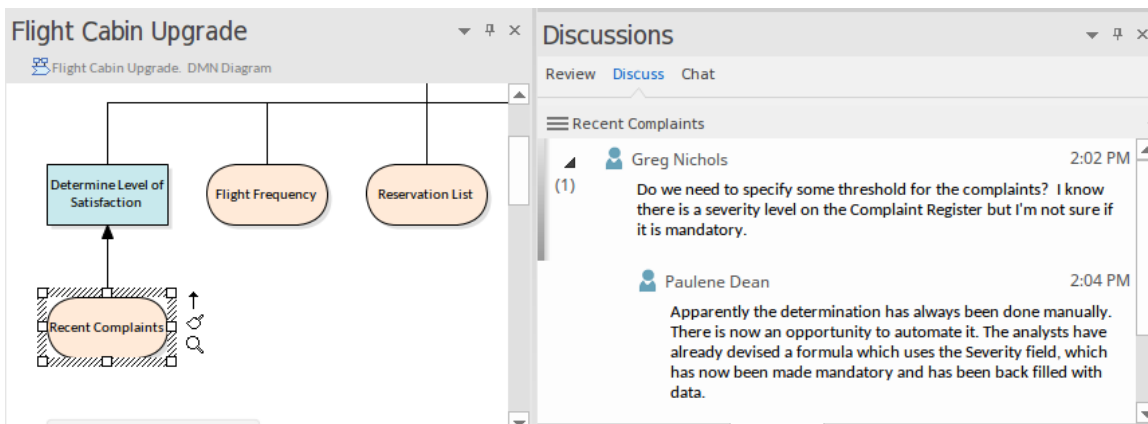
促进协作

决策不是孤立地制定的，而是需要协作努力，通常以某种类型的战略意图开始，并以作为手动业务流程或在信息系统中自动化的决策模型的实施结束。无论哪种方式，实现的决策过程都可以执行数百次，或在计算机系统的情况下，每天可以执行数百万次。

Enterprise Architect是一个团队协作平台，具有复杂的工具，可确保所有需要为模型的创建、开发和维护做出贡献的团队成员可以在同一个模型中进行协作，无论他们在团队中的角色如何，他们在哪里位置，或它们如何连接到模型。这些工具包括：

- 评论 - 可以分析和批评模型和表达式
- 团队图书馆- 可以查看的文档和网络资源的集合
- 元素和图表讨论 - 与建模人员和业务用户就元素和图表进行交流的能力
- 聊天 - 用于即时对话的模型内聊天
- 模型邮件- 一种产品内通信工具，允许消息参考模型元素和图表
- 模型日历 - 用于重要团队活动的模型日历

此图显示了讨论功能，可用于促进在不同建筑物甚至不同国家和时区工作的团队之间的协作。



当客户使用云服务器环境时，任何人都可以从任何地方查看模型并为模型做出贡献，包括通过这些设备上的内置浏览器：

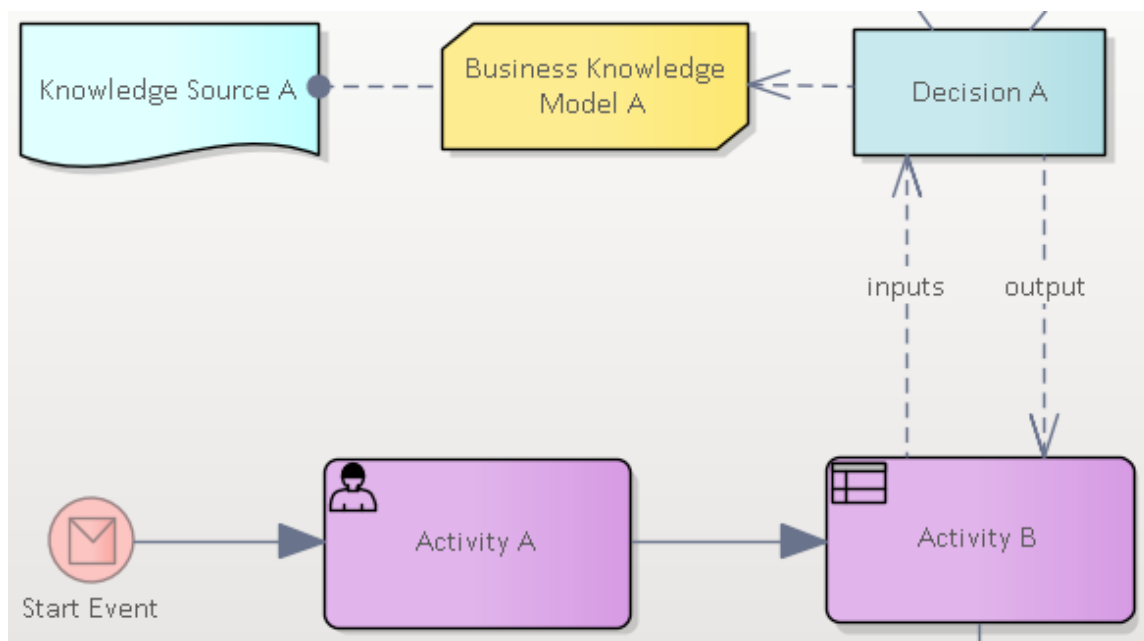
- 智能手机
- 平板电脑
- 笔记本

这意味着从战略思想家到实施和支持人员的所有级别的用户都可以访问模型并随时随地使用他们手头的任何设备进行协作。因此，业务线经理可以在机场打开他们的智能手机并参与有关特定决策的数据输入的讨论，或提供有关给业务源模型的最来源的信息。软件工程师可能乘坐家旅行，并使用平板电脑审阅一些实现细节或生成的编程代码。

简化业务流程图表

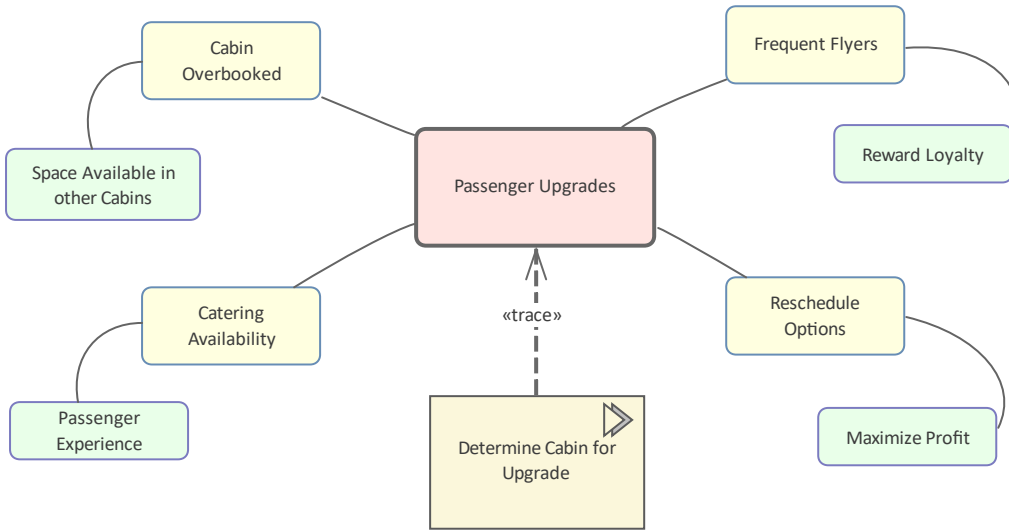
使用业务流程模型和表示法 (BPMN) 标准或 Unified Modeling Language (UML) 活动图创建的业务流程模型通常包含大量级联网关形式的决策。这些引入了不必要的复杂性并抑制了读者理解过程意图的能力。它们与直通式处理原则背道而驰，使流程变得庞大且常常笨拙，从而削弱了它们对业务和技术受众的价值。

使用 Enterprise Architect 对业务流程和决策模型进行建模意味着可以以符合直通式处理原则的方式简化、精简和创建流程图。复杂的决策逻辑可以被移除，一个元素可以简单地引用一个单独但连接的决策模型，所有这些都方便地位于同一存储库中。

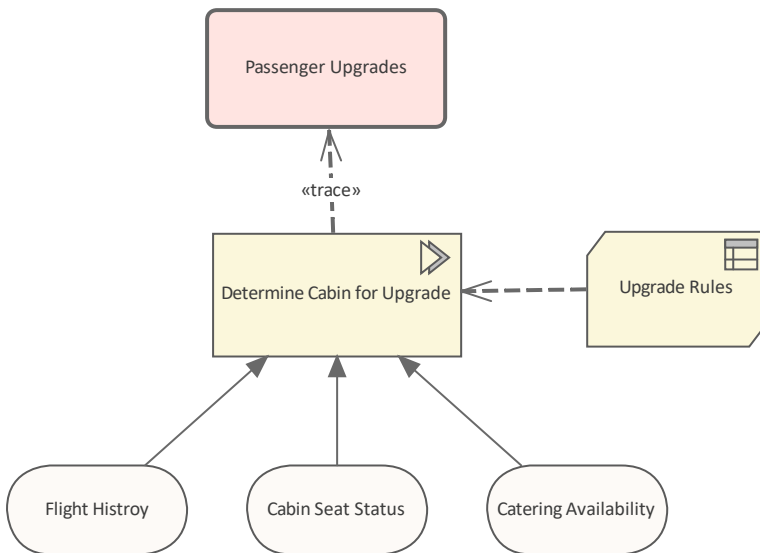


交流共同的理解

组织制定的决策和规则通常会塑造组织被客户和供应商感知的各种方式。这些决策不是由一个人做出的，而是许多人工作的结果，包括战略思想家、经理、业务分析师、系统分析师、软件工程师、测试人员等。通常，这些小组使用不同的工具在不同的地点工作，并且只在会议或电子邮件中与每个小组进行交流。这些决定通常记录在会议记录或电子邮件中，并通过“注记通信”过程开发和实施，其中很少有其来源的踪迹。



Enterprise Architect是一个协作平台，它为所有团队成员提供组织和项目信息，从构思决策需求的战略家到实施决策并确保其正常工作的工程师和支持人员。决策的必要性通常在研讨会期间确定，并且可以使用思维导图或会议记录立即将最初的想法记录在Enterprise Architect注记。



可以创建决策，A业务分析师决策模型，该模型描述了做出决策所需的输入这可以包括作为该知识源的业务专家和权威机构。决策可以回溯到思维导图中的主题，可以追溯到战略层面，业务经理也可以清楚地看到他们的想法是如何被分析和描述的。

发现自动化机会

一个组织应用的决策和规则通常是经过多年创建和实施的，并且通常被隐藏在表面之下，几乎需要业务分析师作为考古学家才能找到它们。这些规则内置于工作流程和工作描述中，写在鲜为人知的手册中，部分定义在编程代码或数据库过程中。所有这一切都让试图自动化流程和提高组织效率的技术专家变得困难，因为他们看不到规则。

使用Enterprise Architect来定义决策模型中的决策和规则不仅可以使它们可视化，而且可以清楚地理解自动化的机会以及随后实施的机会。决策模型与业务流程模型的结合意味着整个流程块可以被自动化决策制定解决方案移除和替换。结果通常会被组织的客户直接观察到，他们将享受减少的等待时间、更大的灵活性、更准确的流程，并可能降低服务成本。

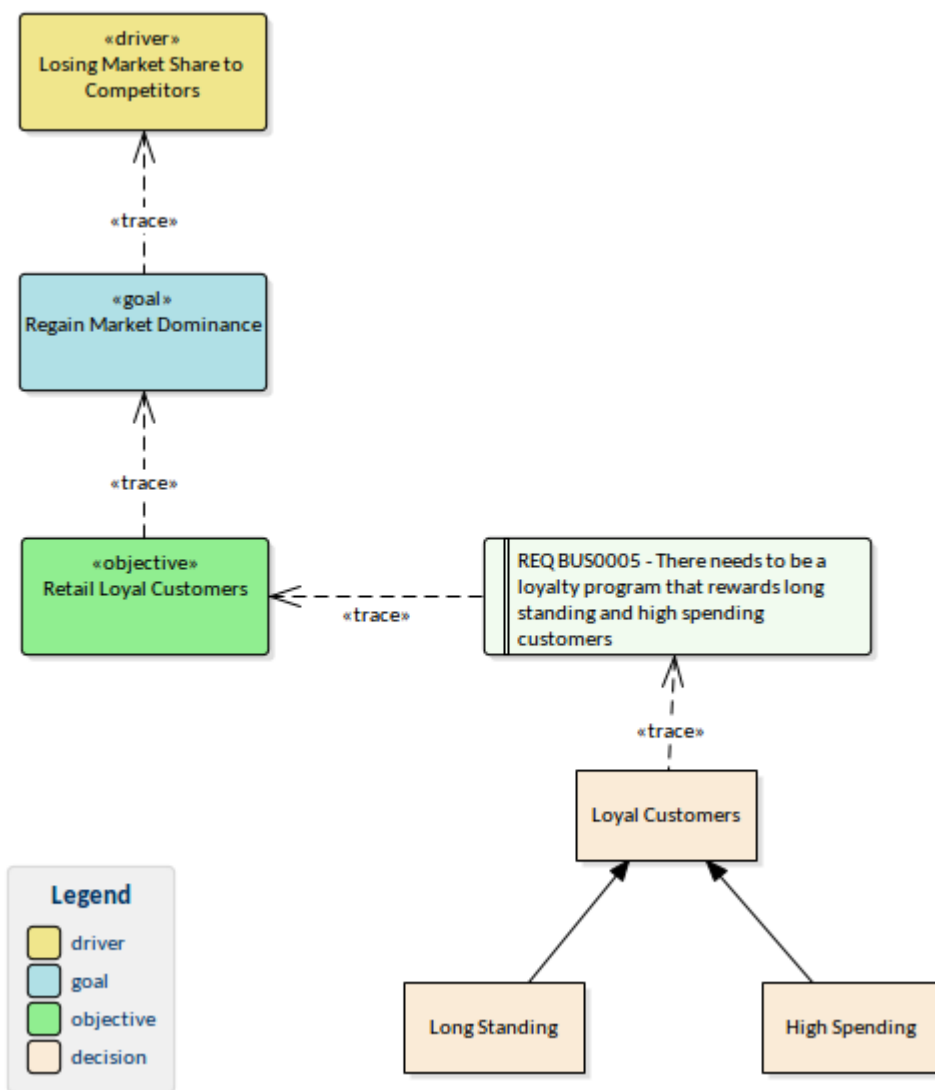
The screenshot shows a software interface titled "DMN Expression" with a toolbar and a decision table. The table is titled "Cabin Upgrade Policy with Catering" and is used for "Input Parameter Values for Simulation". The table content is as follows:

(Flights in the last month, Number of Pax Overbooked)					
U	Flights in the last month	Number of Pax Overbooked	Catering Available	Upgrade Cabin	Annotation
			True, False	Business Class, First Class	
1	<=1	<=2	True	Business Class	
2	<=1	(2..8]	False	Business Class	
3	<=1	>8	True	First Class	Start Filling First Class when heavily over...
4	(1..5]	<=2	False	Business Class	

并入架构模型

创建架构是为了将企业或计划从基线（当前）状态引导到目标（未来）状态。它们通常包含从业务到技术架构的广泛架构级别和类型，并将目标、驱动因素和能力等概念转化为路线图，这些路线图定义了企业或计划将采取的方向，以达到其目标状态并因此目标。可以将决策构建到架构模型中，从而将它们可视化为架构的一部分。这些决策通常会在高层次上定义，但会捕捉到战略和解决方案层次的思维，否则这些思维会在文本文档中丢失。

Enterprise Architect正在迅速成为企业和解决方案级别架构的行业首选工具，并且能够将决策模型构建到架构模型中将确保从低级别实现到设计和架构级别（包括需求）的可追溯性水平。

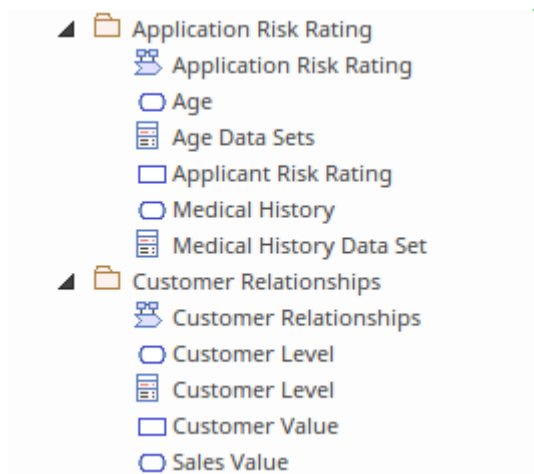


为决策创建单一真理源

困扰组织并导致许多错误和不一致并成为灵活性和敏捷性障碍的事情之一是业务规则和决策传统上在许多地方进行管理。这些范围从电子表格到战略文档、思维导图、需求文档、数据库模型和编程代码。

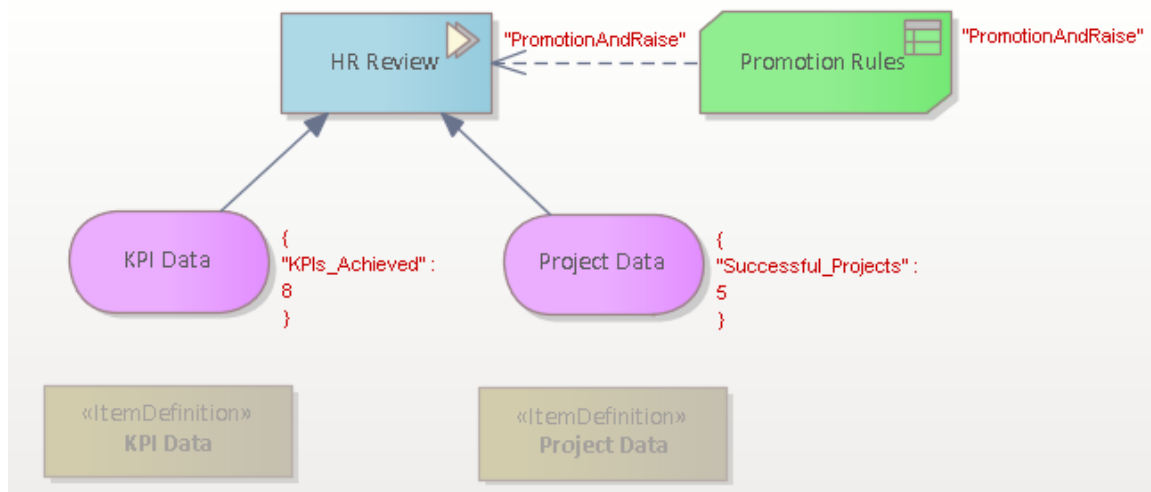
使用 DMN 和 Enterprise Architect，就有可能拥有单一的事实源，并能够在可搜索的存储库中管理这些所有重要的公司信息，该存储库可以生成文档和实现代码。

将决策放在一个存储库中解决了管理决策网络的问题，这些决策创建了一个复杂的结构，支撑了组织提供的业务价值。



仿真决策模型

模拟可以被视为模型建模的“决策解决方案”，但除了它们的即时吸引力之外，它们提供了一种有效的方法来可视化、理解、分析和传达模型的意图和含义。模拟运行在模型开发时、投入生产之前、结构模型更改之后或规则或示例数据集更新之后在模型上运行。模拟功能从根本上说是对整个模型或复杂模型中的单个决策运行试验模型执行的机制，允许团队或个人利益相关者查看输入，可视化执行路径，并分析中间决策和给定模型或模型片段的输出。模拟对于执行假设分析和检查域中的新数据集将如何影响决策输出也很有用。

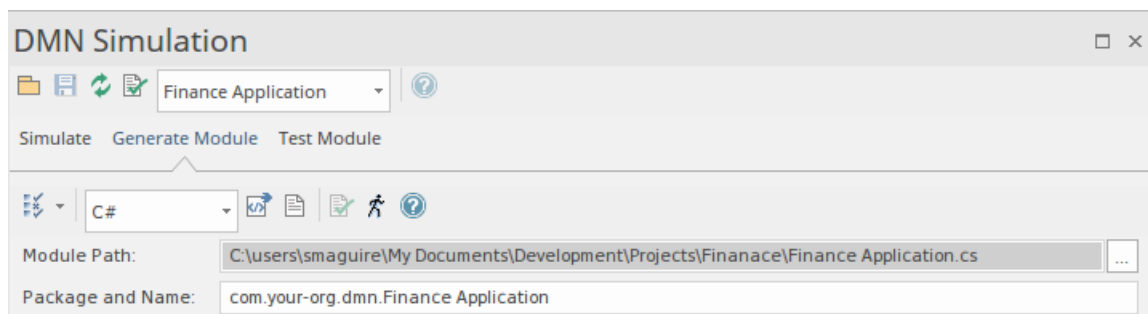


测试决策模型

模型最引人注目的一点是能够在模型实施或投入生产之前对其进行测试。如本指南前面所述，错误成本很高，一些错误会导致灾难性后果，例如在航空或安全系统中。Enterprise Architect决策的建模特征模型允许以模拟的形式进行测试，该工具还具有许多有用的测试特征，可以创建测试用例并记录测试运行。

生成编程代码

在Enterprise Architect中使用决策模型和表示法最引人注目的优势之一是能够直接从逻辑级别自动执行决策，从而允许将决策表、表达式和其他规则定义生成为多种编程语言。这一特征将决策建模推进到数字时代，并填补了业务和技术团队之间存在的深层裂痕。实施团队不再需要在特定域中使用基于文档的业务规则定义，而是接收经过测试并指定到所需细节级别的完整工作模型。此外，业务团队不再需要创建上述耗时且通常在墨水未干之前就已过时的文档。技术团队有时会出于效率目的对规则进行一些改进或协助处理困难的逻辑，但通常会在业务部门的批准下完成。在这些情况下，可以使用相同的数据集重新运行模拟，以确保结果符合企业所有者的预期。

