ENTERPRISE ARCHITECT

用户指南系列

使用 SysML 的 MBSE 指南

Author: Sparx Systems

Date: 2022-08-30

Version: 16.0

创建于 ENTERPRISE ARCHITECT

目录

使用 SysML 的 MBSE 指南 7
具有四个变量的方程 9
工程方法或进程 10
建模作为一门学科 13
开始 15
定义模型的目的 17
决定从哪里开始 19
连接部分模型 20
确保模型的质量 21
系统建模语言（SysML） 22
Enterprise Architect建模工具 23
协作平台 26
项目管理工作台 27
模型存储库 28
开始 29
设置模型结构 31
定制应用程序 32
设置蓝图 33
选择视觉风格 35
选择工作空间 36
设置首选项 37
导入现有材料 38
创建图表元素和关系 41
可视化模型 45
与外部数据同步 46
我们前进的方向 47
了解 SysML图表 51
图形的公共图表 58
块定义图表 68
需求图表 71
用例图表 73
包图 75
活动图表 77
内部块图表 79
参数图表 81
序列图表 83
状态机图表 86
系统建模语言概述 88
语言架构 90
关键语法概念 96
模型、图表、元素和视图 97
作为工程团队进行协作 101
中央共享库 102
云计算 103
讨论和聊天 105
看板资源和日历 106
模型评论 107

在团队图书馆中共享资源 109
在移动设备上查看模型 110
建模未来 111
版本控制和基线 114
可重用资产服务器 116
使用包构建存储库 119
函数的包 121
介绍包图表 122
包组织制度 127
浏览器窗口 129
使用模型视图访问存储库 132
需求定义与管理 133
作为第一类公民的需求 138
需求图表介绍 141
开发需求 145
引出 147
文件来源 148
用户观察 149
利益相关者研讨会 150
创造需求 153
外部和内部需求 154
需求分类 155
需求属性 156
规格 158
满足规范管理器 159
分析 162
优先满足需求 163
验证 166
可视化需求 168
需求图表 169
规范管理器 170
浏览器和视图 171
关系矩阵 173
需求表 174
管理需求 175
追踪需求 176
追踪需求 179
管理不断变化需求 181
修改的影响分析 183
需求波动率 185
重用需求 187
需求关系 189
为需求添加细化 192
遏制关系 193
复制现有需求 195
从另一个人那里得到一个需求 196
确保需求得到满足 197
可追溯性到模型元素 199
核实关系 200
可视化需求关系 202
记录需求 207

项目词汇 208
软件需求规范 209
使用案例描述用户目标 210
需求和使用案例 211
介绍用例图表 213
认识场景构建器 217
构建一个用例模型 219
生成行为图表 220
用例报告 221
使用模块构建和约束模型 223
用块开始 228
造型约束作为积木 230
介绍块定义图表 231
基本结构建造块 236
建模结构特征 237
建模行为特征 243
其它块关系 249
建模交互 254
使用值类型建模数量 257
使用属性和部件来模型块用途 259
内部块图表 260
建模和连接部件 261
建模参数方程 264
介绍参数图表 265
使用元件连接的部件关联 268
使用参数的有效性度量 270
协调行为与活动 272
行动基本行为建造积木 274
介绍活动图表 277
创建活动层次结构 280
使用控件流指定行动序列 281
使用物件流指定项目流 283
使用参数和销对输入和输出建模 285
使用模拟可视化活动 287
分配和其他关系 289
用状态机建模更改 291
状态和行为 293
状态机图表 294
触发器和过渡 297
复合州和区域 300
伪国家 - 交警 302
状态表-另一种视图 306
使用模拟进行可视化和实施 309
作为消息序列的交互 311
生命线、消息和激活 312
介绍序列图表 316
信息配片段 320
模拟可视化 322
Modelica 和 Simulink 中的 SysML仿真 323
SysML仿真如何工作 325
使用开始 327