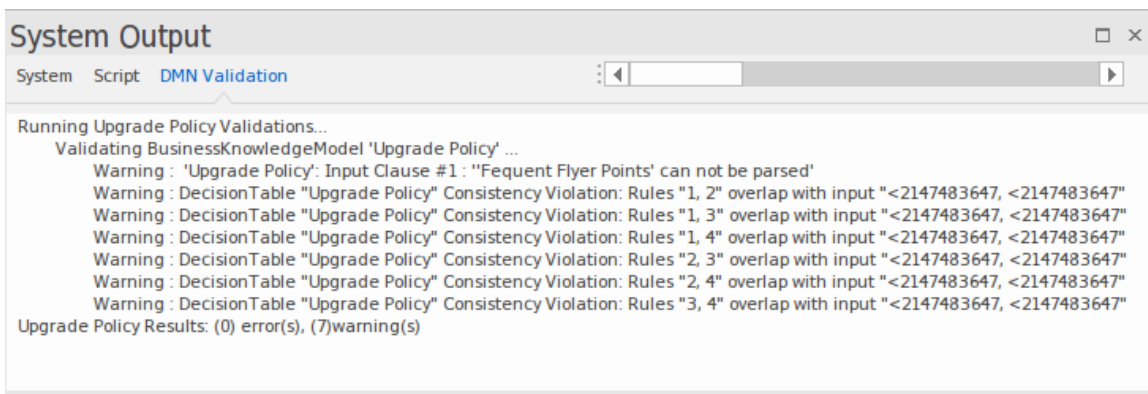


选择一致性和正确性

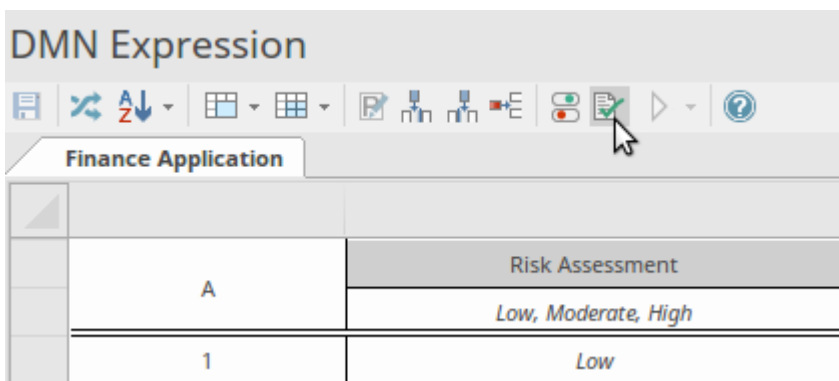
即使经过精心计划和执行的测试，以软件为中心的系统中的许多故障也是在运行时由用户发现的。决策和业务规则也是如此，直到错误的信息被发送给错误的客户或安全设备允许访问本应被阻止的请求，我们才知道出了问题。

决策和描述它们的逻辑通常很复杂，人类难以应对。这种情况更加复杂，因为决策通常在图表中连接在一起，从而形成一个交织的结构，其中一个决策依赖于其他三个决策，而其他三个决策又依赖于其他决策，依此类推。

DMN 标准的Enterprise Architect实施使可视化和协作变得容易，共同定义和自动化一组决策。可以在决策表中体现的表达式编辑器中定义和显示规则，该表达式编辑器类似于电子表格，业务分析师和其他业务利益相关者将熟悉并放心。可以检查这些库表中定义的规则的一致性和正确性，提醒分析师注意缺失或重叠条件的问题和其他问题。



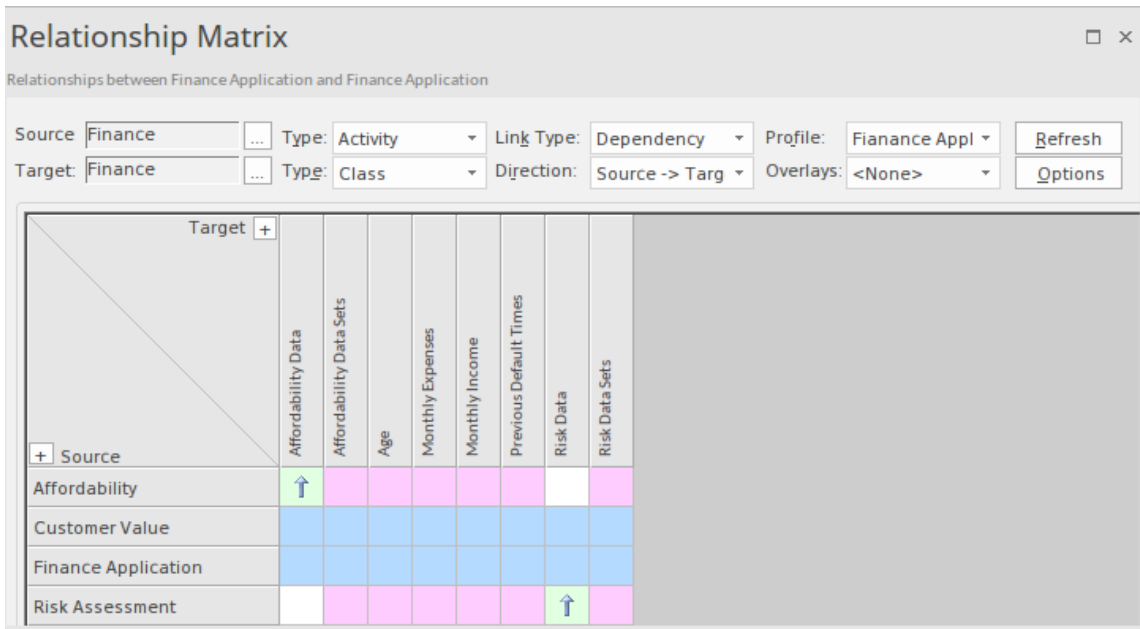
执行验证的工具方便地内置在决策表窗口中，这意味着问题可以在流程的早期解决，早在它们被写入编程代码并到达客户之前。



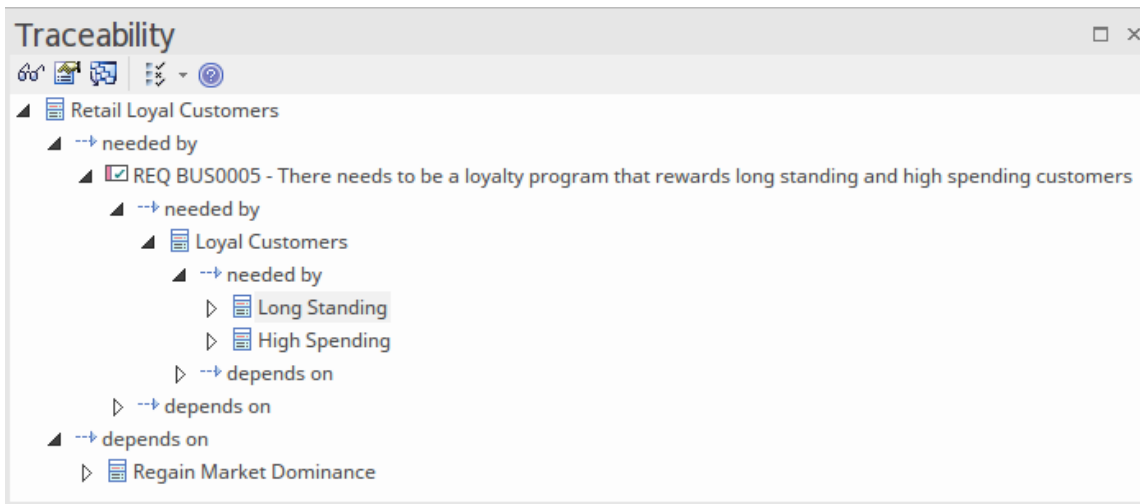
发现规格差距

创建模型是从复杂规范或问题中删除细节的有用方法，它允许架构师、业务分析师和其他利益相关者聚焦问题或解决方案的重要方面。将组织或倡议的规则和决策提炼成单个模型，可以将它们连接到网络（数学图）中，从而立即使它们的交互方式可视化。它还提供了一种机制来定位差距或缺失的决策、输入、业务和权威参考。

Enterprise Architect有许多工具可用于查找模型中的空白或缺失元素。结果展示了建模的力量，即在问题或缺陷被纳入解决方案之前发现问题或缺陷，并且已实施解决方案的用户发现它们。整改可能既费钱又费时，对品牌的潜在损害或失去客户信任可能会造成商业破坏或在安全关键领域无法挽回。

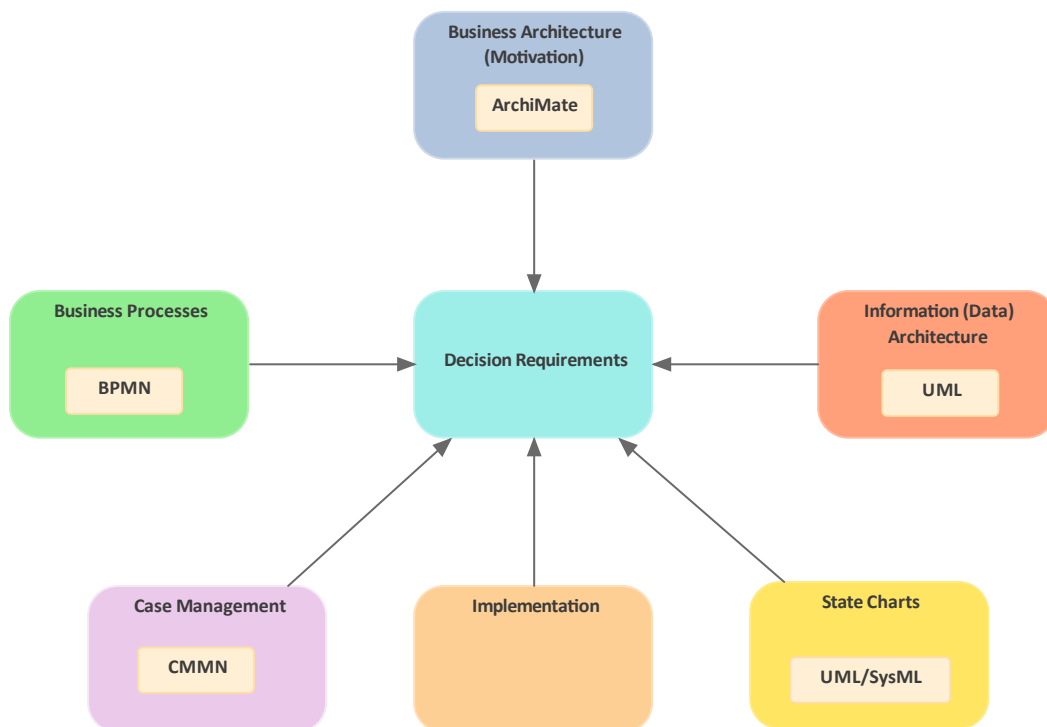


此图显示了如何使用关系矩阵通过没有任何数据输入的情况下可视化决策来快速定位决策图中的差距。该矩阵可用于查找任何其他相关元素之间的差距或重复，例如没有相应知识权威的业务模型。可追溯性和关系窗口显示了将决策与需求等其他元素可视化的另一种方式。这两个窗口都允许您定位浏览器窗口中的元素以及包含对象的任何图表。



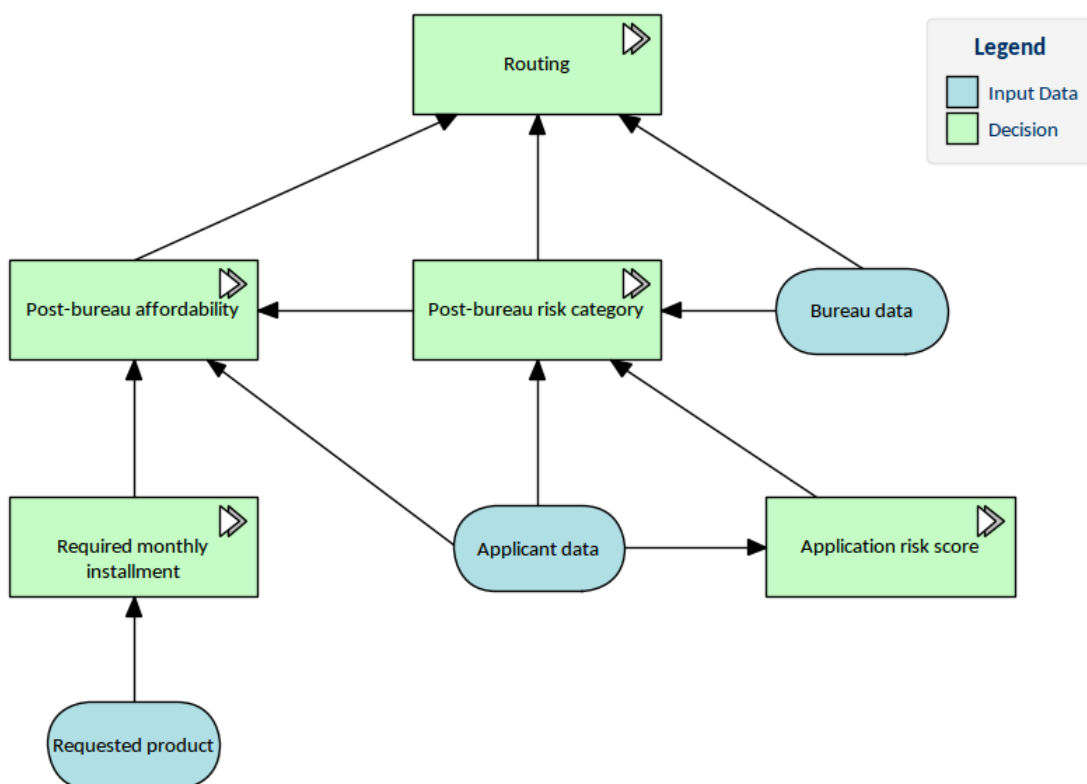
与其他模型一起表达

Enterprise Architect的真正强大之处在于它能够关联来自广泛学科的模式，从而整合战略、想法、知识、应用程序和技术。该工具之所以能够做到这一点，是因为其特征的深度和广度，从战略地图到低级工程图，如可执行状态机和代码图。这些相同的多维特征适用于决策建模，但其强大之处在于将所有模型集成到模型模型中。例如，策略可以与包括驱动因素和目标在内的决策相关联，并一直到将驱动驱动决策的运行引擎的实现代码；一切都可以在Enterprise Architect中建模。



可视化输入和分析

决策被误解或决策引擎或手动或自动化流程的输出通常令人惊讶的A常见原因是因为它们在实施之前没有经过测试。该过程的一个重要部分是对决策的输入进行模型和分析的能力，记住决策可以被建模为层次结构，其中决策充当其他决策的输入，直至任何级别。Enterprise Architect允许将这些输入可视化，并提供一种机制来创建任意数量的数据集，这将有助于测试规范示例以及边界和异常数据点。



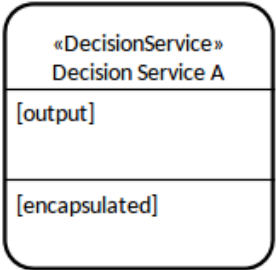

符号简介

决策需求图是图A图形表示，它使用图中元素和连接器之间的关系来描述决策模型。连接器在规范中称为需求，但在图中表示为用于将元素连接到图形中的线。此处的库表列出了 DRD 工具箱页面中可用的元素和连接器，并且可以放置在图表上以创建模型的决策需求级别。实际上，这是通常由业务分析师、架构师或其他业务人员创建和维护的级别。

元素决策模型




这些元素是二维形状，用于对决策需求图的各个部分进行模型，它们可从 DRD 工具箱中获得。元素是构成有向图（有向图）的节点，通过关系（简称需求）以线的形式连接起来。这些元素用于定义从决策本身到知识源的一系列概念。

部件	描述
<p>决策</p>  <p>Decision A</p>	<p>决策表示使用可以引用A或多个业务知识模型的决策逻辑从多个输入中确定输出的行为。决策可以通过多种方式表示，A决策决策表、调用、上下文或文字表达。</p>
<p>业务知识模型</p>  <p>Bussiness Knowledge Model A</p>	<p>业务知识模型表示封装函数A功能，例如业务规则、决策表或分析模型。它作为一个可重用的组件，可以存储在库中，并且可以包含在任意数量的决策模型中。</p>
<p>输入数据</p>  <p>Input Data A</p>	<p>输入数据元素表示用作一个或多个决策的输入的信息。当包含在知识模型中时，它表示知识模型的参数。输入数据元素中定义的信息可以是结构化的。</p>
<p>知识源</p>  <p>Knowledge Source A</p>	<p>知识源表示业务知识模型或决策A权威。信息在决策模型，它们的影响是强制（法规或法律）、控制（政策）、指导（最佳实践）、影响（推荐）的连续统一体。知识源中的信息在形式上与文档、网页、印刷材料、视频或音频内容有很大不同。</p>
<p>决策展开的</p>	<p>决策服务（扩展）表示A组可重用的决策，并用作可调用元素，通过调用逻辑连接到其他元素的知识需求连接器。它提供了一种将决策模型的一部分打包成基于组件或服务的架构的机制，并提供了一个接口，用于指定所需的信息输入和结果输出。使用这种扩展形式，建模者可以显示服务的详细信息，包括其封装和输出隔间。</p>

	
<p>决策服务折叠</p> 	<p>决策服务（折叠）表示A组可重用的决策，并用作可调元素，通过调用逻辑通过知识需求连接器连接到其他元素。它提供了一种将决策模型的一部分打包成基于组件或服务的架构的机制，并提供了一个接口，用于指定所需的信息输入和结果输出。使用这种折叠形式，建模者可以隐藏服务的详细信息，包括其封装和输出隔间。</p>

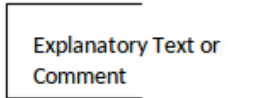
决策模型的需求（关系）


决策模型和符号标准定义了三个关系，可用于连接决策需求图中的组件。这些关系是有向的，它们的应用创建了一个连接模型各个组件的有向图（有向图）。使用了两种线型（实线、虚线）和三个连接器结束标记（封闭箭头、开放箭头和实心圆），在本表中进行了描述。

<p>信息需求</p> 	<p>信息需求表示输入数据或决策输出被用作决策的输入之一。因此，它指定了用于确定输出的决策所消耗和处理的数据。它表示为带有实线箭头的实线。</p>
<p>知识需求</p> 	<p>知识需求表示调用模型A这表明决策、决策服务或其他业务知识模型调用业务知识模型以接收其输出。正是这种机制有效地允许业务知识模型在不同的模型和上下文中被重用。</p>
<p>权威需求</p> 	<p>权威要求表示决策需求图表（DRD）元素对作为指导或知识源的另一个DRD元素的依赖。这是一种有用的机制，可用于解释信息的来源，这些信息用于确定做出引用的元素。</p>

工件

这里列出了两种类型的工件，作为规范一部分的工件和另外两个元素；其中一个规范但未在符号列表中看到，另一个是Enterprise Architect工件模型的模拟部分，用于直观地配置模型。

<p>文本注释</p> 	<p>文本注释为建模者提供了A向模型添加解释性文本或注释的机制。这些注释没有任何建模语义，但提供与整个图表或特定元素集相关的非正式信息。它们可以浮动在图表中或通过关联附加到一个或多个模型元素以指示其适用性。</p>
---	---

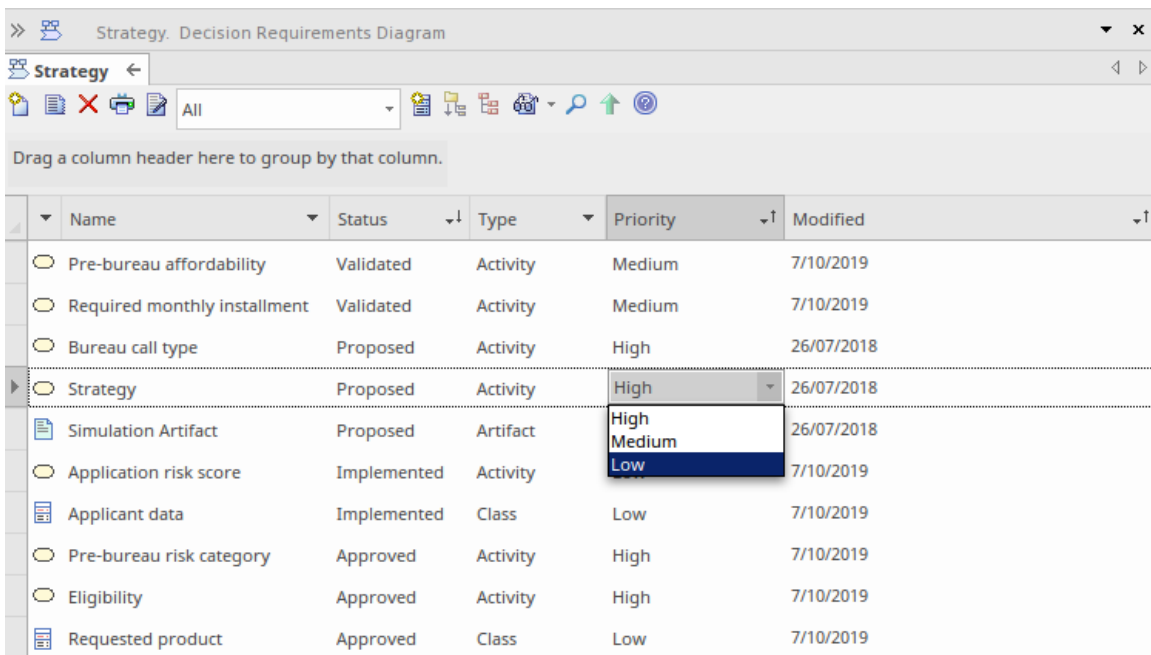
<p>关联</p> <p>-----</p>	<p>关联是将文本注释链接到决策模型图中的一个或多个组件的连接器。除此之外，它没有任何其他语义。</p>
<p>项目定义</p> <div data-bbox="172 376 456 533" style="border: 1px solid black; padding: 5px;"> <p>«ItemDefinition» Item Definition A</p> </div>	<p>项目定义用于模型输入数据的结构和值的范围以及决策的结果，使用诸如 FEEL 或 XML Schema 之类的类型语言。项目定义用于定义输入数据的结构，并可选择限制数据允许值的范围。项目定义的范围可以从简单的单一类型到复杂的结构化类型。项目定义元素的核心属性通过 DMN 表达式窗口访问。</p>
<p>仿真配置</p> <div data-bbox="172 633 456 734" style="border: 1px solid black; padding: 5px;"> <p>«DMNSimConfiguration»  DMN Simulation Configuration A</p> </div>	<p>工件模拟配置是一种用于指定模拟配置的产品。它允许包含要指定的决策和用于输入数据的值集的包。它不是 DMN 规范的一部分，而是 Enterprise Architect 建模环境的一部分，用于提供可视化机制来配置模拟。该工件必须包含要模拟的决策。</p>