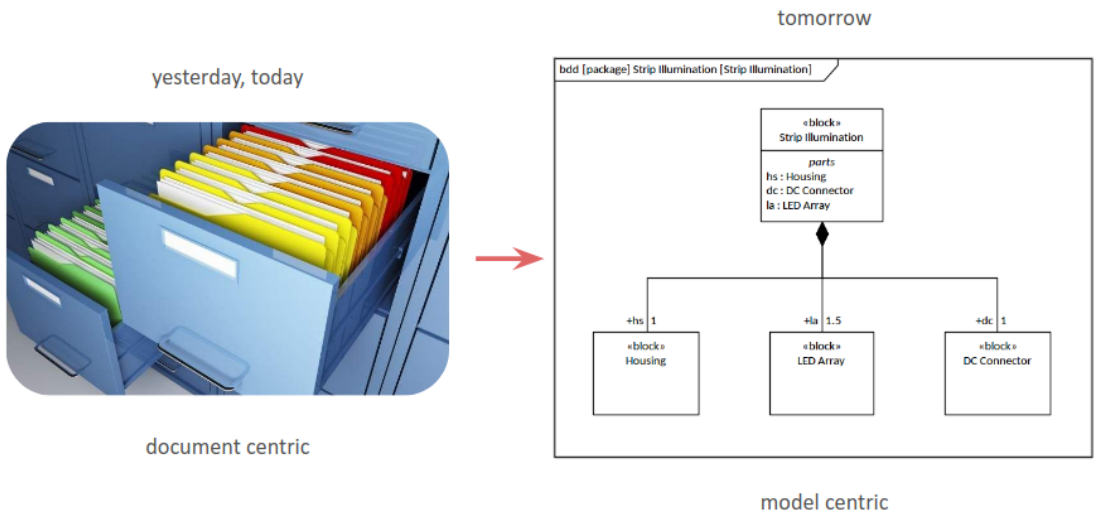
创建仿真模型 332
示例SysML模型 337
包Overview (Structure of the Sample模型) 338
包图-应用SysML配置文件 339
包图-展示包模型结构 340
设置上下文（边界和使用案例） 343
运营域模型- 设置上下文 344
用例图表- 顶级使用案例 346
用例图表- 操作性使用案例 348
细化行为（序列和状态机图表） 350
序列-图表Black Box 351
状态机图表-图表运行状态 355
序列图表开始车辆黑盒和白盒 357
建立需求（需求图表和表） 359
需求图表-图表需求层次结构 360
需求图表- 衍生需求 361
需求图表- Acceleration需求关系 362
库表-需求库表 363
图表Down the图表（块定义图，内部块图） 364
块定义图表-汽车领域 365
块定义图表-混合动力SUV 366
内部块图表- 混合动力SUV 367
块定义图表-电源子系统 368
电源图表内部块模块图 369
定义端口和流 371
块定义图表- ICE Flow属性 372
内部块模块- CAN图表 373
块定义图表- 燃料流量属性 374
参数图表- 燃料流量 375
分析性能（约束图表、时序图表、视图） 376
块定义图表-分析上下文 377
包图-性能视图定义 378
包图-观点定义 379
包图-视图定义 380
包图-视图层次结构 381
参数图表- 有效性测量 382
参数图表- 经济 383
参数图表- 动力学 384
（非规范）正时图 -图表加速度 385
定义、分解和分配活动 386
活动图表- 加速（顶层） 387
块定义图表-加速度 388
活动图(图表) - 加速（细节） 389
内部块图表- 电源子系统行为和流量分配 390
内部块图：属性Specific图表- EPA Fuel Economy测试 391
认识系统工程工具 392
需求图表 393
活动图表 396
用例图表 398
场景生成器 400
审计 402

日历 404
块定义图表 406
内部块图表 407
仪表板图表 409
决策图表 411
状态机图表 413
文档 415
差距分析矩阵 417
热图 419
导入和导出电子表格 420
参数图表 421
模式 422
关系矩阵 424
路线图图表 426
规范管理器 428
策略地图 430
图书馆 432
时间感知建模 434
可追溯性窗口 436
价值链 438

使用 SysML 的 MBSE 指南

基于模型的系统工程（MBSE）已成为复杂系统管理和工程的一种有价值的方法。它是从以文档为中心的方法的转变，允许开发模型并将其用于广泛的目的，包括需求规范、设计、权衡、架构、验证、确认、模拟和支持等。