决策需求图表

在开始主题中，我们介绍了在Enterprise Architect中使用图表的基本概念，并解释了如何创建决策需求图表 (DRD)、向图表添加许多元素并将元素与关系联系起来。DMN 是一种可视化语言，图表是您使用决策模型的主要方式，包括创建、更新和模拟它们。然而，重要的是要了解图表只是底层存储库的视图，并且它用于与给定的受众进行交流，并且作者可能会故意选择不显示所有相关元素。例如，在与业务利益相关者沟通时，显示与业务知识模型相关的知识源可能很重要，因此可以创建显示这些关系的图表。在与技术受众交流时，作者可能会认为这些价值有限，并创建另一个不显示知识源的图表。这两个图将共享许多元素，并且更新一个图上元素的属性也意味着它们会在它们存在的所有其他图上更新。在下一节中，我们将了解一些用于处理图表的功能。

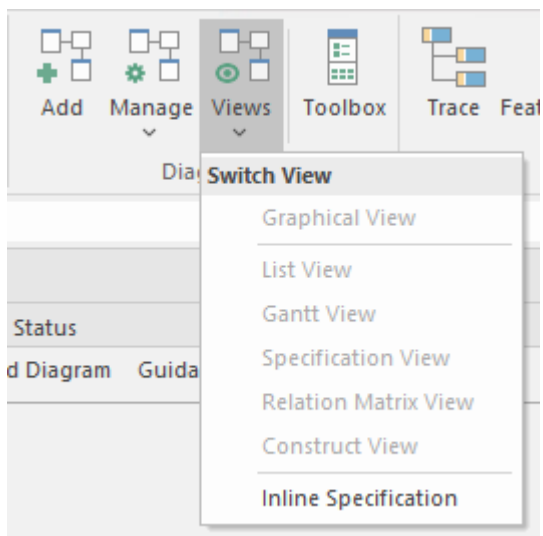
选择视图

尽管图表是使用一组元素的最常用方式，但该工具提供了一系列其他方式来查看包或图表中的元素。这些提供了极大的灵活性，在特定情况下很有用，并且对于那些发现使用其他演示模式（例如列表）更容易的人也很有帮助。



这些选项可从图表的上下文（右键单击）菜单以及此功能区选项中获得：

功能区：设计>图表>视图



其中最有用之一的是列表视图，它以列表的形式显示图表元素，并允许像在电子表格中一样可视化和编辑它们的属性。这对于状态、版本和作者等属性特别有用。属性A一些属性，但可以使用标题上下文（右键单击）菜单选择和添加其他属性，包括标记值。



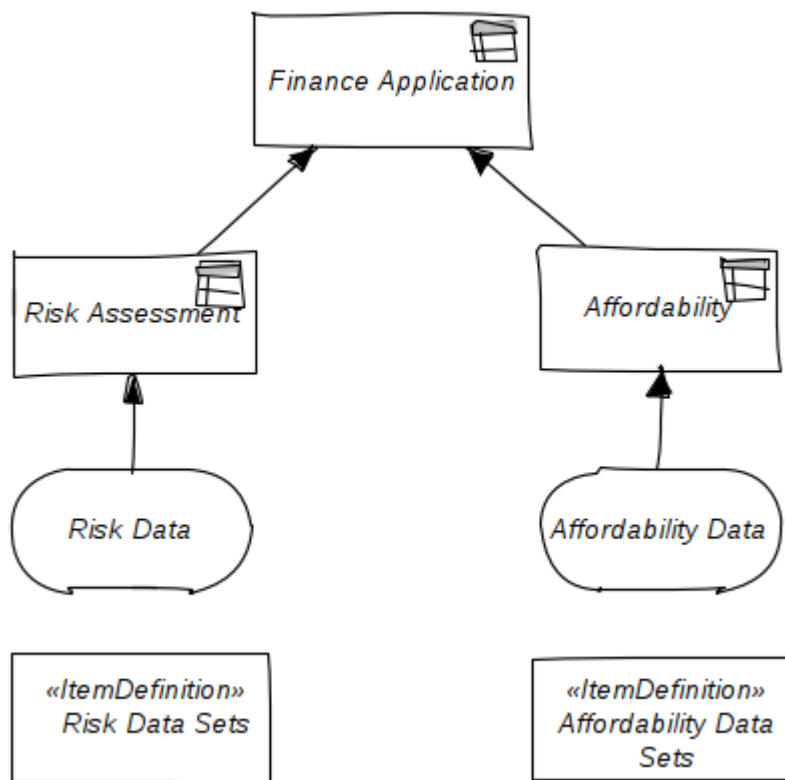
可视化图表或包中元素的另一种独特方式是规范视图。这既呈现为文字处理器，又呈现为电子表格视图，为那些更熟悉这些类型的办公自动化工具的业务或技术建模者提供了舒适的体验。

当资源已分配给元素时，还有两个有用的视图——甘特视图和构建视图。甘特图显示了一个有用的视图，可以以资源或object为中心。

模式和风格

图表以多种不同的模式查看图表，这些模式提供了在各种业务环境中呈现图表信息的有用方式。可以将三种模式或样式应用于图表。

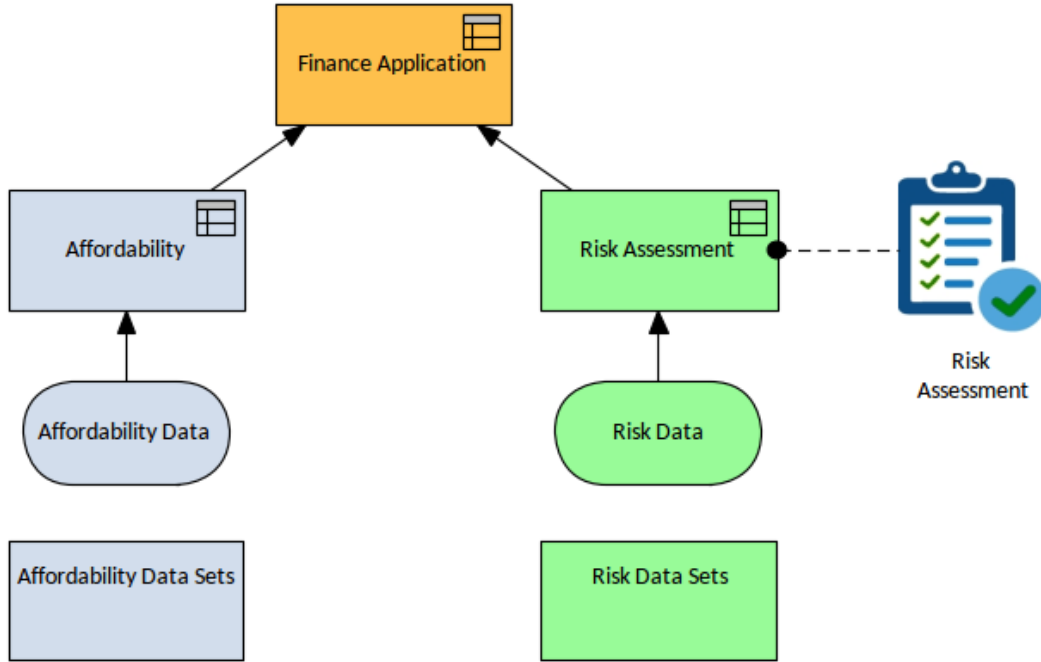
- 手绘模式- 应用元素已手绘的效果
- 白板模式- 应用手绘效果并将元素填充颜色更改为白色，就像在白板上绘制图表一样
- 自定义样式- 在标签位置和旋转、透明度等方面提供极大的灵活性，以创建视觉上引人注目的图表



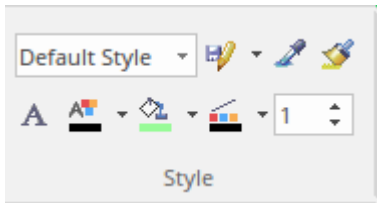
功能布局功能区

当图表结构良好且布局图表时，图表会产生最大的影响，Enterprise Architect提供了大量工具来创建引人注目且高度精美的图表，这些图表非常适合向高管层、业务和技术利益相关者等人展示。我们已经了解了图表主题、模式和样式，现在我们将了解如何更改图表上元素的外观。

许多利益相关者更喜欢图表中的图像，添加一些图像内容可以使图表对这些利益相关者更具吸引力和平易近人。使用图像的一个示例是将知识源元素替换为代表组织或源来源的图像。在此图中，我们看到使用代表风险评估的图标；颜色已用于显示高级决策的单独输入，并且图中已抑制了诸如刻板印象之类的一些细节。



除了图表主题和样式，Enterprise Architect还允许为每个单独的元素设置样式。这些样式包括填充、边框和字体颜色，以及线条和边框粗细。这些设置和其他设置可从布局功能区的样式面板和图表object右侧的元素图标中获得。

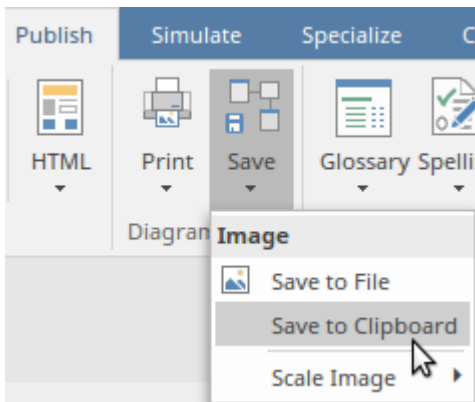


还有一组有用的对齐工具，可应用于选定的一组元素以设置统一的高度和宽度，水平和垂直对齐边缘和中心等。



出口

Enterprise Architect中的模型内容，包括图表，可以与其他用户和工具共享，允许模型内容在其他上下文中可视化。图表可以简单地复制到剪贴板，然后粘贴到任何其他应用程序中，例如幻灯片演示、文字处理器文档、网页或电子邮件。此选项可从“发布”功能区的“图表图像”面板中获得。

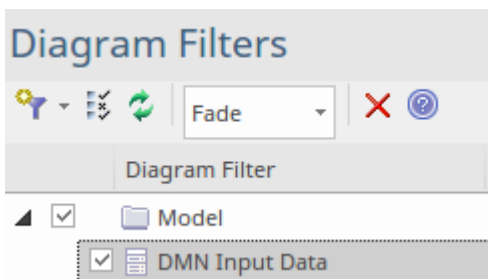


还有一个有用的图表报告，可用于导出包中包含的所有图表，包括子包。图表可以导出为多种格式，包括 gif、bmp、wmf 和决策。决策模型中的元素也可以导出为 CSV 文件，以便在电子表格应用程序中打开。

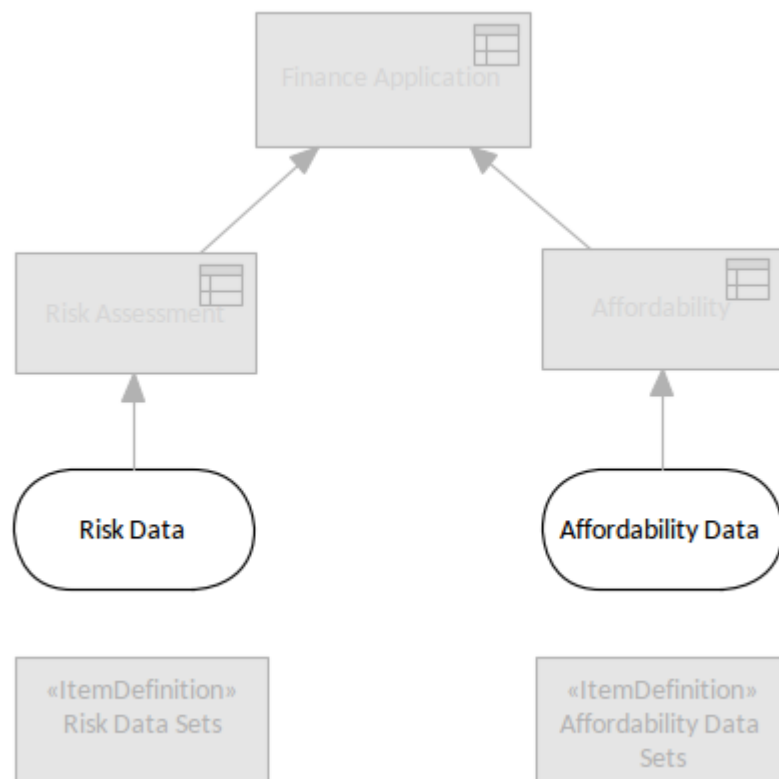
当需要高质量文档时，Enterprise Architect 可用于使用许多预定义或自定义模板生成一流的出版物。模板引擎提供了大量配置点，并允许导入组织的公司样式，包括带有图像的封面页，这些图像将使最终文档吸引战略和业务利益相关者。

过滤器

图表过滤器是一种方便的聚焦设备，其中图表中的元素可以隐藏或以褪色或改变的样式表示，以提高其余元素的突出度，这些元素是图表的聚焦。过滤器可以切换并且是演示或出版物的有用设备，因此观众可以可视化作者想要引起他们注意的元素。过滤器可以基于搜索并且可以动态地应用于具有相同效果的任何图表。



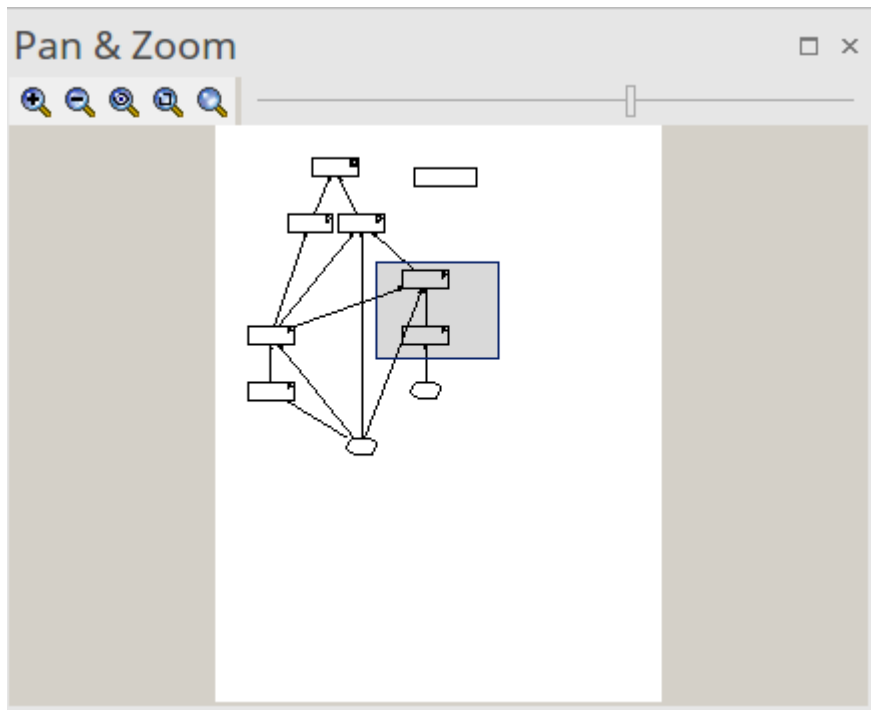
此图显示了应用此过滤器的效果；注意选择的“效果”是淡入淡出，这意味着不是输入数据类型的图表元素将在图表中淡出，只有输入数据元素将保持全强度。其它效果可以选择为可视化元素提供有用的选项，具体取决于关于如何查看图表和观众在场。



有关详细信息，请参阅[Visual Filters](#)帮助主题。

平移和缩放

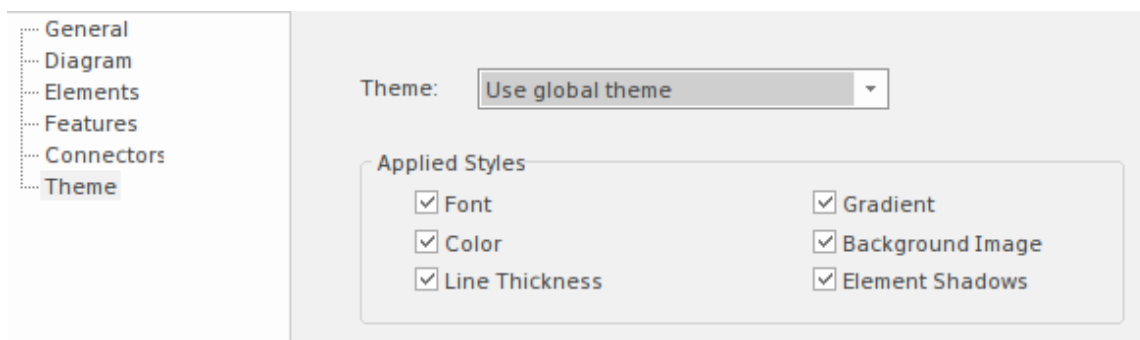
平移和缩放功能是可用于在大型决策图中导航的工具之一。通常必须降低图表的分辨率以确保它完全可见，但是通过使用平移和缩放窗口，您可以使图表保持可读的分辨率并平移到感兴趣的区域，并在必要时放大。即使您有幸使用大型显示器，您也经常希望更改查看图表的比例，然后平移以找到图表中感兴趣的部分或元素，放大该部分以查看更详细的视图。平移和缩放窗口将允许您对任何尺寸的图表执行此操作，其中包含在研讨会或聚焦小组中特别有用的平移和缩放选项，以与可能不熟悉图表的观众讨论模型。



有关详细信息，请参阅[Pan and Zoom](#)主题。

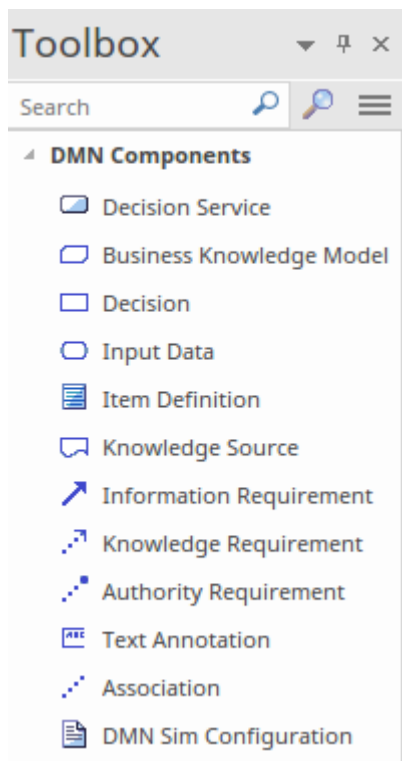
主题颜色和样式

颜色的使用是一个重要的视觉提示，如果谨慎且一致地使用，可以使图表更具吸引力和视觉吸引力。从可以在全局级别设置的图表主题开始，有一系列可用的工具，但可以为单个图表覆盖此设置。也可以覆盖元素外观的默认设置，包括：它的大小、字体、背景和线条颜色。字体类型也可以为单个元素设置。元素也可以由图像表示，作为默认值元素出现的每个图表或仅针对指定的图表。



工具箱

您会记得，当您创建新的决策图或打开现有的决策图时，Enterprise Architect将在主画布中显示该图，并显示DMN 组件工具箱页面，该页面提供了一系列项目，包括元素、关系（需求）和工件，可以拖放到图表上。



虽然 DMN 组件工具箱页面中包含的元素和连接器是建模者创建决策需求图 (DRD) 所需的项目，但 DRD 中可以包含任意数量的其他元素，或者可以将任意数量的元素复制到另一个可以修饰的图表。这在本指南的前面部分进行了描述，其中决策与其他战略要素（例如目标和目的）相关联。

表达式决策表达式类型

决策建模和符号的逻辑级别使用表达式，允许决策建模者使用一系列表达式、语言及其预定义函数来构造将输入值与输出值相关联的语句。每个决策模型的业务、工程或科学上下文可能会有很大差异，正如决策需求模型在这些不同的上下文中需要不同一样，决策逻辑级别的表达方式也需要不同。正是为了适应这些不同的情况，规范提供了四种不同的表达逻辑的表达式方式，由建模者选择最合适的一种来使用。四种值表达式类型是：文字表达式、盒装上下文、调用和决策表。决策表无疑是最灵活和最普遍适用的表达式类型，虽然在这里将讨论它们，但本指南中的整个主题都致力于如何使用它们。

在接下来的部分中，我们将提供一些关于在给定时间使用哪种表达式类型的上下文，但在许多方面，这些类型的呈现顺序反映了它们的适用性和复杂性。Enterprise Architect支持所有四种表达式类型，并添加了超出规范中规定的一系列附加特征，这些特征将使建模者富有成效并有助于确保决策模型提供巨大的业务、工程或科学价值。

我们还将提供有关表达语言的一些信息，包括F riendly E nough E xpression L语言，或简称FEEL。

值表达式类型

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。
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。
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。
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。

决策表



表 决策元素或业务（决策）模型元素右上角的这个图标，表示它是作为决策决策表来实现的。决策表是最常用的表达类型，因为它采用表格格式，大家都很熟悉适用于通常使用电子表格和其他表格表示的业务经理和其他非技术人员。从根本上说，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。可以水平或垂直组织的规则指示哪个输出条目适用于一组特定的输入条目。例如，我们可能有两条规则，一条规定如果客户拥有“黄金”信用评级并且每年花费 20,000 美元或更多，他们将获得 20% 的折扣，第二条规则说如果他们拥有“Gold”的信用评级和消费低于 20,000 美元他们将获得 15% 的折扣。当决策决策表投入生产时，每次执行决策时都会提供输入值，并且根据这些值，客户将获得 15% 或 20% 的折扣。

DMN Expression

Application risk score model Input Parameter Values for Simulation

(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	

该图显示了用于定义决策表的DMN 表达式窗口的关键部分。回想一下，还有其他三种类型的值表达式，并且每种类型的窗口都会以不同的方式出现。此列表包含决策表的主要部分：

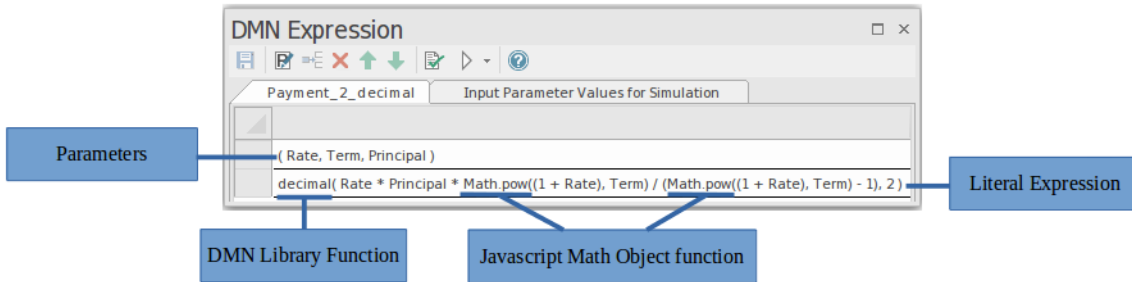
- 规则A，其中每个规则包含特定的输入条目和相应的输出条目
- 输入子句A，定义为通常涉及一个或多个输入值的表达式
- 输出子句A，定义对应于一组特定输入的输出
- 指定如何应用规则的库表命中策略

输入子句由一个表达式和一个可选的允许值列表组成。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”。允许的值适用于表达式结果，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。库表本身包含一个编号规则列表，定义了一组输入条目和相应的输出条目。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。

文字表达

=X 决策或业务模型 (BKM) 元素右上角的这个图标表示它是作为 *Literal Expression* 实现的。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。当表达式变得更复杂时，建模者可能会选择封装上下文而不是文字表达式，或者为了提高可读性，建模者可以将一些逻辑封装为 DMN 库中的函数。文字表达式是决策和业务知识模型 (决策) 元素中使用的一种价值表达。



此图显示了用于定义文字表达式的DMN 表达式窗口的关键部分。回想一下，还有其他三种类型的值表达式，并且每种类型的窗口都会以不同的方式出现。


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。

DMN 表达式窗口将 *Literal Expression* 显示为一个表，其中包含两个关键行：

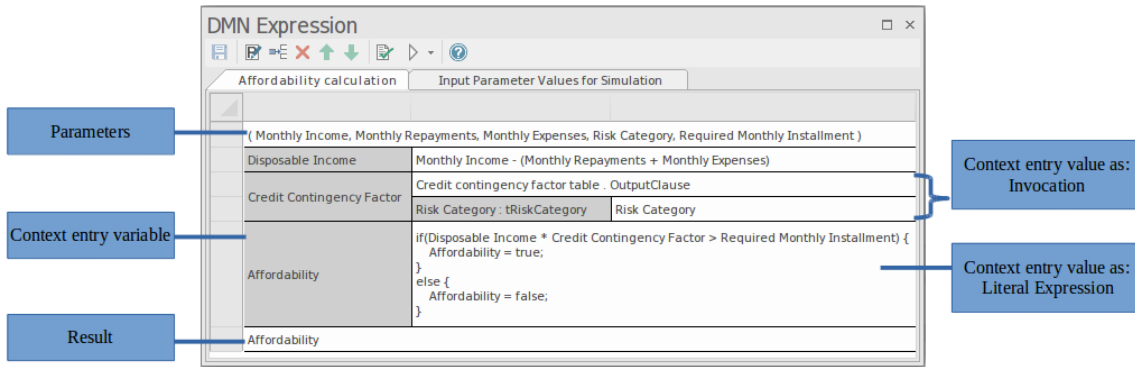
- 参数：定义表达式中使用的输入参数
- 文字表达式：定义表达式的公式的地方 - 这定义了决策的输出

为了支持模拟和执行，*Literal Expression* 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。

盒装上下文

 决策业务 (BKM) 模型元素右上角的这个图标表示它是作为盒装内容实现的。Boxed 表达式用于文字表达式会导致复杂且难以理解的逻辑的情况。与决策表一样，它的表格形式使业务经理和其他非技术利益相关者更容易理解。

基本上，盒装上下文是上下文条目的集合，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，并且可以被认为是可以在任何后续上下文条目的值表达式中使用的中间结果。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。



此图显示了用于定义盒装上下文的DMN 表达式窗口的关键部分。回想一下，还有其他三种类型的值表达式，每种类型的窗口都会以不同的方式出现。

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。盒装上下文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，但它必须能够被评估以提供结果。

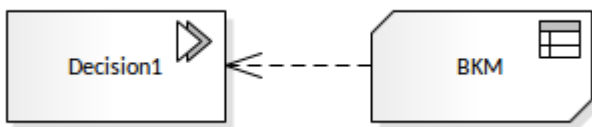
调用

决策或业务知识模型(BKM)元素右上角的这个图标表示它被实现为一个 *Invocation*。调用表达式类型是一种机制，通过它可以在不同的上下文中访问和重用决策表达式。您会记得在之前的主题中我们讨论了决策建模符号并引入了业务知识模型(BKM) 作为重用预定义决策逻辑的方法。调用表达式类型是用于访问 BKM 中的决策逻辑的方法，提供传递给 BKM 或决策服务参数的值，然后接收输出。调用可以应用于决策元素和 BKM 元素，它们都将调用业务知识模型或决策服务。

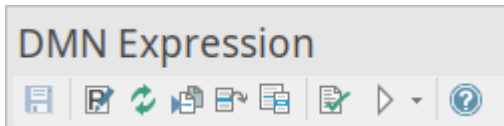
调用是参数绑定的容器，为业务知识模型的主体评估提供上下文。调用有两种常见的用例：

- 用于将输入数据绑定到业务知识模型
- 用于绑定参数或上下文入口变量到业务知识模型

调用是一种表格表示形式，表示在可调用元素（业务知识模型或决策服务）中定义的决策逻辑如何被决策或另一个业务知识模型调用。



When an Invocation is selected, the 布局of特征accessible in the DMN 表达式window is:



有关更多详细信息，请参阅调用编辑器工具栏帮助主题。

Invocation 的参数绑定提供了用于评估可调用元素主体的上下文。

U	Existing Customer	Application Risk Score	Credit Score	Post Bureau Risk Category
	true	90	590	MEDIUM
1	-	<120	<590	HIGH
2	-	<120	[590..610]	MEDIUM
3	-	<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

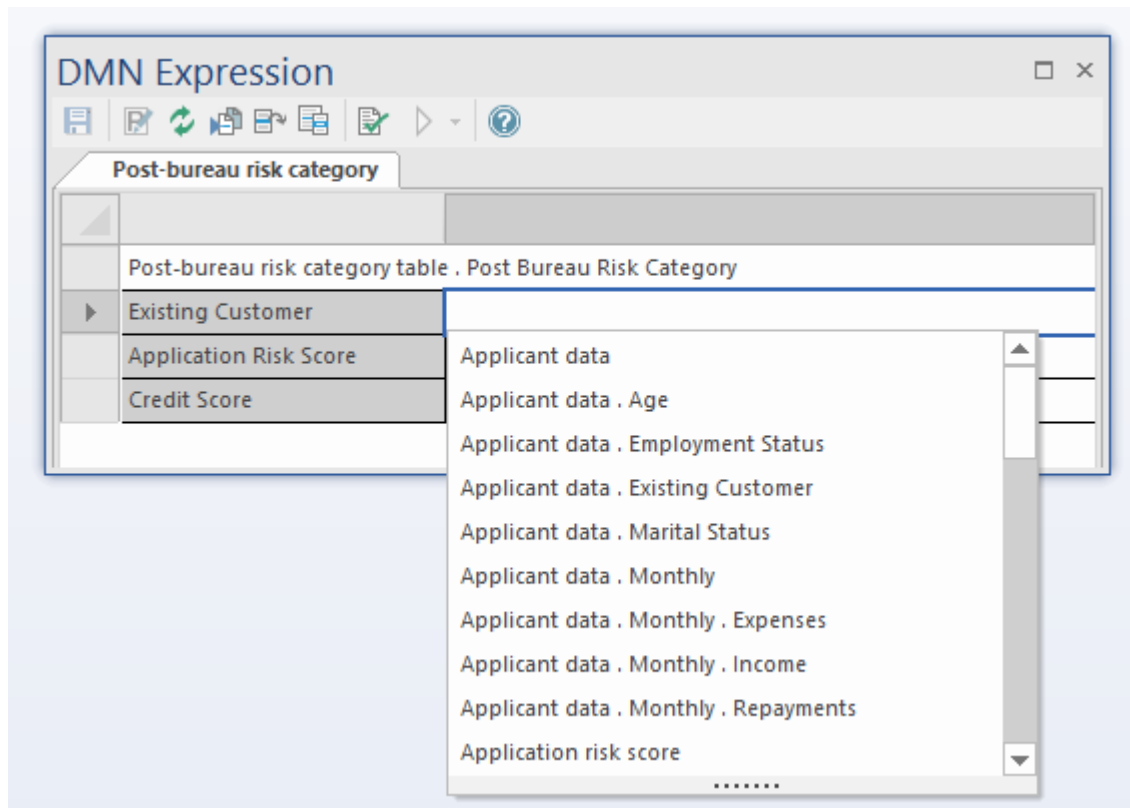
Post-bureau risk category table . Post Bureau Risk Category	
Existing Customer	Applicant data . Existing Customer
Application Risk Sc...	Application risk score
Credit Score	Bureau data . CreditScore

在这个例子中：

- 决策'Post-bureau risk category'表示为连接到业务知识模型“局后风险类别表”的调用，实现为决策表
- 决策'Post-bureau risk category'是来自两个输入数据和一个决策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”；如果决策表定义了多个输出子句，则调用必须明确请求一个输出子句

作为表达式的结果

可以通过按字段中的空格键来设置来自其他 Decisions 和 InputData 元素的输入。



由于调用只能调用一个业务知识模型因此输出由业务知识模型输出定义。

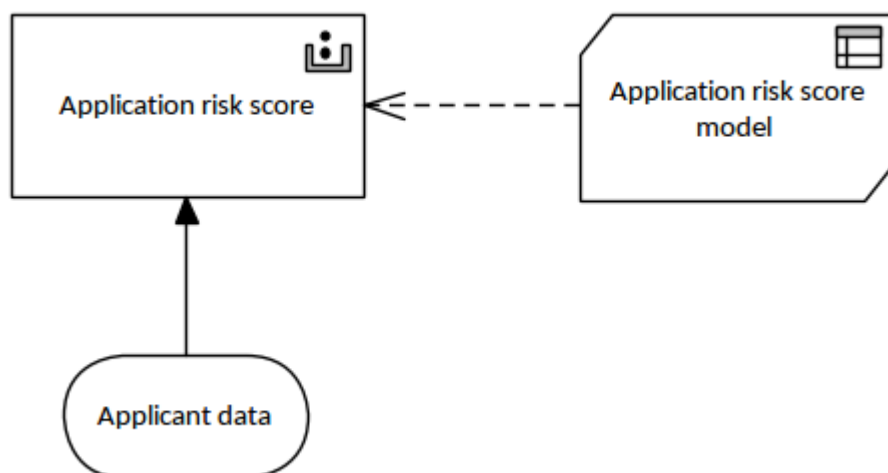
表达语言

这就是橡胶上路的地方，在决策需求图级别上完成的所有工作都需要用表达决策逻辑的表达式进行修饰。与设计的许多方面一样，事情开始很简单，但不可避免地会变得相当复杂。在决策模型定义的这一点上，重要的是要记住模型的目的是确保将决策分解为对业务、工程或科学利益相关者有意义的可理解块。对于许多应用程序来说，逻辑和附带的表达式对于大多数利益相关者来说都足够简单理解。各种表达式类型中使用的值表达式用**F**riendly **E**nough **E**xpression **L**anguage或简称 FEEL 编写。细节将详细介绍该语言的基本方面以及企业架构师代码编辑器和验证器如何支持它。

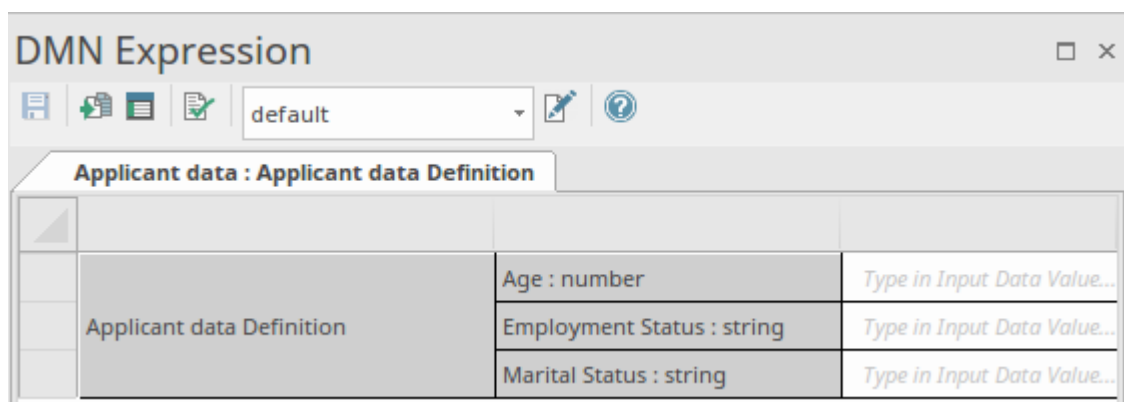
除了 FEEL 语言结构 (例如 `not([10..35])`、日期和时间 (`"2022-02-04T07:42:00"`))。

表达式编辑器

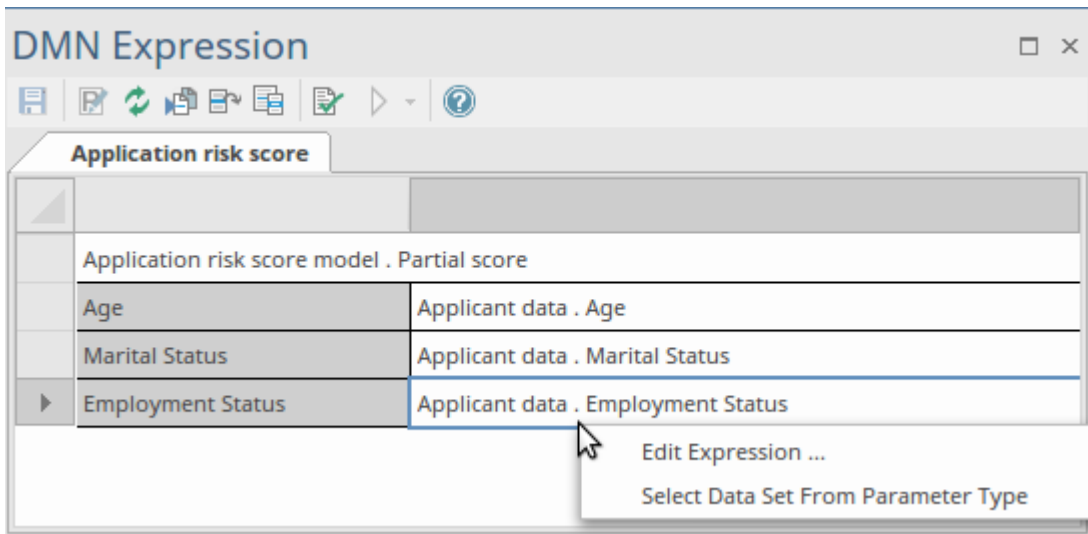
表达式编辑器是一种生产力工具，可用于在DMN 表达式窗口中定义逻辑时创建表达式。此图显示了一个简单模型，其中包含使用业务知识模型元素的 Boxed Context 表达式类型的决策。我们将看看如何使用 intellisense 否则知道作为代码完成



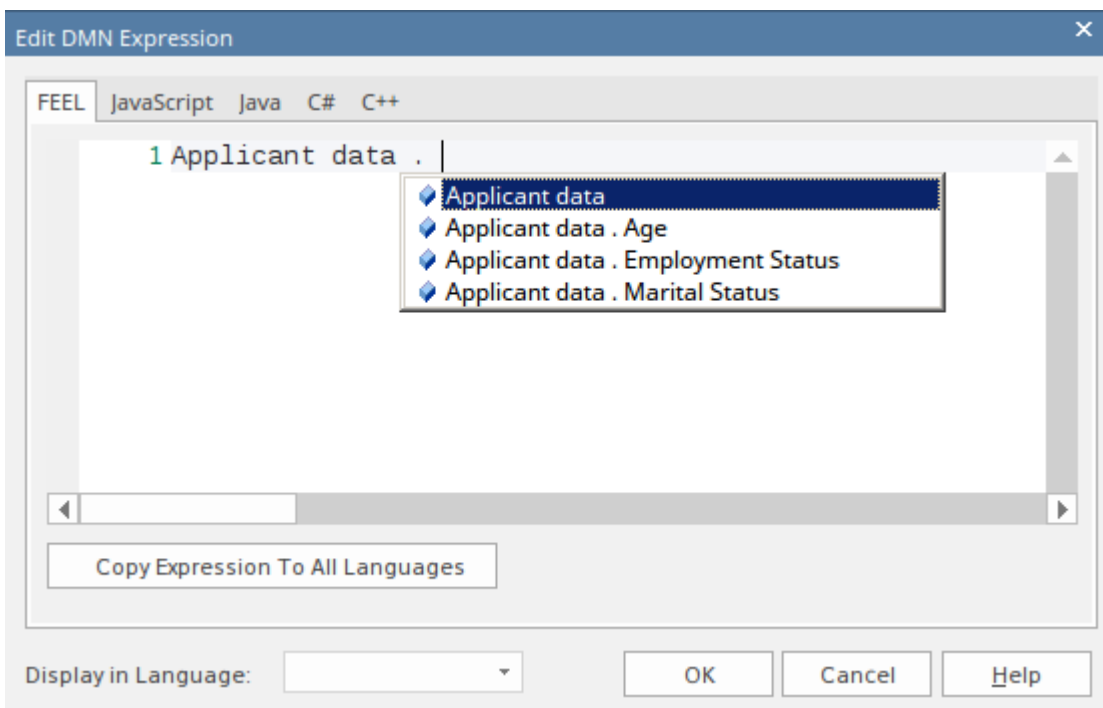
表达式编辑器对话框用于在盒装内容、调用和文字表达式元素类型中设置表达式。它提供智能感知支持，用于构建基于 FEEL 语法的表达式以及可用于模型生成的代码语言。



此图显示了用于定义输入数据的多用途表达式窗口，正如您将从图中看到的那样，为风险决策提供输入数据。我们将在接下来的几个步骤中看到Enterprise Architect如何通过智能感知使表达式编辑器可以使用这些数据项。下图显示了如何启动表达式编辑器。



打开编辑器后，详细说明表达式和逻辑的建模者可以利用此生产力工具并通过使用智能感知访问数据项来减少逻辑错误，从而使输入数据在列表中可用。



足够友好表达语言(FEEL)

最初的足够友好表达语言 (FEEL) 是为了支持决策建模注释而创建的，并在规范中进行了严格定义。它是一种轻量级语言，专注于创建具有足够数据类型表达式和语法的表达式，使其能够描述决策。一些技术含量更高的读者会质疑，当现有的语言 (如JavaScript) 适合其用途时，为什么还需要另一种语言。这个问题的答案似乎是为了让非技术人员能够理解表达并为对话做出贡献的足够简单。像任何语言一样，它的语法必须遵守，否则它将是格式错误的并且不会被编译器接受，或者更糟的是会产生错误或不可预测的结果。

Enterprise Architect具有从决策模型生成实现 (编程) 代码的功能，在 FEEL 中定义的表达式可以自动转换为多种不同的目标语言。所以没有误解让我们清楚 - FEEL 是一种表达语言，它本身不是一种实现语言。

数据类型

计算机科学中使用的几乎所有语言都具有数据类型，旨在帮助程序员或分析师正确指定他们的意图，并确保编译器或解释器接收指定格式的输入。数据类型是限制表达式 (例如变量或函数) 可以采用的值的机制。数据类型规定了可以对数据执行的操作、数据的含义以及可以存储该类型值的方式。FEEL 具有本表中指定的四种数据类型。

FEEL数据类型

数据类型	描述
数字	数字基于 IEEE 754-2008 标准中指定的十进制 128 格式。所以它们可以是整数或小数。例如123、2089、0.005
字符串	字符串被定义为字符的双引号序列，例如 "Gold"、"Approved"、"Alert"序列。
布尔值	布尔值被定义为逻辑对 True或False，例如 Flase
日期、时间和持续时间	有日期、时间、日期时间、日和时间的持续时间。这些包括日历、日期和持续时间。例如日期 ("2022-04-06")，日期和时间 ("2022-04-06T07:42:00")

语法表达式

表达式用于定义逻辑并包含定义运算符和参数以及返回值的名称和 FEEL 表达式。语法规则有四类：算法、比较、间隔和附加。我们现在将在细节中查看这些内容。

算法

算术表达式是一些最常用的语法规则，只需应用我们都熟悉的算术运算符，包括指数 (x**n) 来将数字 x 提高到 n 次方。

加法 (+)	75 + 2.75 (计算为 77.25)
--------	-------------------------

减法 (-)	$100 - 24.5$ (评估为 75.5)
乘法 (*)	$50 * 5$ (计算为 250)
分配 (/)	$144 / 6$ (计算为 24)
求幂 (**)	$2 ** 6$ (计算为 64)
一元减号	$- 200$ (计算为 -200)

比较

当我们想对程度做出决定时，比较运算符很有用。表达式中总是包含两个操作数，中间的操作符以 x 操作符 y 的形式出现。

名称	示例
相等 (=)	$24 = 8 * 3$ (计算为 <i>True</i>)
大于 (>)	$24 > 2 ** 5$ (评估为 <i>False</i>)
大于或等于 (>=)	$21 >= 7 * 3$ (计算为 <i>True</i>)
小于 (<)	$16 < 25$ (评估为 <i>True</i>)
小于或等于 (<=)	$0.25 >= 1 / 4$ (计算为 <i>True</i>)
不等于 (!=)	$25 != 8 ** 3$ (计算为 <i>True</i>)

间隔

区间运算符对于允许在范围中包含或排除上限和下限的数字范围很有用。这些是非常有用的运算符，但必须小心以确保括号（round或方形）不会放错位置，否则会无意中改变含义。验证引擎在这种情况下很有用，可以检测表达式中的意外间隙或重叠。

包容 - 包容	$2 \text{ in } [2 .. 5]$ (计算为 <i>True</i>)
包容 - 独家	$5 \text{ in } [2 .. 5)$ (评估为 <i>False</i>)

独家 - 包含	$5 \text{ in } (2 .. 5]$ (评估为 <i>True</i>)
独家 - 独家	$2 \text{ in } (2 .. 5)$ (计算为 <i>True</i>)

额外的

附加的运算符是逻辑连接、析取和否定运算符，它们在形式逻辑中被广泛理解为 OR、AND、NOT。

连词	$(3 * 4 = 12) \text{ 和 } (3 * 5 = 15)$ (计算为 <i>True</i>)
析取	$(3 * 4 = 12) \text{ 或 } (3 * 5 = 17)$ (计算为 <i>True</i>)
否定	<i>不是</i> $(3 * 5 = 15)$ (评估为 <i>False</i>)

决策表解释

决策表是最常用的表达式之一，并且在定义规则时提供了极大的灵活性，允许逻辑“或”的逻辑表达式，这在许多其他表达式类型中是不可能的。

DMN Expression					
Application risk score model					
Input Parameter Values for Simulation					
(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	

该图展示了决策表的剖析，其部分将在后续章节中讨论。决策表本质上是A组在执行决策时将被评估的规则。每个规则包含任意数量的输入，当执行逻辑并选择规则时，将返回针对该规则定义的输出。给定规则中的输入表达式组合为逻辑“与”，因此在选择规则之前，规则中的每个输入表达式都必须计算为True。