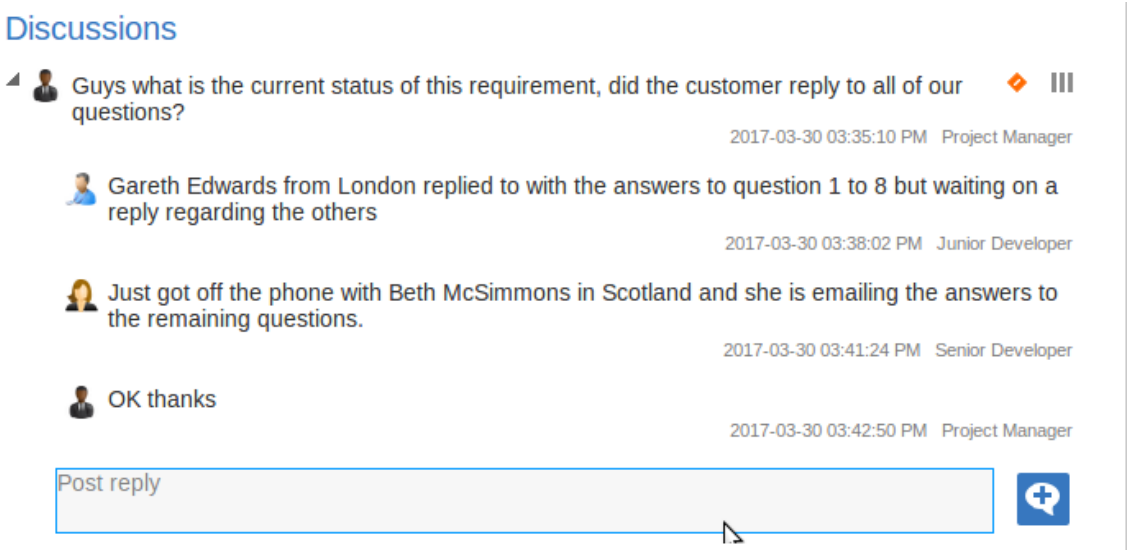
这些模型充当了针对灾难性工程错误的 保险政策”，有助于降低规范、设计、测试、实施和支持阶段的高昂故障成本。



当您使用Enterprise Architect for MBSE 时，您打开了通往全新思维和工作方式的大门。 Enterprise Architect是一个严格的协作平台，允许广泛的利益相关者共享想法、问题、解决方案和实施，包括：

- 顾客
- 高管
- 工程经理
- 工程团队负责人
- 架构师和设计师
- 系统工程师
- 软件工程师
- 测试人员
- 供应商
- 系统集成商
- 技术支持人员
- 用户或其代理人

这些模型还可以通过安全的浏览器界面实时查看、贡献、评论和讨论，利用协作的力量来创建健壮且结构良好的架构和设计。



在本指南中，我们将探讨Enterprise Architect的许多特征，这些特征可用于将单个工程师、团队、组织或整个行业细分到以创新和创新为主的时代所需的实践和绩效水平。前所未有的技术变革。该指南将为系统建模语言提供背景知识，并展示如何在Enterprise Architect中创建语言结构。它的目的是让新人接触到语言和工具，让他们接触到使用这个工程和协作平台在系统工程领域可能发生的一切。用 17 世纪著名物理学家艾萨克·牛顿爵士的话说：

" 如果*I*比别人看得更远，那是因为*I*站在了巨人的肩膀上。"

Enterprise Architect提供了一个可以实现这一愿景的平台，通过协作特征和功能（例如模拟和自动化），工程师可以看到机会、设计解决方案并构建未来。

Enterprise Architect包括系统工程师、技术架构师和其他希望将Enterprise Architect中的建模和仿真工作与MATLAB、Octave、OpenModelica 等结合起来的主要特征。 JavaScript引擎中的 求解器"类和广泛的数学库提供了显着扩展的仿真功能。有关详细信息，请参阅Mathematical Simulations帮助主题。

具有四个变量的方程

本指南的标题暗示只有两个学科需要学习：

- 基于模型的系统工程（MBSE）和
- 系统建模语言（SysML）

但是，在采用这种方法时需要考虑四个方面。这就像一个包含四个未知数的方程式，必须先解决每个未知数，然后团队才能通过 MBSE 项目或计划取得成功。方程中的变量是：

- 工程方法或过程
- 建模学科
- 系统建模语言（SysML）
- Enterprise Architect ，首选的建模工具

并非必须立即掌握这四个方面，但重要的是要了解它们并努力研究它们，并且团队发展技能以单独理解它们以及它们之间的关系。例如，如何使用Enterprise Architect创建 SysML需求图，应该包含哪些属性，应该什么时候完成，应该与模型的其他部分相关联。

*MBSE*方法=进程+语言+建模+工具

该指南将解决所有这些问题，到最后，读者将不会遇到未知变量，但方程将具有基于学习的值，从而用我们开始的四个变量求解方程。在这一点上，读者将朝着良好的工程建模实践迈进。

工程方法或进程

系统建模语言与过程无关，可以与任何方法或过程一起使用。语言的新手有时不理解这一点，他们希望它应该是规范的，并就应该创建什么元素、图表和模型以及何时创建明确的指导方针。这种不可知论的立场提供了极大的灵活性，并允许以适用于流程和潜在问题或解决方案域的方式使用语言。