在此示例中，有两个输入列 - 年龄和病史。对于申请者风险因素=中的输出，申请者年龄<25岁并且他们的病史必须是公平的。

Applicant Risk Rating			
U	Age	Medical History	Applicant Risk Rating
	[18..99]	Good, Fair, Bad	Low, Medium, High
1	<25	Good	Low
2	<25	Fair	Medium
3	<25	Bad	-

命中策略

每个决策表必须有一个命中策略，指定如何评估表中定义的规则默认为“U”，表示“唯一”，这意味着规则不能重叠，并且只会选择一个规则。命中策略有返回单行（Unique、Any、Priority、First）和返回多行的命中策略（输出、Rule Order、细节）。命中策略在后面的主题中命中策略。

规则列表

决策表本质上是A组定义输入和输出的规则。规则可以水平或垂直定向，并且可以包含任意数量的输入表达式和一个或多个输出表达式。

输入子句列表

每个输入子句由表的每个规则中定义的表达式组成。表达通常包括一个或多个输入值，例如'Age>25'。

输出子句列表

每个输出子句由一个表达式组成，该表达式可以是一个简单的元素或包含一个或多个输入值，例如“高、中、低”。

决策表工具栏包含许多与库表一起使用的单元功能，包括添加行和列、合并以及验证表的语法和逻辑正确性。有关决策表工具栏的更多详细信息，请参阅 [Toolbar for Decision Table Editor](#) 帮助主题。

允许的值字段

输入表达式以决策表的单元形式编写，通常非常简单，可以是：

- 值，例如 “黄金”或
- 测试，例如 “年龄 < 25”

这些表达式中的值或变量必须通过输入数据提供，这些数据应该在决策需求图中可见。可用或可能的输入值或范围的范围通常会在定义级别进行讨论，这些构成决策表定义的所有重要参考。例如，贷款申请的决定可能依赖于顾客级别的输入，它可以在业务中定义为值的枚举——“金、银、铜”——也可以依赖 定义为的简单年龄值 '18 < 年龄 < 65'。在表中定义规则及其逻辑时包含这些约束非常重要，因为这将使表更容易被人类理解并且更容易为工具验证。如果没有定义这些，建模者可能会假设表已完成，但验证者可能会另外规定。

无论建模者如何定义表达式，输入值都应该是互斥的和完成的，以确保表被正确定义并在运行时产生预期的输出（在实践中实现和使用之后）。

- Exclusive - 表示输入值是不相交的；也就是说，它们不会相互重叠 - 没有重叠
- 完全 - 表示已捕获域中定义的所有相关输入值 - 没有间隙

这些输入约束或限制可以添加到输入和输出列正下方的标题中，在带有水平方向规则的表中。

Finance Application			
		Risk Assessment	Affordability
	A	Low, Moderate, High	[0..10]
	1	Low	>8
	2	Moderate	[5..8]
	3	High	<5

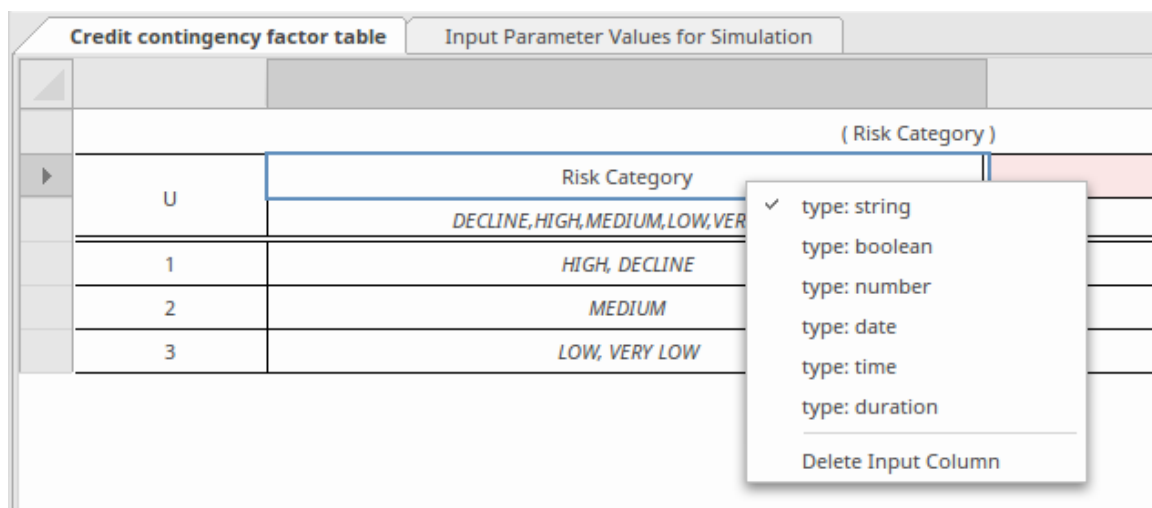
此屏幕截图显示建模者已为财务应用程序决策表的输入列定义了允许值。风险评估输入只允许具有三个值，即高、中和低。A在表中列出了 Minimal 等值，则会导致验证错误。我们将在本指南的后续部分详细介绍验证，但正如我们所讨论的，这是在错误使用“Minimal”的情况下验证功能的示例输出。

```
Running Finance Application Validations ...
Validating Decision 'Finance Application' ...
Warning : DecisionTable "Finance Application" Input Violation:
Input Value is not allowed for "Rule[7].Risk Assessment": "Minimal"
Finance Application Results: (0) error(s), (1)warning(s)
```

输入输出子句的数据类型


为确保模型格式正确、逻辑正确指定以及模拟工作，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，请右键单击输入子句或输出子句，然后从列表中选择所需的类型。



规则和输入输出


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，输入条目可以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，使用方括号表示包括边界数。）输出条目应指定每个单元的单个值。

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。


现有规则可以通过首先选择规则来复制和粘贴到表中，（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，则可以使用空格键从允许值列表选择一个值。

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



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”决定了表行如何产生输出；我们将在下一节中了解每个策略的含义。

命中策略

术语“命中策略”是指在模型执行过程中，决策表中规则的选择方式。决策表通常包含一些规则，在表A左上角有一个框，其中包含一个字母，指示应用了哪些命中策略（以及如何选择规则）。默认情况下，决策表中的规则重叠，但如果规则确实重叠（意味着一个规则可以匹配一组给定的输入值），则需要命中策略指示符才能识别表类型并允许决策逻辑要明确理解。命中策略可以根据匹配规则的数量分为两组：

- 选择A规则（唯一、任意、优先级、第一）
- 选择了多个规则（输出顺序、规则顺序、收集）

Risk Assessment			
		Risk Data . Age	Risk Data . Previous Default Times
	U		
	1		>3
	2		[2..3]
	3		<2
	4		>3
	5		[2..3]
	6		<3
	7		-

U : Unique

A : Any

P : Priority

F : First

O : Output order

R : Rule order

C : Collect - List

C+ : Collect - Sum

C< : Collect - Min

C> : Collect - Max

C# : Collect - Count

不要太在意每一个的含义，因为我们将很快讨论它们，并提出一些基本的方法来选择一种命中策略而不是另一种 - 如果有疑问，请选择唯一（U）策略，因为它是默认策略并且是最常用。Enterprise Architect支持所有定义的命中策略，以及我们将在下一节中学习的功能- 使用命中策略来确定决策表中的规则是否有差距或重叠

选择正确的命中策略对于成功规范决策模型的逻辑级别决策。与建模中的许多其他事情一样，有一些在实践中使用得更频繁的命中策略，因为它们被证明是表达特定决策的最佳方式，另一些不经常使用，而另一些则很少使用。命中策略包括字母代码、名称和描述，如下所示。

单一规则命中策略：

- **U**（U）：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**（A）：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等，因此可以使用任何匹配
- **优先级（P）**：可以匹配多个规则，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
- **第一个（F）**：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（O）：按输出优先级降序返回所有命中
- **规则顺序（R）**：R规则顺序返回所有命中
- **收集（C）**：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

独特的

具有唯一性 (U)命中策略A表定义了一组不重叠的规则，这意味着这些规则是互斥的，或者在正式的集合术语中是不相交的。对于给定的一组输入，一个且只有一个规则将匹配，并且将产生一组适用的输出。因此，这是一个“一击一出”的政策。它无疑是所有命中策略中最常见的，因为它广泛应用于许多逻辑上下文，每个规则都可以独立推理，并且很容易为业务和非技术利益相关者理解。

规则顺序是自由变化的，这意味着规则的顺序不会影响决策的结果。这具有额外的好处，即允许以最大化对决策逻辑的整体理解的方式对规则进行排序，并且还允许将规则作为独立实体进行开发和推理。

任何

A表和Any (A)命中策略定义了一组可能重叠的规则，如果它们确实重叠，则输出值是相同的。因此，它是单次命中单输出策略，这样只要规则匹配所有输入值，就会返回结果，因为任何后续匹配（可能发生）都将产生相同的输出值。Any策略，如Unique，非常常用，它的主要优点 - 以及使用它而不是Unique的原因 - 是它减少了引入条件以排除规则条件的需要，否则会导致相同的结果。因此，总是可以创建一个具有唯一命中策略的等效表，但是具有Any策略的表通常是首选，因为它更容易理解并且创建起来也不那么繁琐，因为不需要确保输入值的每个排列都匹配一个且只有一个规则。

Offer Category					
		Customer Risk Profile	Customer Years	Order Quantity	Offer Type
A		High, Medium, Low			Gold, Silver
1		Low	-	-	Gold
2		-	>5	-	Gold
3		-	-	>=100	Gold
4		Medium, High	-	<100	Silver

在这个例子中，为了方便表格的表，已经定义了规则的顺序，使逻辑更加透明。请注意，前三个规则具有相同的结果，并且已被排序以使表的逻辑更加透明。

当一个表被认为有很多重复的规则和相同的结果时，考虑使用任何命中策略。这种命中策略的好处之一是它使决策表更容易理解，逻辑更透明。

优先

A表级(P)的命中策略定义了一组重叠规则，这意味着规则可能具有不同的输出。因此，它是单次命中/单输出策略，一旦规则匹配所有输入值，就会返回结果，因为任何后续匹配（可能发生）都将产生相同的输出值。与Any一样，Priority策略非常常用，它的主要优点和使用它而不是Any的原因是它支持存在枚举的输出值列表的情况。结果列在输出列中的顺序决定了规则顺序，选择的规则是具有最高优先级的规则。

第一的

具有第一(F)命中策略A表。定义了可能与规定重叠的规则。规则的顺序指定应该命中哪个规则这意味着对于给定的一组输入，表顺序中的第一个规则将触发其输入匹配并产生其输出。因此，它是单次命中单输出策略。它并不常用，许多支持者和经验丰富的决策建模者根本不喜欢使用它，因为它违反了最佳实践规则，即决策表中的规则顺序不应影响结果。然而，对于具有库表规则并且有一些非常重要且可能不太频繁出现的条件在逻辑上会覆盖任何更频繁出现、不太重要的输入条件的情况下，它非常有用。

合并和取消合并单元格

Enterprise Architect有一个有用的特征，允许合并单元。只能合并输入单元而不能合并输出单元，因此必须满足这三个条件才能合并两个或多个单元：

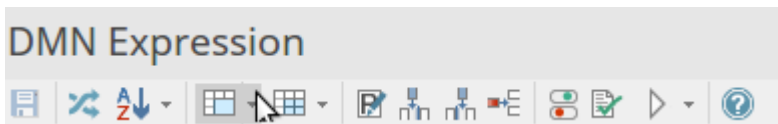
- 必须全部单元单元
- 单元必须是连续的
- 单元必须具有相同的值表达式

在这些条件下，可以合并单元以尝试和简化表，使非技术和技术受众更容易理解。在图中的决策表中我们可以看到有单元满足条件可以合并的单元，分别是：

- 两个风险单元 < 18 和
- 有由区间 [18..55]定义的 Risk Data.Age的三个单元格

Risk Assessment				
		Risk Data . Age	Risk Data . Previous Default Times	Risk Assessment
	U			<i>Low, Moderate, High</i>
▶	1	<18	>3	<i>High</i>
	2	<18	[2..3]	<i>Moderate</i>
	3	<18	<2	<i>Low</i>
	4	[18..55]	>3	<i>High</i>
	5	[18..55]	[2..3]	<i>Low</i>
	6	[18..55]	<3	<i>Low</i>
	7	>55	-	<i>Moderate</i>

这些选项可从DMN 表达式窗口工具栏中获得。在可能合并多个单元的情况下，可以选择仅合并特定单元或合并所有单元。

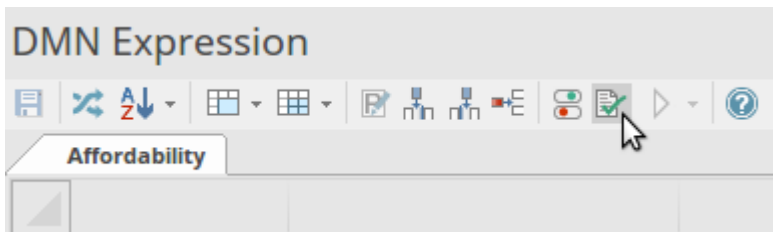


一些决策建模者可能不喜欢这样做，因此问题在于哪种演示最适合观众。Enterprise Architect可以在这里提供帮助，因为它还方便地提供了取消合并选项，因此如果有人之前合并了多个单元，则可以撤消此操作。显然，这可以切换以适应不同的受众。

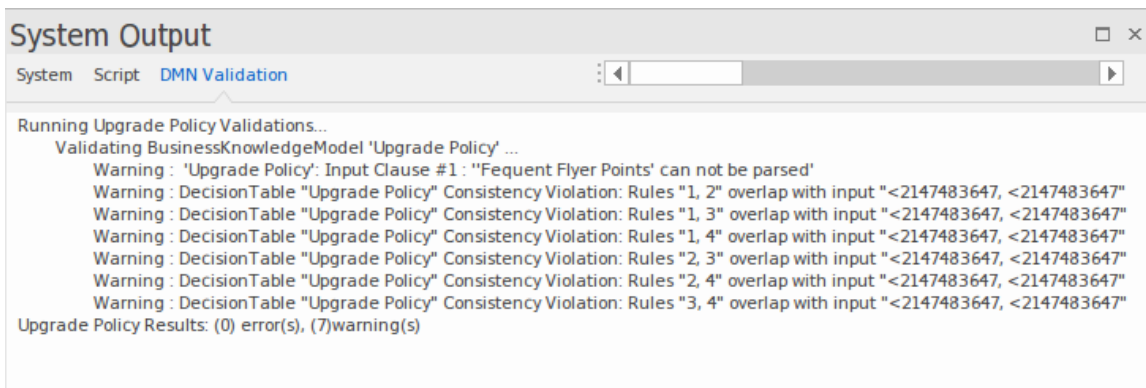
Risk Assessment				
		Risk Data . Age	Risk Data . Previous Default Ti...	Risk Assessment
	U			<i>Low, Moderate, High</i>
	1	<18	>3	<i>High</i>
	2		[2..3]	<i>Moderate</i>
	3		<2	<i>Low</i>
	4	[18..55]	>3	<i>High</i>
	5		[2..3]	<i>Low</i>
	6		<3	<i>Low</i>
▶	7	>55	-	<i>Moderate</i>

验证决策模型

除了作为决策模型的协作开发、管理和模拟平台之外，Enterprise Architect还有一个有用的特征来验证表达式，包括决策表，以确保它们在语法上是正确的，并发现规则之间的差距和重叠。被定义。此特征有助于确保在将决策模型部署到运行时引擎或服务时，规则将正确触发，并将输出形式的适当结果提供给调用应用程序或服务。创建规则的过程通常很复杂，这种验证函数对于业务和技术利益相关者来说都是一个受欢迎的特征。



由于各种原因，为决策模型寻找业务规则和其他输入的过程可能是一个相当大的挑战，当这些规则被收集后，它们将需要转换为决策规则。例如，它们可能以不同的方式定义，并以不同的格式分散在一系列不同的来源中。这可能会给业务带来挑战，因此帮助建模者验证和断言已正确输入规则的特征将有助于确保决策模型形成良好且适合用途。技术人员还可以使用任何托管浏览器的设备（例如智能手机、平板电脑或笔记本电脑）使用讨论、聊天和评论的协作特征来为这些模型和规则定义做出贡献。技术人员还可以使用Enterprise Architect客户端访问模型，并与业务人员一起制定或重组规则以实现最佳理解、消除冗余和/或缺失条件。



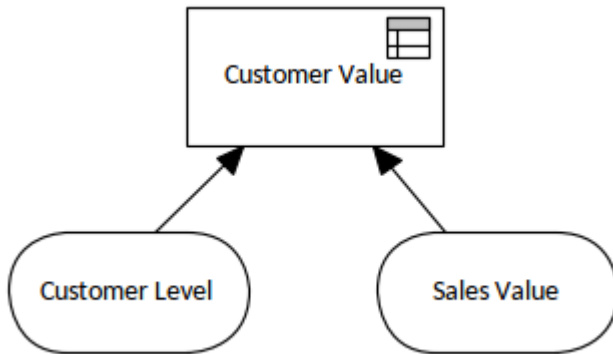
准确定义正在建模的决策的表达式创建就像技术的许多方面一样 - 既是科学也是艺术。科学部分更容易一些，因为它可以在课堂上学习；艺术部分要困难得多，通常是从使用大量决策模型的经验中学习的。事实证明，当遇到相同的问题时，一组决策分析师总是会以不同的方式处理问题，并提出完全不同的决策模型；没有比使用命中策略决策表真实的了，在这种情况下，每个分析师通常都会对特定的政策有预先的行动。这些模型都是正确的，只是表达方式不同。在这些情况下，Enterprise Architect提供了一种高度兼容的标准实现，但在规范没有说明的情况下，或者当分析师更喜欢灵活性时，该工具提供了许多功能，这些功能将受到新手和有经验的建模者的欢迎。

DMN 标准规定了 DMN 表达式语法的许多方面。包括决策表，遵守这些很重要；同样重要的是，规则的其他一些方面也很完善。Enterprise Architect检查中决策表的验证：

- 句法正确性——确保规则符合规范的句法和表达语言
- 完整性——确保规则之间不存在差距
- 重叠 - 确保规则不重叠

规则间隙

决策规则必须在实施到生产系统或服务之前完成。这是必要的，因为它们必须完成覆盖所有可能的输入。有一个可以为表定义的默认输出条目形式的包罗万象的条件，但许多有经验的建模者更喜欢确保所有输入都被逻辑覆盖。对于经验不足的建模者来说，排除在输入域或 Universe 的时间范围内上下文的输入规则也是很常见的。更有经验A建模者会知道，即使当前的经验结果表明并非如此，任何事情都是可能的，因此将确保规则涵盖这些外围输入。可以执行Enterprise Architect的决策表验证来查找规则覆盖范围中的任何差距。这是一个非常有用的功能，因为决策逻辑中的错误可能会导致规则实施中出现不希望的结果，甚至在某些情况下会导致灾难性影响。该图显示了具有两个输入数据的单个决策的简单决策模型元素；它的创建是为了举例说明可以使用验证的方式。



第二个屏幕截图显示了部分完成的决策表，其中仅在行中定义了Gold客户。建模者无意中创建了包含间隙的规则。这是一个简单的演示，读者应该很容易找到差距，但在更复杂的库表中，有更多的规则和输入，定位问题可能非常具有挑战性。

DMN Expression

Customer Value			
U	Customer Level	Sales Value	Discount
	Gold, Silver, Bronze		
1	Gold	> 50000	10
2	Gold	[10000..40000]	15
3	Gold	< 10000	20

我们能够使用Enterprise Architect的内置验证功能来查找规则逻辑中的任何错误。在这种情况下，建模者已经疏忽了，并且没有涵盖每个有效的输入值。没有为销售价值为 (40,000 - 50,000]”的 黄金”级客户定义规则。

System Output

System Script DMN Validation

```

Running Customer Value Validations ...
Validating Decision 'Customer Value' ...
Warning : DecisionTable "Customer Value" Completeness Violation: No rule exists for Customer Level:"Gold", Sales Value:"(40000..50000]"
Customer Value Results: (0) error(s), (1)warning(s)
    
```

Notes System Output DMN Expression

当验证运行时，Enterprise Architect将打开输出窗口并显示验证错误，表明在这种情况下存在“完整性违规”以及表的哪些行受到影响。如前所述，这个例子是刻意去展示特征的，通过扫描表规则很容易识别错误，但是对于更复杂的库表错误可能很难定位。

在此示例中，已为顾客等级输入了允许值，即 金、银、铜”。在定义规则时，决策分析师忘记包含 青铜”输入值的规则。

DMN Expression

Customer Value			
U	Customer Level	Sales Value	Discount
	<i>Gold, Silver, Bronze</i>		
1	<i>Gold</i>	> 50000	10
2	<i>Gold</i>	[10000..50000]	15
3	<i>Gold</i>	< 10000	20
4	<i>Silver</i>	>20000	10
5	<i>Silver</i>	<=20000	5

当验证运行时，错误将报告为间隙，因为为顾客级别定义的枚举值之一根本没有覆盖。

System Output

System Script DMN Validation

```

Running Customer Value Validations ...
Validating Decision 'Customer Value' ...
Warning : DecisionTable "Customer Value" Completeness Violation: No rule exists for Customer Level:"Bronze", *
Customer Value Results: (0) error(s), (1)warning(s)
    
```

可以通过添加另一条规则来纠正该错误，该规则涵盖了顾客级别为*Bronze*的情况。然后，这将确保规则在语法和逻辑上是正确的，并且在验证运行时不会出现错误并且不会生成警告。

DMN Expression

Customer Value			
U	Customer Level	Sales Value	Discount
	<i>Gold, Silver, Bronze</i>		
1	<i>Gold</i>	> 50000	10
2	<i>Gold</i>	[10000..50000]	15
3	<i>Gold</i>	< 10000	20
4	<i>Silver</i>	>20000	10
5	<i>Silver</i>	<=20000	5
6	<i>Bronze</i>	-	2

生成到输出窗口的验证将是：

```

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

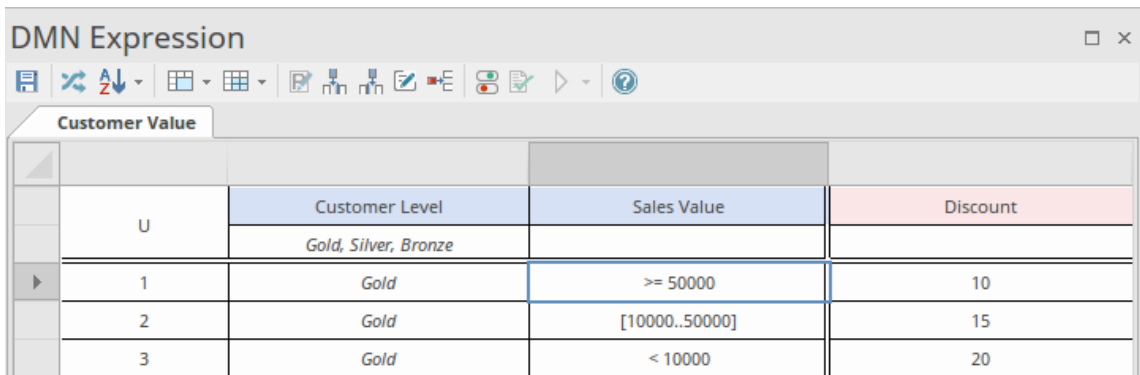
重叠规则

为命中策略为“U”（表示唯一）的决策决策表定义的规则必须是离散的且不重叠。这是一件容易被忽视的事情，即使对于有经验的决策建模者来说，尤其是当一个表变得复杂并且有大量的输入和规则时。使用具有不同含义的括号（round和方形）在 FEEL（足够友好的表达式语言）中使用范围表达式来引入重叠错误是很常见的。

决策表中允许与定义的其他命中策略重叠的规则，例如：

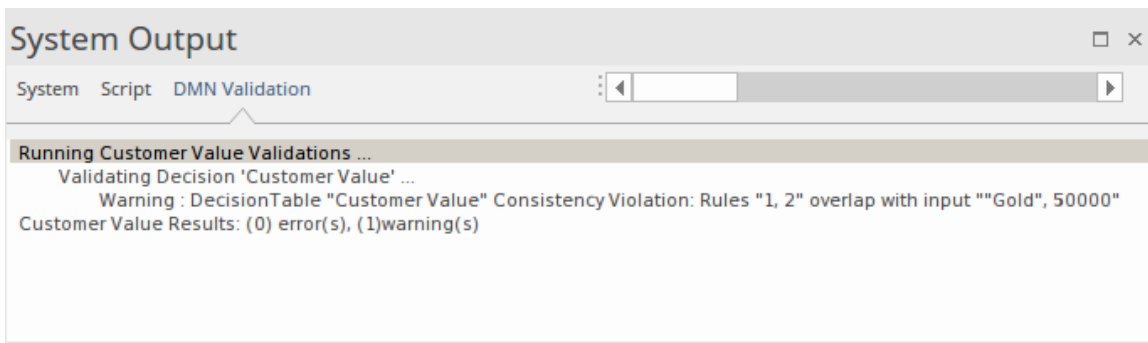
- 具有A（Any）命中策略的A表可以有重叠规则，只要所有重叠规则具有相同的输出值
- A表（优先级）命中策略的表格即使输出值不同也可以有重叠规则
- 即使输出值不同，命中策略为C（Collect）A表也可能有重叠规则

使用上一节中的插图，我们将展示一个简单的重叠示例，该示例由刚刚讨论过的 FEEL 表达式括号的使用错误引起。此屏幕截图显示了两个规则相互重叠的错误，其中问题有点难以识别。引入该错误是因为 Rule-#2 包括使用方括号，它可以有效地重写为 'Sales Value >= 10,000 and Sales Value <= 50,000'。出现问题是因为规则-#1 还涵盖了“销售价值= 50,000”的情况，因此规则-#1 和规则-#2 重叠。



Customer Value			
U	Customer Level	Sales Value	Discount
	Gold, Silver, Bronze		
1	Gold	>= 50000	10
2	Gold	[10000..50000]	15
3	Gold	< 10000	20

我们可以再次使用Enterprise Architect的内置验证功能来帮助我们识别任何违规行为。以与完整性规则类似的方式，验证器可以找到错误，如果表更复杂，则很难找到。该图显示了向系统输出窗口生成的违规，用于识别违规的规则和值。



System Output

System Script DMN Validation

Running Customer Value Validations ...

Validating Decision 'Customer Value' ...

Warning: DecisionTable "Customer Value" Consistency Violation: Rules "1, 2" overlap with input ""Gold", 50000"

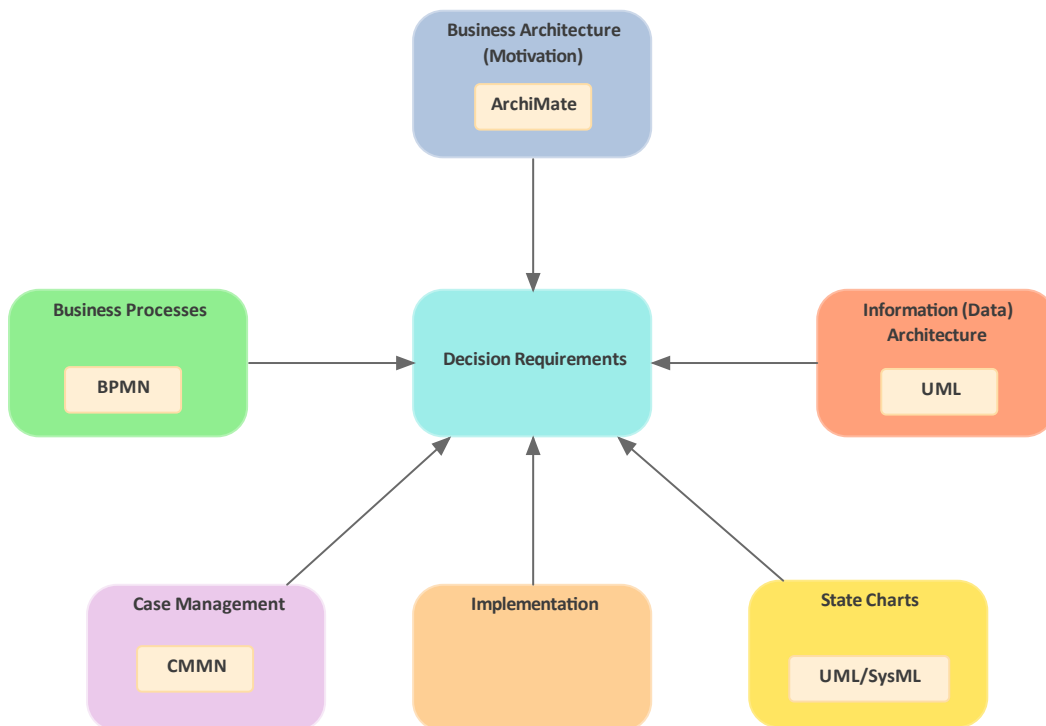
Customer Value Results: (0) error(s), (1)warning(s)

决策模型和符号的上下文

制定决策流程模型和表示法 (决策) 标准是为了补充业务流程模型和表示法 (BPMN)，主要用于业务流程模型图；这两个标准旨在协同工作。BPMN 有一个专门的活动，称为企业规则任务，它充当业务规则计算的占位符，通过提供输入并等待规则引擎提供的输出来执行业务规则计算。该元素充当决策模型的出发点，从而允许与流程模型分开定义和管理复杂且经常不稳定的业务规则。

然而，决策模型和符号规范清楚地表明，DMN 可以独立于 BPMN，并且可以与 DMN 一起使用许多其他标准和语言。该列表确定了可以与决策模型和符号结合使用的语言（通过设计或推理）；未来新的和现有的语言将定义与 DMN 接口的语法。

- 业务流程模型和表示法
- Unified Modeling Language
- 系统建模语言
- 案例管理模型和符号
- ArchiMate



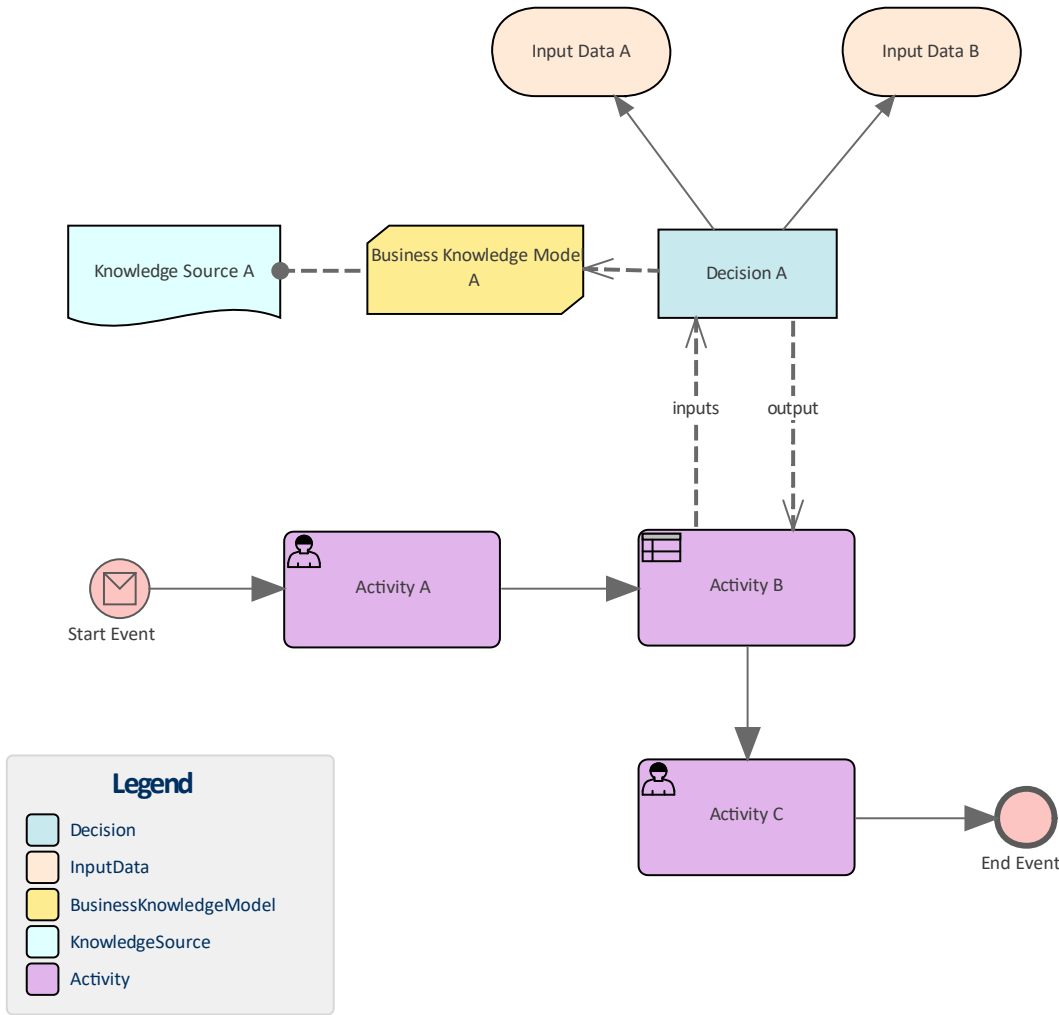
决策不是孤立存在的，它们也不仅仅是流程模型的简化，而是业务意图的表达，通常构成组织与竞争对手的区别。决策需求图允许组织表达决策是如何相关的，而业务流程流程图则阐明了它们在一组流程中的哪些点被调用。Enterprise Architect 的独特定位是作为企业建模平台，以展示决策如何与其他企业建模内容相关，包括本主题其余部分所述。

业务流程模型和表示法 (BPMN)

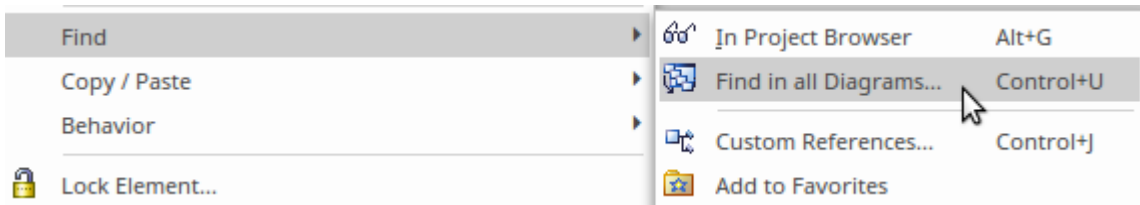
DMN 和 BPMN 语言之间的这种联系将简化业务流程模型，并在组织所做的描述和决策之间明确分离，最终使组织能够快速有效地响应变化。

企业规则任务

企业规则任务被添加到后续版本的 BPMN 规范中，以允许在业务流程中的指定点调用任何业务规则引擎，并允许引擎在处理请求后将结果返回给流程。正是这种机制可用于将决策模型与业务流程集成，从而提供一种句法机制来使这些完全不同的关注点彼此分离。Enterprise Architect 是一个丰富的建模平台，它不仅允许创建两种类型的模型，还允许它们在同一个图表上可视化。



这个通用图表显示了如何在工具中查看这些模型；复合视图只是显示两个模型的一个选项，在这个视图中，建模者只包括最高级别的决策和两个贡献决策。决策使用“模型上下文”菜单并选择“查找|查找所有图表”。

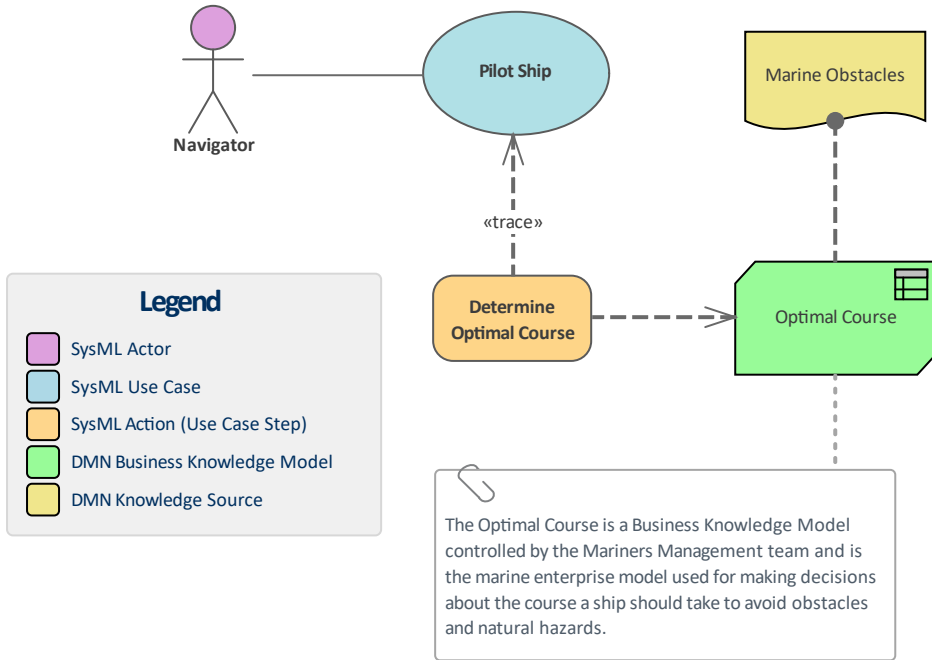


系统建模语言

系统建模语言(SysML) 被系统工程师广泛使用，他们使用一种称为基于模型的系统工程(MBSE) 的方法来描述复杂的现实世界系统。在许多情况下，决策构成这些系统描述的一部分。

使用案例

SysML用例可用于描述用户试图通过使用系统实现的目标。可以使用一系列步骤来描述该用例，这些步骤通常会在用户和系统之间创建向后和向前的交互。系统执行的步骤通常需要做出决策，并且可以使用决策模型对其进行建模。考虑一个描述用例导航系统方面的例子。使用案例中的A步骤可能是“系统决定最佳路线和航路点”。该决策通常会有许多输入，这些输入可以方便地记录在决策模型中。



活动与行动

SysML活动图是 BPMN业务流程流程图的近亲（但更早），并使用类似的符号和语义。传统上，决策逻辑与流程流交织在一起，使用决策、合并、分叉和汇合来描述做出决策的选择和条件。这导致了复杂且通常进程的流程图。使用 DMN，可以将决策（包括它们的逻辑）从图中移除并置于决策模型中。这具有简化图表的效果，从而产生直通的流程和模型，其中可以推理、轻松更改决策并生成实现代码。

Unified Modeling Language

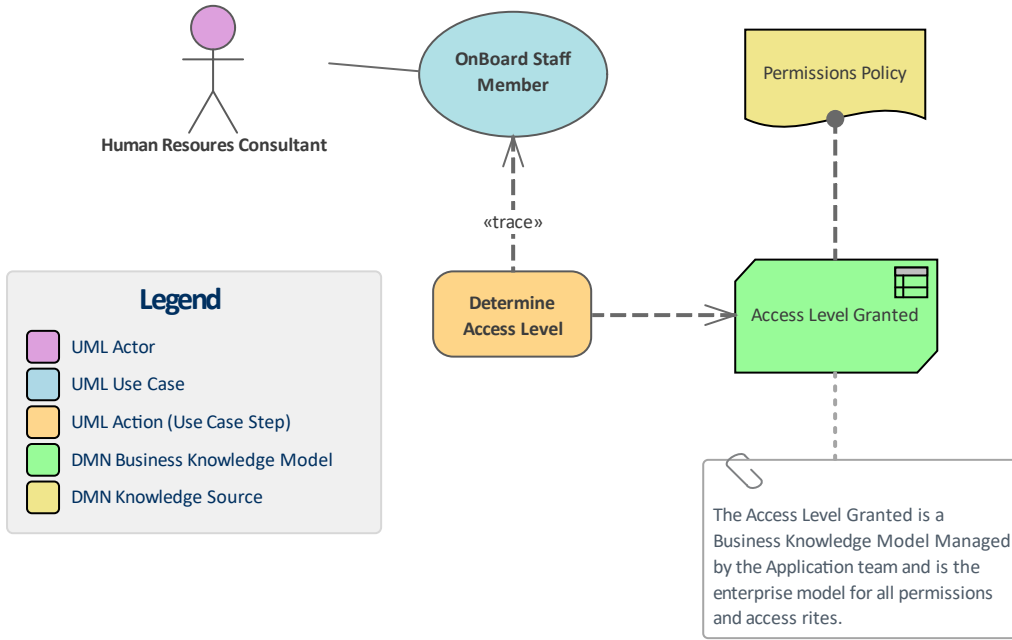
Unified Modeling Language (UML) 已成为对以业务和软件为中心的系统进行建模的事实标准。使用UML建模的系统类型通常具有构成其规范和实现的一部分的重要决策。决策建模在许多地方发挥着重要作用。

活动与行动

UML活动图是 BPMN业务流程流程图的近亲（但更早），并使用类似的符号和语义。传统上，决策逻辑与流程流交织在一起，使用决策、合并、分叉和汇合来描述做出决策的选择和条件。这导致了复杂且通常进程的流程图。使用 DMN，可以将决策（包括它们的逻辑）从图中移除并置于决策模型中。这具有简化图表的效果，从而产生直通的流程和模型，其中可以推理、轻松更改以及生成实现代码的决策。