定义为系统建模语言一部分的元素、连接器、图表和语言定义都是为了允许工程师创建以下模型的明确目的而创建的：

- 使命
- 利益相关者
- 要求
- 有效性措施
- 系统的结构和行为方面，例如最终实现需求的组件

团队用来管理和传播工件的过程是在组织或团队级别上完全任意定义的，并且必须是组织或团队。

系统通常需要一种协作或多学科的方法，团队一起工作以产生满足利益相关者需求的结果。任何过程都有两个重要方面：

- A管理过程——管理利益相关者、风险、进度、预算和质量
- A技术流程——管理架构、分析、设计、集成和测试

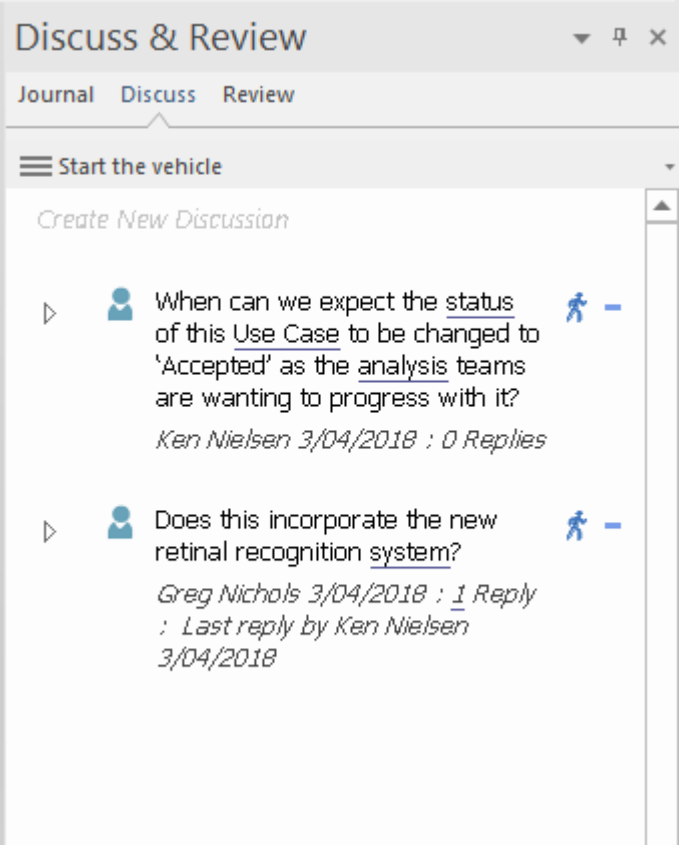
然而，这两个过程显然需要接触点，以确保实现项目的总体使命和目标。

Enterprise Architect允许您使用任何类型的流程，无论它是正式定义的、标准的一部分还是内部制作的。Enterprise Architect中还有一些功能允许您定义、发布和共享定制流程。

支持良好A团队

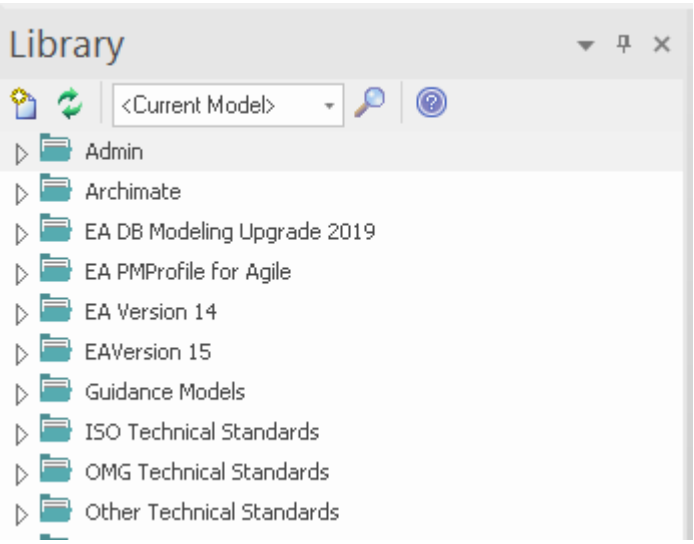
Enterprise Architect提供了大量的工具，可以帮助团队进行协作，无论他们位于何处，也无论他们是如何被时间和距离分开的。该产品从头开始构建为一个协作平台，允许工程和非工程、技术和非技术利益相关者在一个协作和集成的结构中一起工作。

该存储库可以是基于云的，用户可以从全球任何地方安全地连接，有效地创建一个虚拟团队。这对于一些在当地没有专业知识或项目本身是全球性的项目很重要。用户和团队可以使用讨论、聊天、评论和模型邮件等协作特征进行协作。其结果将是协同的架构和设计，这不是一个工程师的工作成果，而是许多头脑的输出，并且工作将超过其部分的总和。

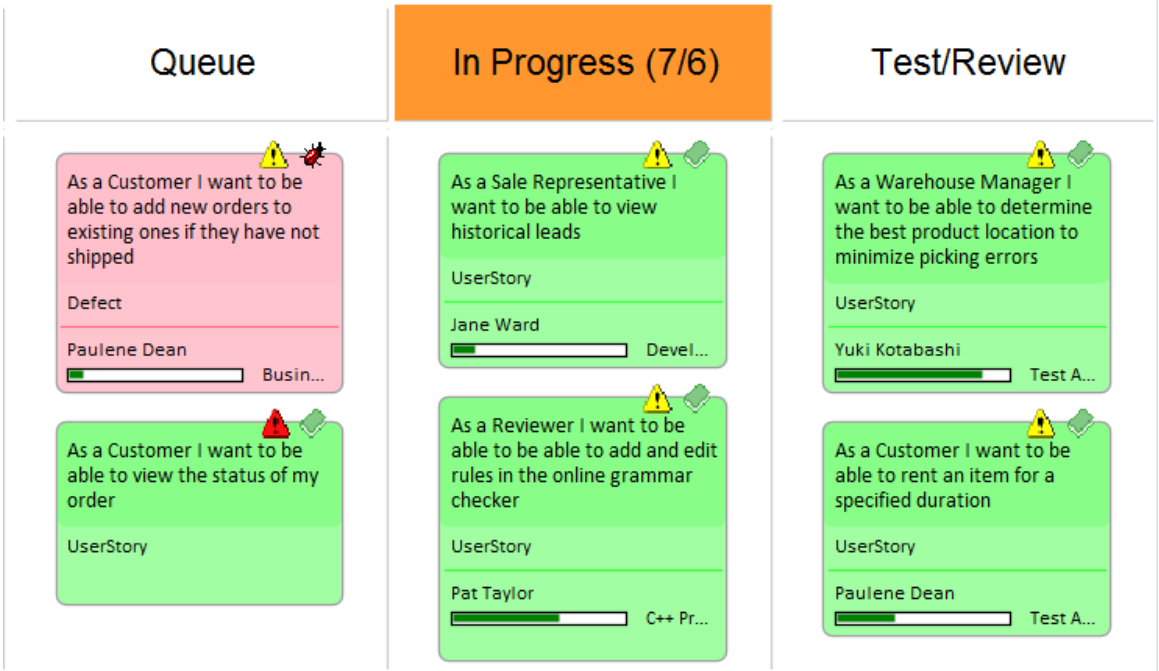


这些工具之所以有效，是因为它们可用于注释模型、元素和图表，使用户可以像在同一个房间的白板上协作一样一起工作。

团队图书馆是另一个方便的协作工具，它允许将任何类型的文件包含在存储库中，或者通过超链接和/或 URL 引用列出其在网站上的外部位置。标准、规范、指南、指南、示例、导师和其他材料等文档都可以在团队图书馆中进行分类。



还有许多其他工具可用于促进团队工作，包括图像管理器、日历、发布、看板、项目管理特征等等。此示例显示了一个看板图，可用于可视化开发系统的物理或软件组件的敏捷团队正在做什么。有关更多信息，请参阅The Modeling Team帮助主题。