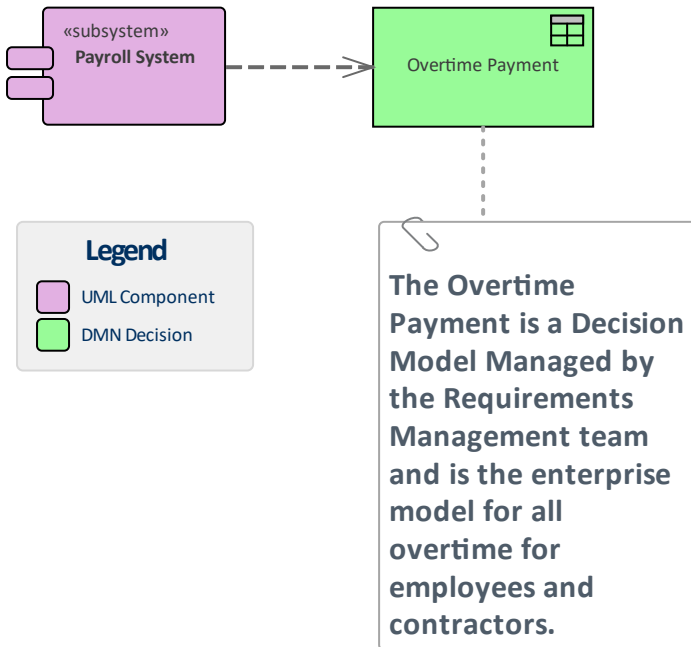
使用案例及其堂兄弟用户案例

尽管关于使用案例和用户案例之间的差异经常存在激烈的争论，但它们都关注用户试图实现的目标。其中许多目标需要在用例或用户故事的不同点做出决定。在使用案例的示例中，可以使用决策模型来描述使用示例中的系统用例，例如“系统确定授予用户的访问级别”。



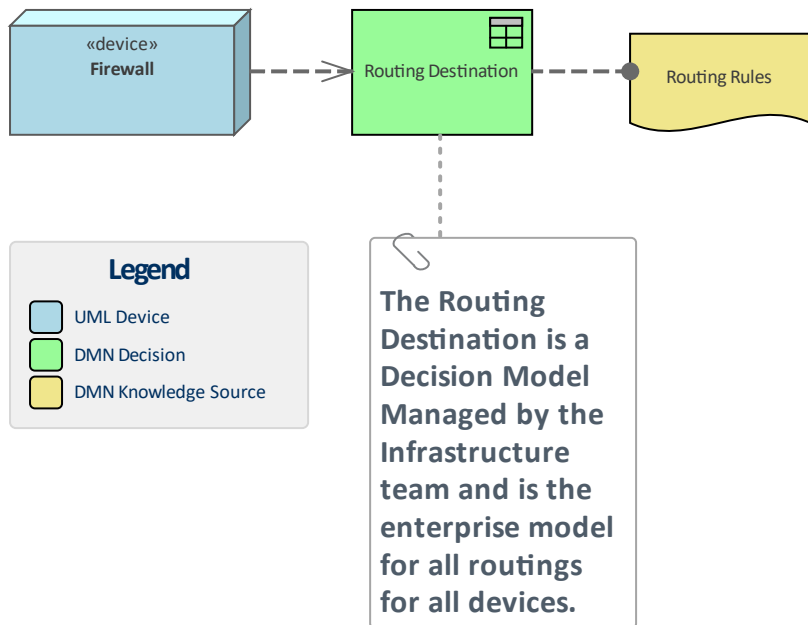
成分

许多系统被划分为一系列组件，这些组件负责系统函数或服务的离散部分。为了让一个部件执行它的工作，它经常需要做出决定。考虑一个工资系统，它必须确定是否在特定情况下适用超时，或者空中交通控件系统，它必须决定是否将进入的飞机置于等待模式，以及等待多长时间。（大多数人曾多次接受这一决定！）



设备

无论是虚拟的还是物理的，许多设备都需要做出复杂的决定。考虑一个必须对网络流量路由到哪里做出复杂决策的路由器，或者一个必须调度各种流量控制机制以优化流量的流量控制器，或者一个保护组织网络的防火墙。

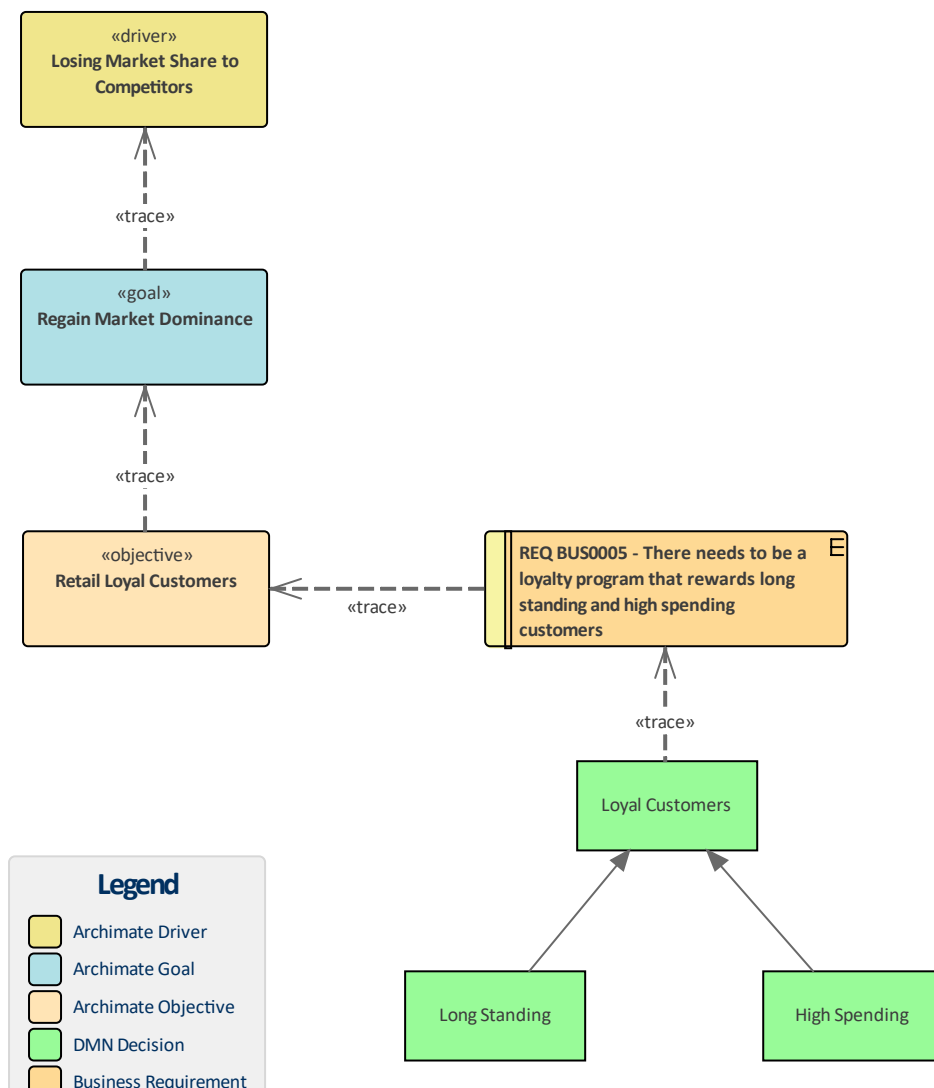


ArchiMate

ArchiMate 是一种企业架构建模语言，用于在多个不同级别创建、管理和可视化架构。有许多重要的地方可以定义和描述决策，包括在战略层面。考虑一个架构，它定义了一个应用程序服务，它选择一组默认的产品来提供给客户。关于捆绑哪些产品的决定可以在决策模型中表达。

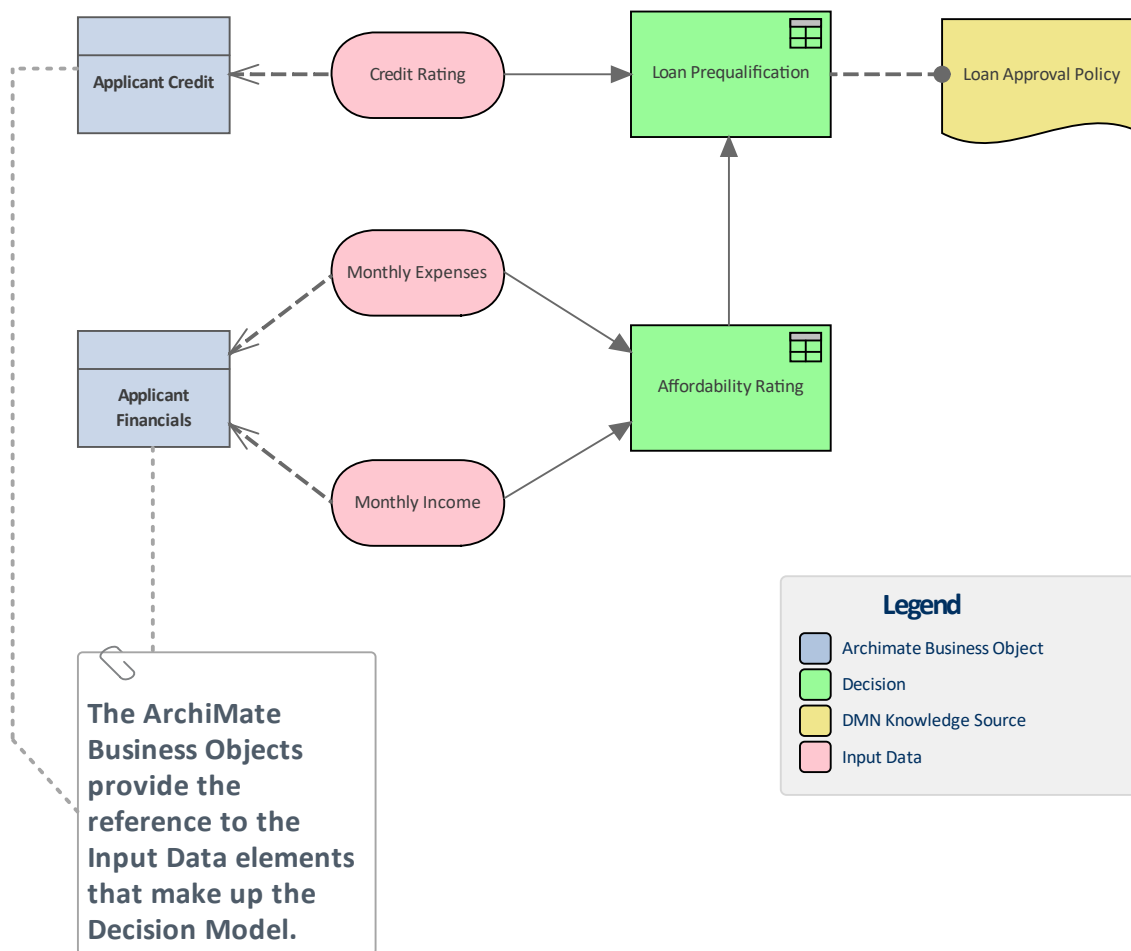
驱动因素与目标

决策可能与创造、激励和推动组织变革的驱动因素有关。为了充分阐明目标，可以使用决策来显示方向设置的潜在差异。通常需要在组织的这个级别做出高层决策。



企业信息模型

决策模型所需的输入数据可以在任何细节级别与信息模型中的实体相关，从高级概念模型到物理数据模型模式。将决策模型与信息模型连接起来可确保决策所需的数据在实施决策时可用。



政策和标准操作过程

决策、业务模型或知识源可以与模型策略、标准操作程序或工作流程的元素相关。这些通常是决定或指导决策的信息源。

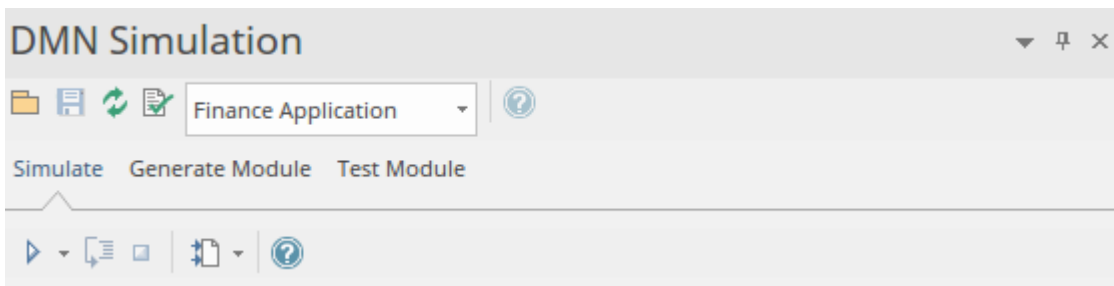
应用服务

与提供服务相关的决策可以与应用服务相关，以展示服务如何做出决策。

仿真决策模型

大多数业务和技术利益相关者都熟悉纠正生产系统中发现的错误的高昂成本。有一些众所周知的例子和指标说明了这样一个事实，即在投入生产之前修复某些东西的成本通常比系统交付时的补救成本低一个数量级（或更多）。发射哈勃太空望远镜的天文学家非常清楚这一点，一家未能识别数百份欺诈性贷款申请的公司也知道这一点。错误决策的成本可能是灾难性的，对企业、技术、工程或科学事业造成严重影响。正是考虑到这一点，Enterprise Architect的设计者在决策模型以及一系列其他技术和语言的建模环境中构建了模拟能力。这与建模、协作特征和代码生成功能一起，使Enterprise Architect成为用于决策建模和管理的卓越平台。

模拟运行在模型开发时、投入生产之前、结构模型更改之后或规则或示例数据集更新之后在模型上运行。仿真本质上是一种在复杂模型中为整个模型或个体决策运行试运行模型的方式，允许团队或个体利益相关者查看输入，可视化执行路径，并分析中间决策和输出对于给定的模型或模型的片段。



模拟无缝运行，无需建模者执行任何配置，并且两级工具栏提供了许多有用的选项。仿真窗口顶部的顶部工具栏提供以下选项：

- 包图标：允许选择一个包
- 刷新图标：允许重新加载模块
- 验证图标：允许执行验证
- 下拉列表：允许选择模型中的任何决策作为模拟的起点

在仿真窗口下部选择“仿真”面板后，工具栏提供：

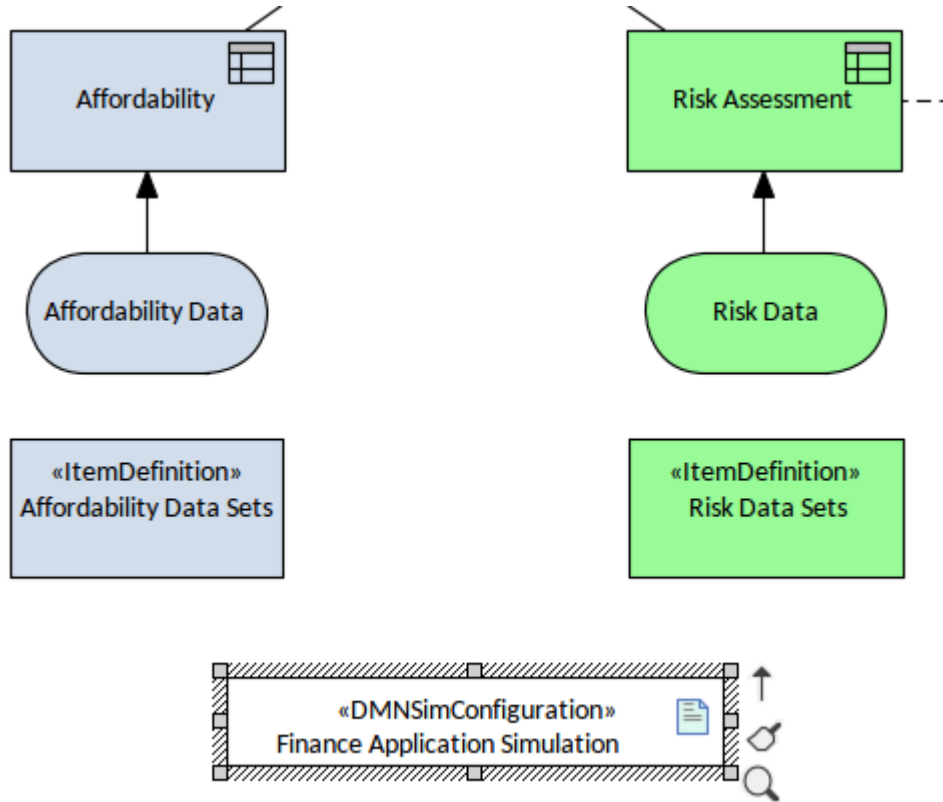
- 播放按钮：运行模拟
- 节通过按钮节通过模型中的每个可执行元素
- 停止按钮：在执行过程中停止模拟
- 导出按钮：导出所有或选择的输入到 BPMN 数据物件

有关更多信息，请参阅[DMN Simulation Toolbar](#) 话题。

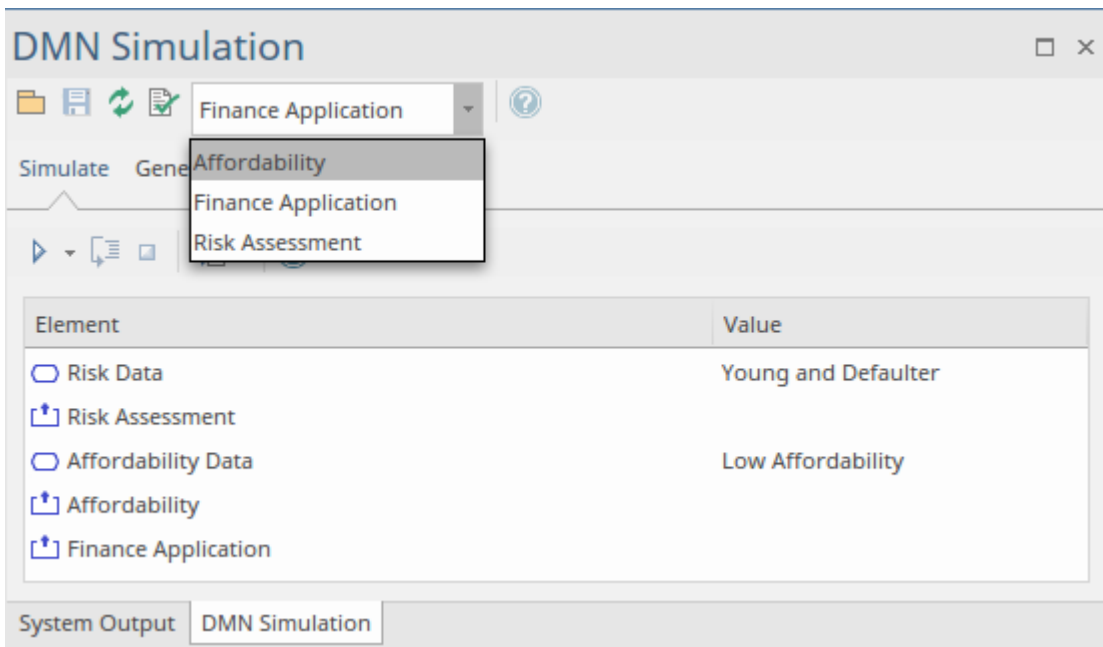
搭建仿真

设置模拟很简单；Enterprise Architect提供了工件可以简单地添加到任何决策需求图中的工具，以开始配置决策模型的过程。DMN仿真配置元素可从DMN工具箱页面获得，其缩写名称为DMNSimConfiguration。

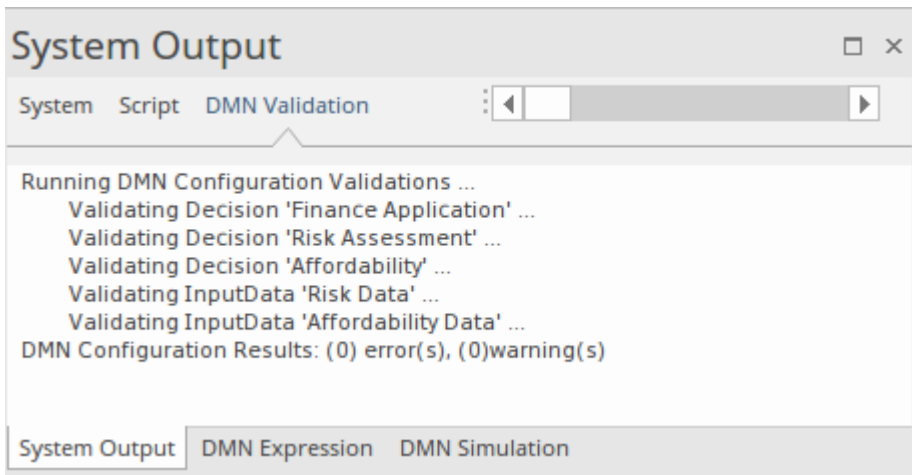
要启动该过程，只需将DMNSimConfiguration元素拖放到决策需求图上，然后双击该元素以启动仿真窗口。



仿真窗口完成了大部分繁重的工作，您只需执行少量操作即可配置仿真。当窗口打开时，它将预加载所有可能的决策或业务模型，这些模型可以作为模拟的起点；这些在下拉列表中提供，您可以从中选择任何一个作为模拟的基础。只有两件事必须配置，当您打开窗口时将它们设置为默认值 - 选定的决策默认为层次结构中的最高层，数据集预设为默认值。如果您决定使用默认值，您可以简单地运行模拟，而无需配置任何东西。

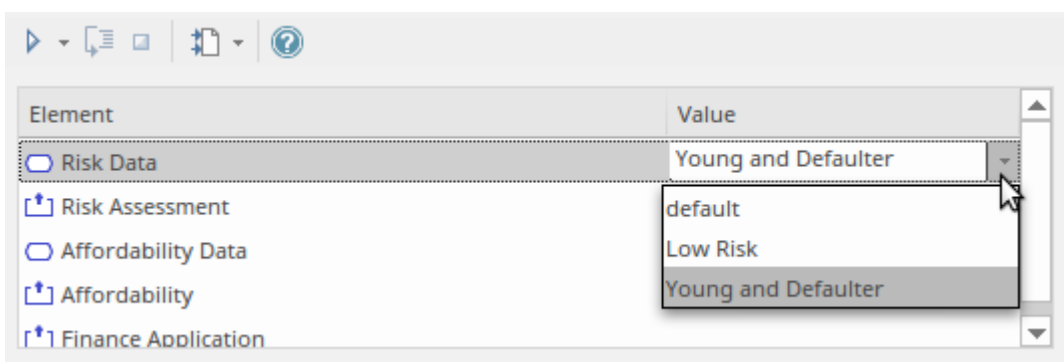


然而，此时运行验证检查将被认为是一种良好的做法，该检查将对图中的所有决策和运行模型（业务）运行。此选项可从仿真窗口工具栏中获得，并允许您在运行仿真之前检查表达式中是否存在句法或语义（重叠和间隙）。如果验证导致错误，那么在运行模拟之前纠正问题是明智的。



下一步是选择参与者决策和 BKM 所需的输入数据。此功能是Enterprise Architect成为该领域市场领导者的原因之一，因为它允许建模者或团队使用不同的输入数据运行模拟，这些输入数据可以保存为一组并重复使用，以可视化决策模型将如何响应在不同的情况下。

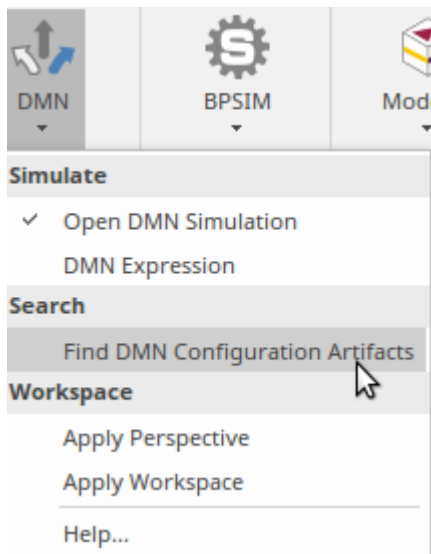
可以为每个输入数据元素选择已预定义并赋予有意义名称A数据集。数据集列表可从列表中输入数据项右侧可见的下拉菜单中获得。从列表中选择个项目会告诉模拟引擎这是您要用于特定输入数据项目的数据，并且这将在您运行模拟时显示。



使用这些设置，您可以运行仿真并查看输出。我们将在下一个主题中探讨可用于模拟的特征。

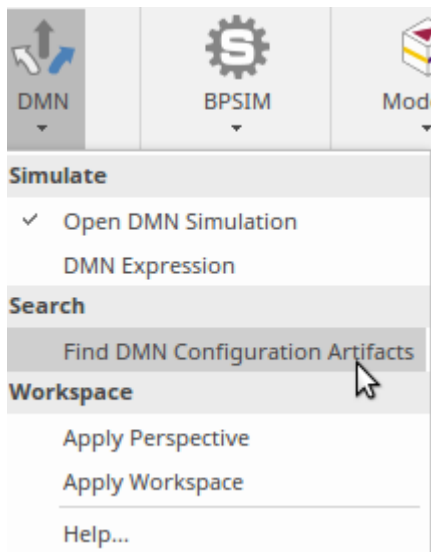
寻找仿真配置

在某些情况下，您可能忘记了之前为模拟设置的决策模型的位置；在这种情况下，Enterprise Architect提供了一个有用的特征，它允许您定位仿真工件，因此可以找到包含它的图表。该功能可从此位置获得：



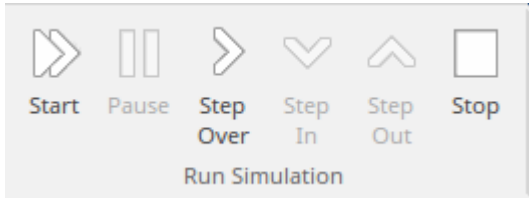
仿真>决策分析> DMN > 查找工件配置。

这将返回发布的仿真工件使用时间（右键）菜单，您可以找到包含该元素的图表，然后双击图表仿真的上下文元素，将启动仿真。

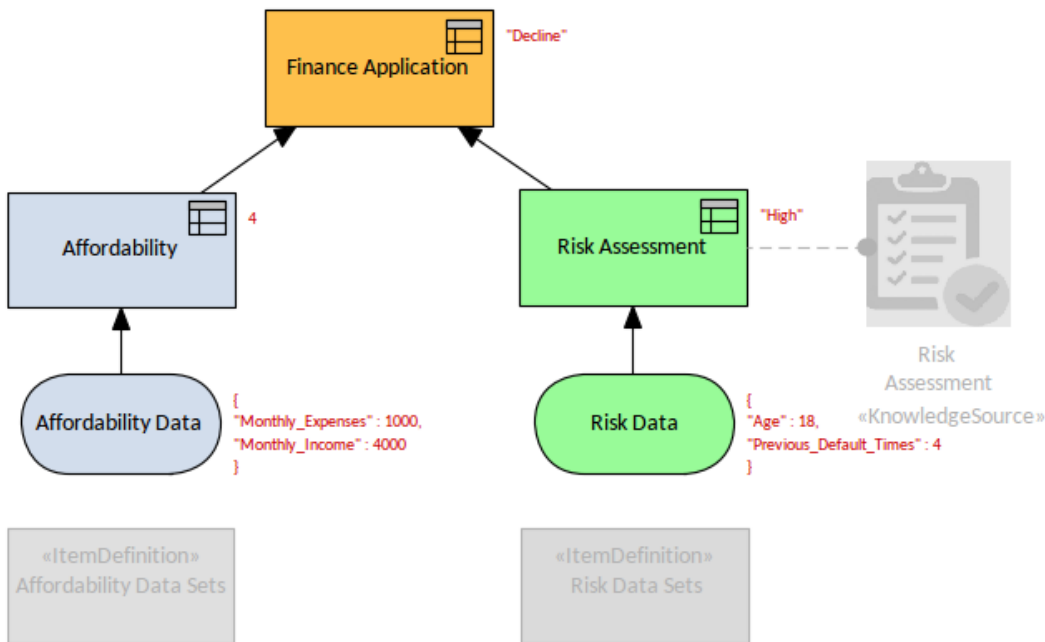


运行仿真