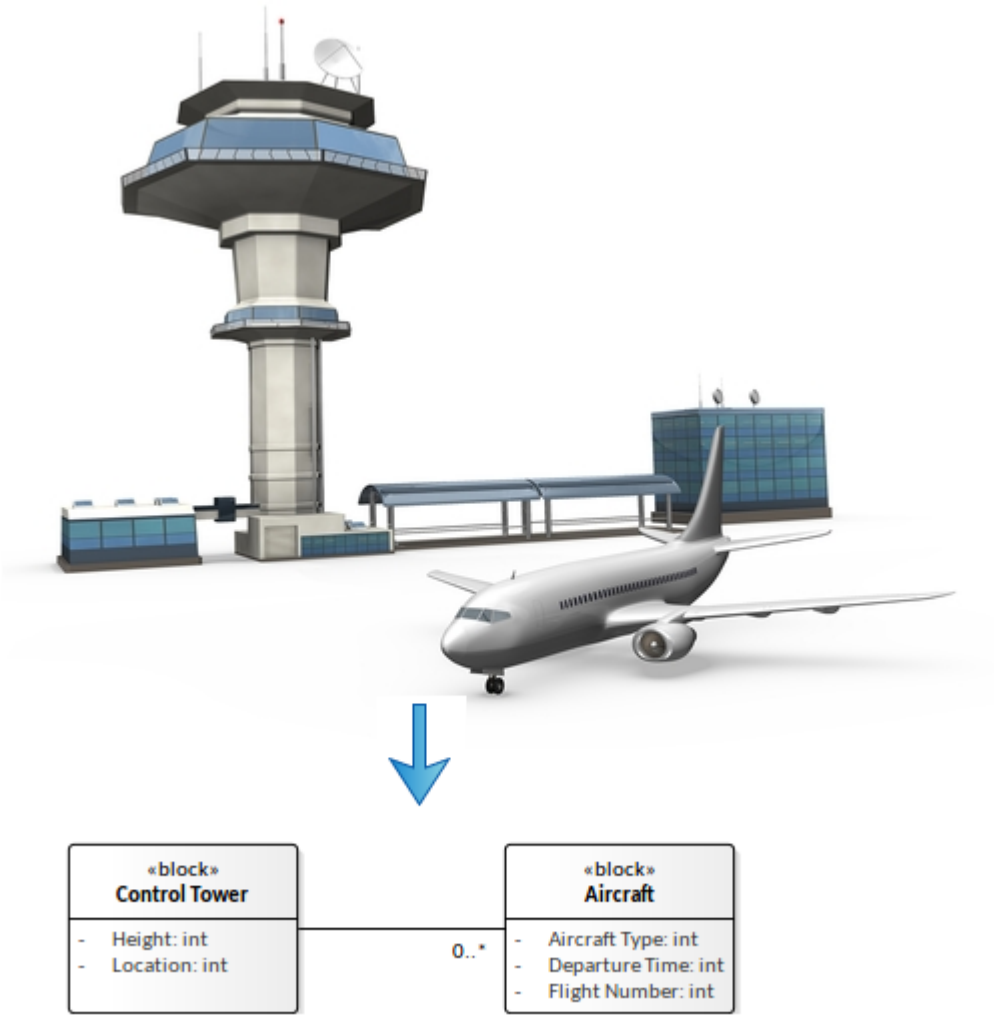

Queue
In Progress (7/6)
Test/Review
As a Customer I want to be able to add new orders to existing ones if they have not shipped
Defect
Paulene Dean
Busin...
As a Customer I want to be able to view the status of my order
UserStory
As a Sale Representative I want to be able to view historical leads
UserStory
Jane Ward
Devel...
As a Reviewer I want to be able to be able to add and edit rules in the online grammar checker
UserStory
Pat Taylor
C++ Pr...
As a Warehouse Manager I want to be able to determine the best product location to minimize picking errors
UserStory
Yuki Kotabashi
Test A...
As a Customer I want to be able to rent an item for a specified duration
UserStory
Paulene Dean
Test A...

建模作为一门学科

大多数人，包括系统工程师，通常会发现写一个主题的长描述比写一个简洁明了的总结更容易——这类似于建模的挑战。

问题不是要包括什么，而是要省略什么。

基于模型的系统工程的优势之一正是这一点——它鼓励工程师创建描述性、清晰和简洁的模型。基于文档的流程的长句（有时是杂乱无章的）被清晰而简洁的图表所取代，这些图表明确地描述了系统的需求、结构和行为。



有些人将建模描述为一门封闭的学科，并将其称为身穿紫色长袍的炼金工程师所练习的 黑魔法”之一。这导致了我们的大学很少将建模作为一门学科教授的问题，也没有大量关于该主题的文献，使其看起来像是一门神秘的艺术，而不是它的本质——一门可以学习的科学。