运行模拟的过程再简单不过了——您只需选择运行仿真窗口或“运行仿真”面板上的“播放”按钮，该面板可从仿真功能区获得。打开决策需求图是常见的做法，但即使不是，Enterprise Architect也会启动该图，因为它将用作描述下一节中阐述的决策步骤的画布。



无论您选择哪种方法，模拟都将运行完成，并且结果（包括中间决策步骤和输入数据）将在图表上进行注释。这些注释将受到决策建模者和其他利益相关者的欢迎，因为它们允许这些利益相关者可视化模拟引擎如何到达最终输出的机制。



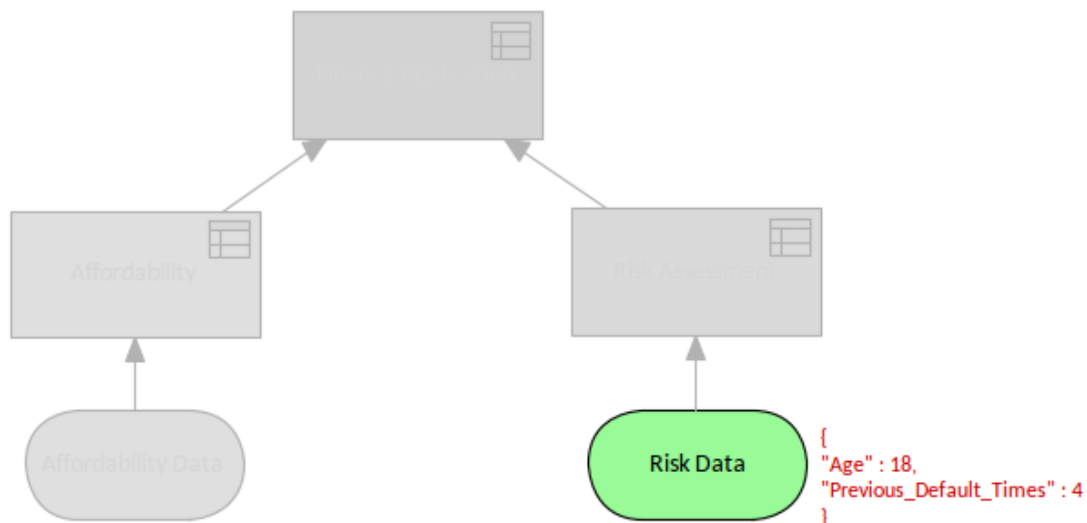
在这张图中，你可以看到从根层到树干的决策已经执行，最高层决策的最终输出是‘Declined’——意味着客户的融资申请没有被接受。即使在这个相对简单的模型中，能够看到中间决策输出值（例如 Affordability=4，风险评估 = 高）也是有用的，但在复杂模型中，能够可视化有助于最终决策输出。在许多情况下，此信息很重要：

- 模型的增量开发：包括在部署后对其进行重构
- 测试模型：确保使用给定的数据集生成正确的结果
- 决策解释：包括向客户等利益相关者解释决策是如何达成的

走过仿真

一个更有效的功能是通过模拟的能力，让您在执行模拟时有效地查看引擎的肩部。同样，这将受到建模者的欢迎，因为它具有在调用决策表时突出显示规则的额外好处，从而允许模拟观众准确地看到在执行过程中的每个步骤中触发了哪些规则以达到输出。

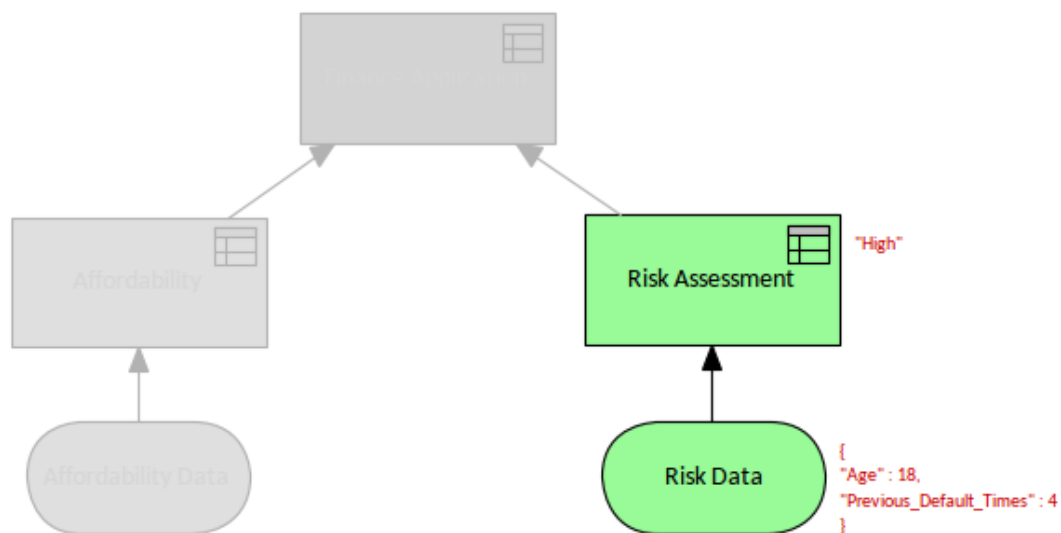
这是一个非常有用的特征，令人惊讶的是此时可以发现多少业务逻辑错误，从而允许对规则进行细化和磨练，以便在将它们放入生产系统之前将它们视为完成并正确。如前所述，要逐步完成模拟，您必须通过单击“播放”按钮来运行模拟直至完成。模拟运行，您可以选择“节”图标，引擎将从第一步开始模拟，并在第二步之前暂停，让您逐步查看结果。此图显示第一步后的图注释：



此图显示了为输入数据配置的数据集，该数据集用于模拟中的此步骤：

Risk Data : Risk Data Sets		
Risk Data Sets	Age : number	18
	Previous Default Times : number	Type in Input Data Values...

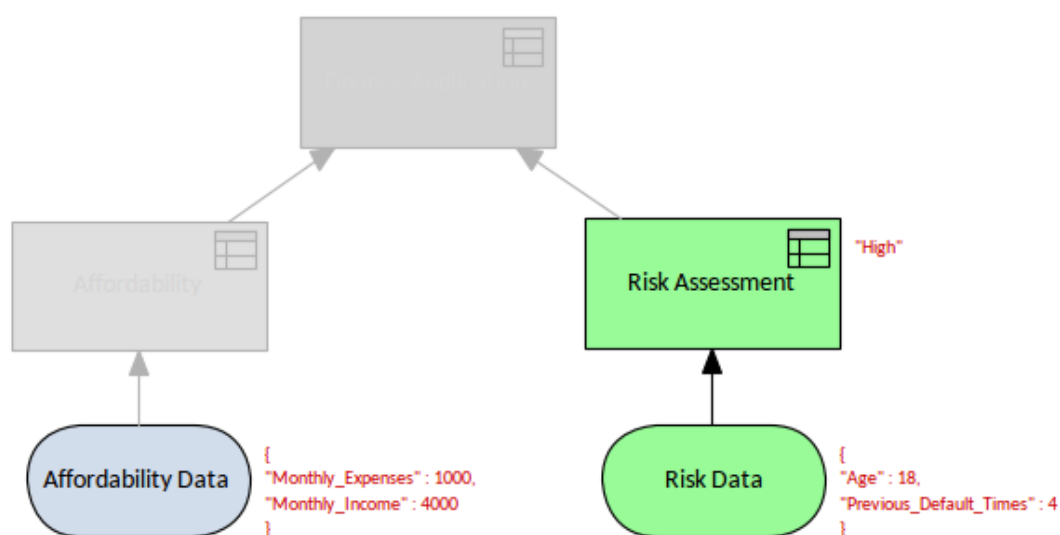
此信息也标注在图表上，随着图表上信息的积累，在模拟的每一步之后都可以看到，因为每一步都在运行。



当您逐步完成模拟时，Enterprise Architect将在表达式编辑器中显示选定的规则，并且 - 在决策表的情况下 - 被触发的规则将在表中突出显示，允许模拟观众清楚地看到规则。此时模拟将暂停，等待您再次点击节按钮恢复模拟。

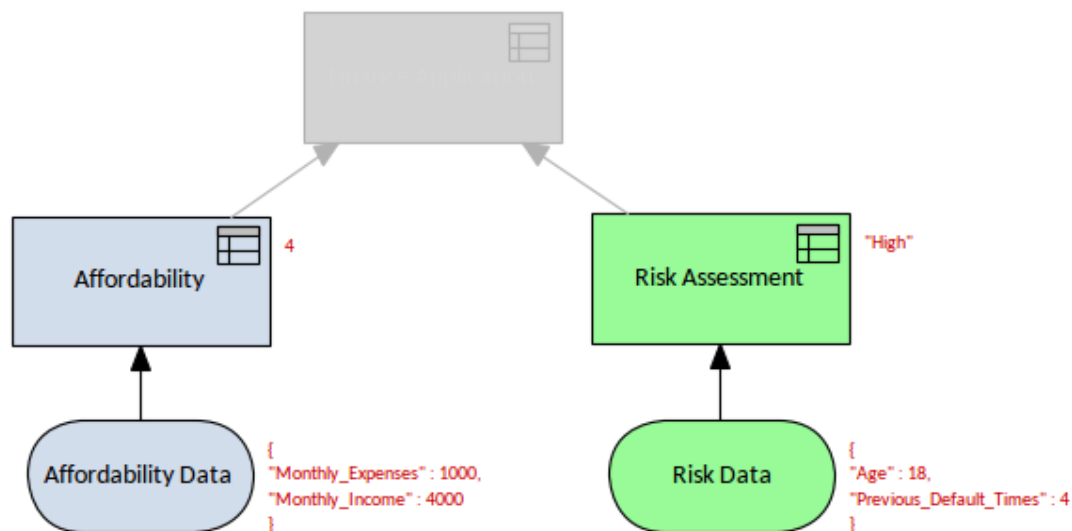
Risk Assessment			
U	Risk Data . Age	Risk Data . Previous Default Times	Risk Assessment
	18	4	High
1	<18	>3	High
2	<18	[2..3]	Moderate
3	<18	<2	Low
4	[18..55]	>3	High
5	[18..55]	[2..3]	Low
6	[18..55]	<2	Moderate
7	>55	-	Moderate

可以一步步继续模拟，在暂停状态中的每一步都可以查看输入数据、选择的规则和图表注释，然后再继续，直到输出最终结果供最高层决策决策层级。



在此图中，模拟引擎输入了负担能力数据，这将用于计算申请人是否有能力偿还贷款。我们将通过下一个插图的结果看到将做出什么决定。

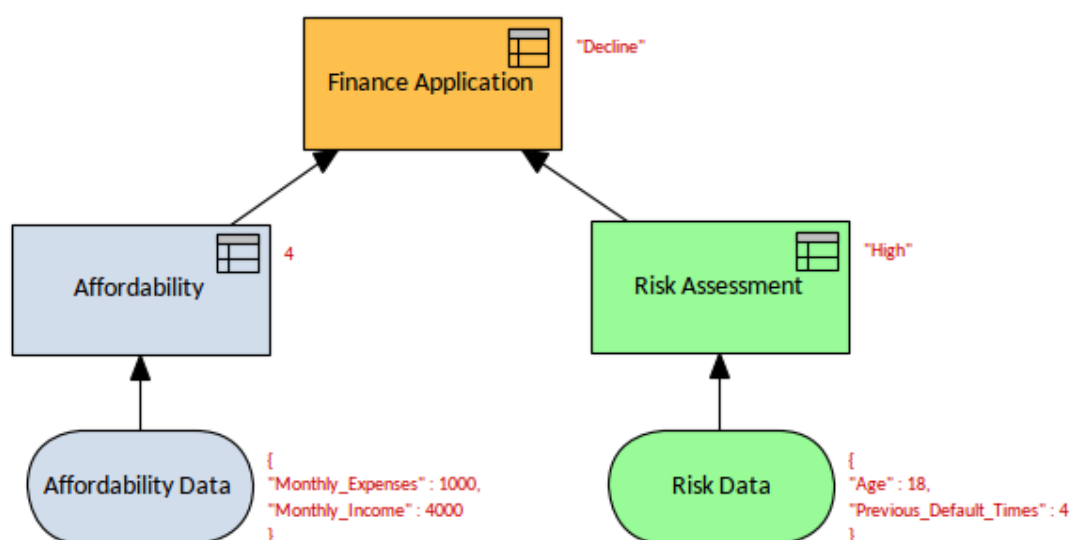
Affordability			
U	Affordability Data . Monthly Income	Affordability Data . Monthly Expen...	Affordability
	4000	1000	4
1	>10000	>3000	10
2	>10000	<=3000	9
3	[5000..10000]	<=2000	8
4	[5000..10000]	>2000	7
5	<5000	<=1000	4
6	<5000	>1000	3



您可以继续逐步进行模拟，现在我们将显示最高级别决策的结果和决策表中选择的规则。

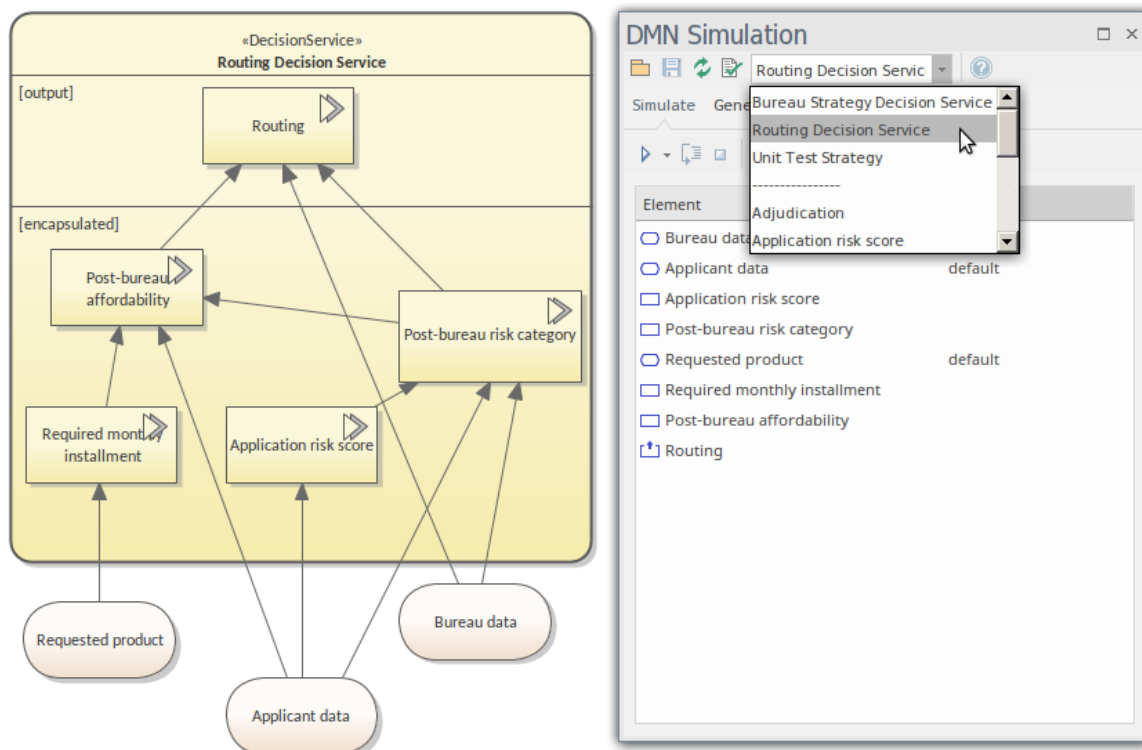
Finance Application			
	Risk Assessment	Affordability	Application Decision
A	High	4	Decline
1	Low	>8	Accepted
2	Moderate	[5..8]	Accepted
3	High	<5	Decline
4	Low	[5..8]	Accepted
5	Moderate	>8	Accepted
6	High	[5..8]	Review

下图显示了模拟的最终结果，不幸的是，使用提供的数据集，申请人的财务申请将被拒绝。这由财务应用程序决策右侧的最终输出注释指示，这是层次结构中的最高级别决策。




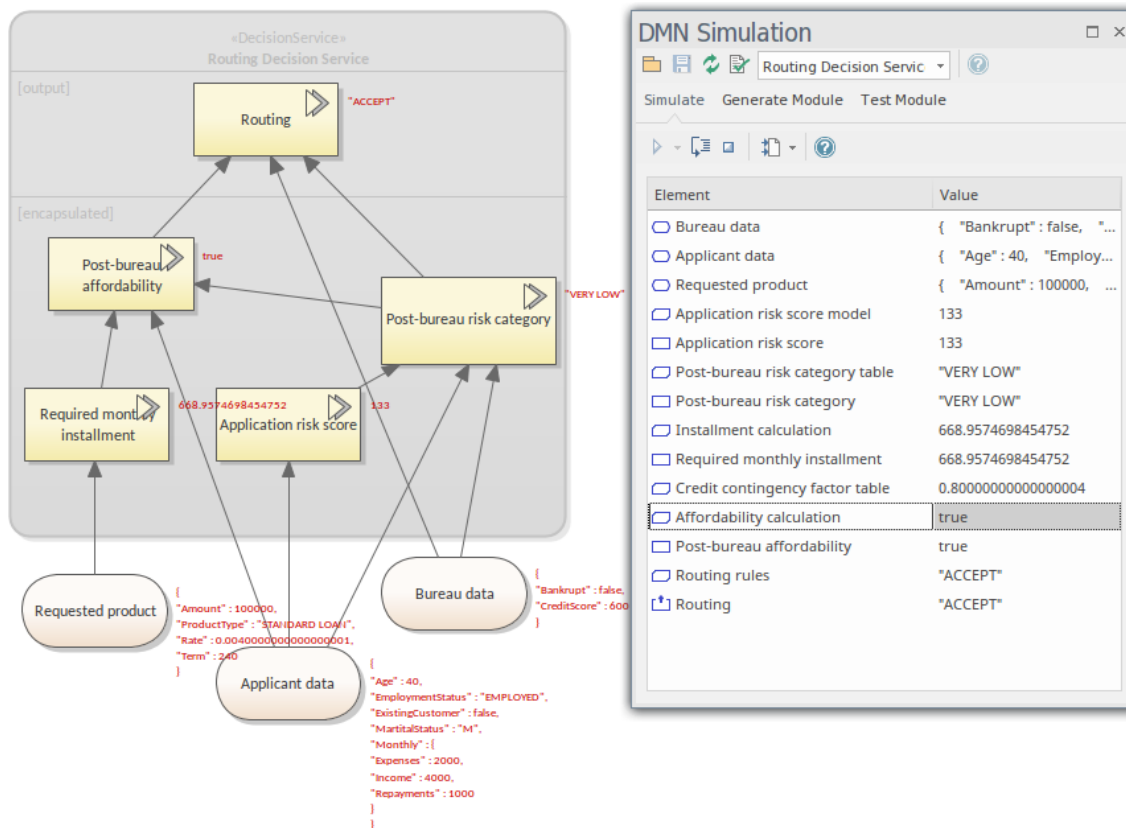
仿真决策服务

决策决策可以像模拟其他决策一样模拟服务。始终将仿真配置作为起点工件到图表上，然后选择决策作为起点和所需的任何输入数据集。



默认情况下，所有决策服务元素和每个离散决策都在对话框工具栏的下拉字段中列出以供选择。

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。这遵循决策层次的原则，从根开始，移动到树干（已选择的最高级别决策）。通过在组合框中选择数据集提供输入数据后，单击保存图标和工具栏上的  按钮。



该图显示了模拟的结果，它允许模拟观众通过使用中间决策和输入数据对图表进行注释来可视化最终决策是如何达成的。也可以在模拟运行后逐步执行模拟并查看每个决策是如何达成的，包括决策表达式编辑器中规则和表达式的决策化。例如，如果已经使用决策表定义了表达式，则将突出显示被击中的规则或规则，从而使观众在应用逻辑时能够理解逻辑 - 并可能纠正错误或疏忽。

从决策模型生成代码

许多潜入生产系统的错误是负责实施系统的技术人员对业务意图的误解的结果。这些错误不是故意的，即使有最好的流程，它们也可能在不经意间找到进入关键系统的途径。Enterprise Architect的代码生成功能将错误的可能性降到最低并几乎完全消除。

此特征允许从已创建、验证和模拟并被断言为正确的模型中直接生成实现代码。代码由Enterprise Architect自动生成，无需人工干预，目标可以是以下任何一种编程语言：

- JavaScript
- Java
- C++
- C#

语言的选择将取决于目标系统。生成的代码可用于许多不同的上下文，包括：

- BPSim 执行引擎 - 一种Enterprise Architect功能，用于模拟业务流程模型
- 可执行状态机 - 一种执行状态机模型的Enterprise Architect功能
- 您组织的专有实施环境部署为平台或服务