有许多不同类型的模型，**包括**：

- **比例模型**
- **物理模型**
- **抽象模型**

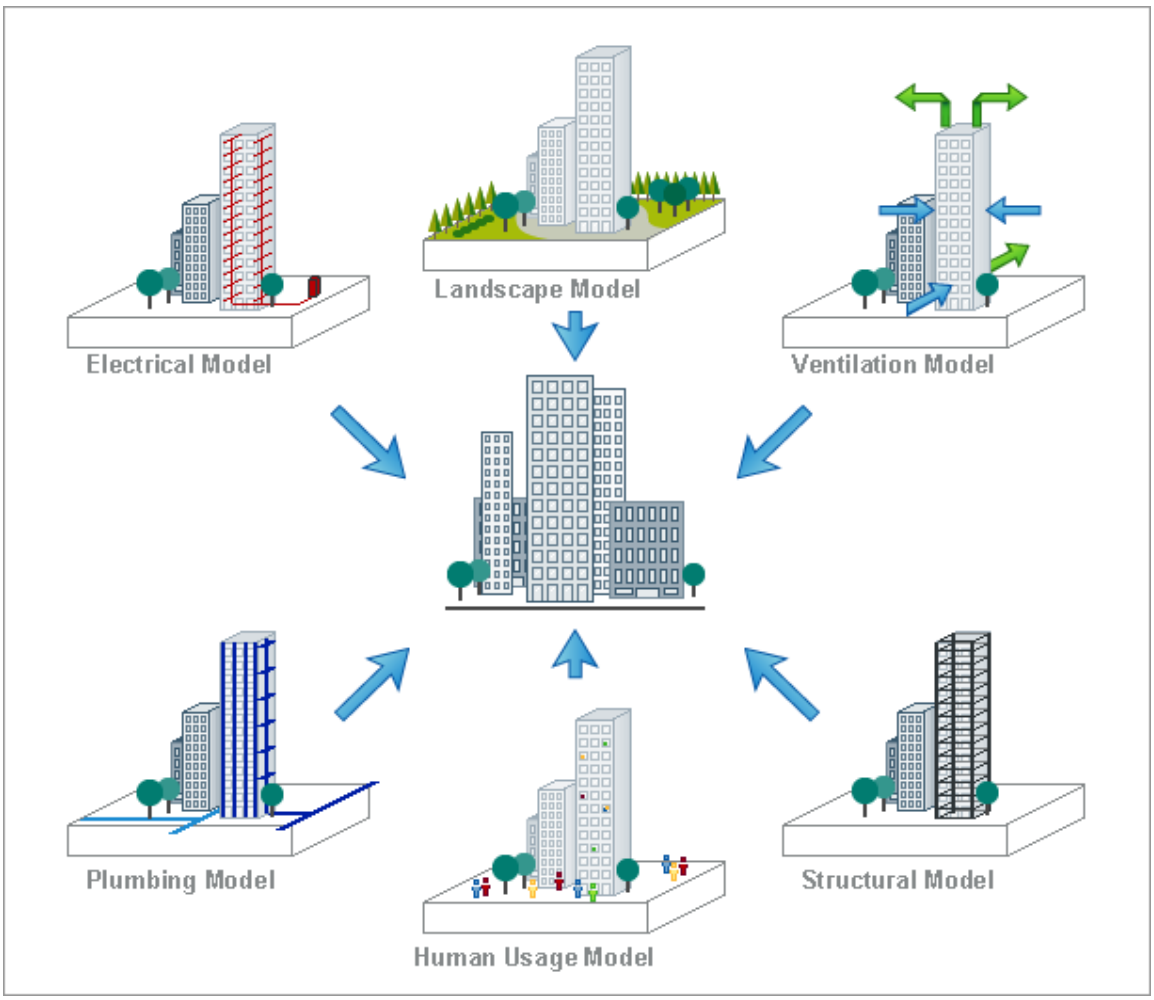
在本指南中，我们最感兴趣的是抽象模型，因为它们是我们通常使用Enterprise Architect和系统建模语言创建的**模型**。

顾名思义，这些模型是对现实的抽象，旨在突出实体、子系统或系统的最重要方面，同时忽略不重要或与该观点无关的事物。有关更多信息，请参阅 Models帮助主题。

开始

抽象模型是对现实世界事物的一种表示，它可以帮助我们在无需查看真实事物的情况下对其进行推理。通常模型要小得多，是系统或其部分之一的简化视图。也可以创建只关注系统的一个方面或方面A模型；例如，飞机的通信或导航系统。

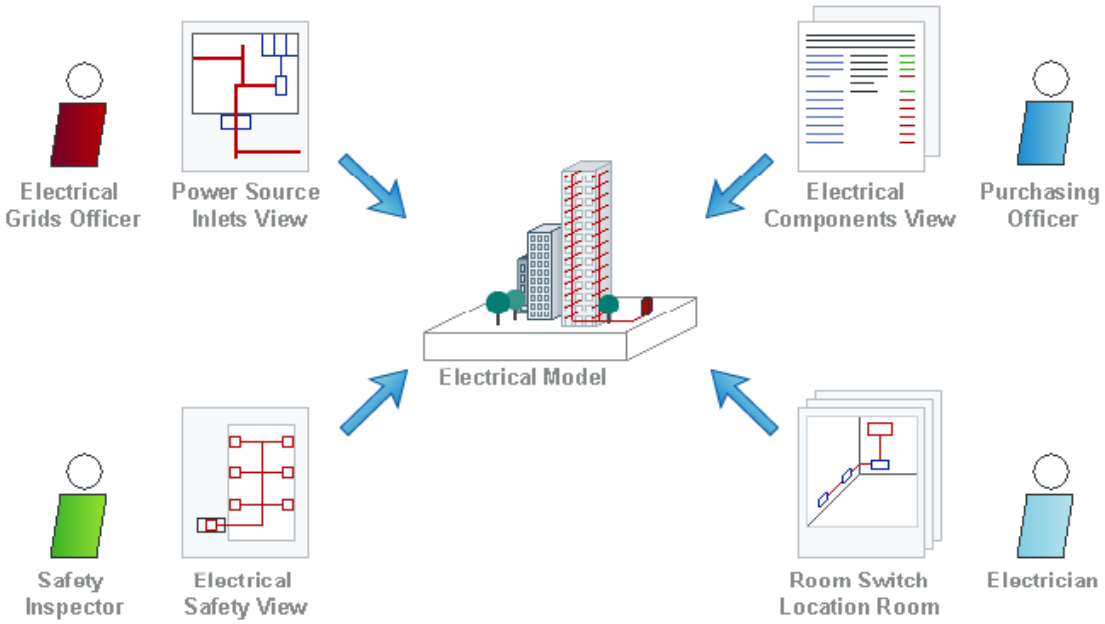
建筑物是A复杂的结构，具有许多不同的系统，包括结构、电气、通风、管道、景观等。



通过构建多个模型，我们能够创建每个子系统的简化视图，从而更容易理解建筑物的这一方面。模型本身也需要相互解决。例如，电气模型中表示的电力系统为通风系统中建模的空调设备提供电力至关重要。人类用途模型需要与景观模型进行对比，以确保园林景观满足居住者及其游客的休闲需求。

A模型通常由许多不同的利益相关者查看，这些利益相关者通常在被建模的系统部分方面具有完全不同的角色。为了确保对特定利益相关者模型者有用，可以创建视图来表示从特定角度查看模型时所看到的内容。

建筑物的电气模型对许多不同的利益相关者都很有用，他们对系统都有不同的看法，包括电网官员、安全检验员、电工和采购员。


Electrical Grids Officer
Power Source Inlets View
Electrical Components View
Purchasing Officer
Electrical Model
Safety Inspector
Electrical Safety View
Room Switch Location Room
Electrician

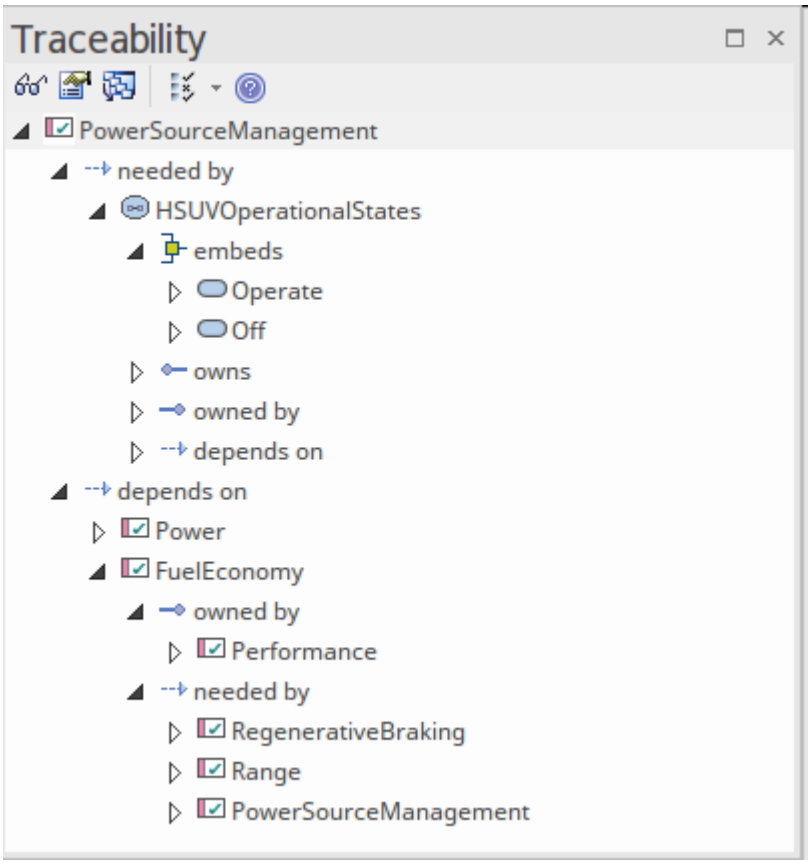
定义模型的目的

转向基于模型的系统为更习惯于使用以文档为中心的方法的团队带来了许多挑战和陷阱。可能最常见的陷阱是在没有清楚地理解或定义模型目的的情况下开始建模。

与以文档为中心的建模方法相比，定义模型的目的比定义一套文档的目的更困难。该模型比文档更有用和有效的数量级，可用于执行基于文档的系统无法想象的工作。基于模型的方法的一些优点是：

- 可以轻松应用一致性检查
- 选择视图可以轻松创建并保持一致
- 如果需要文档，可以自动生成
- 模型是互锁且一致的
- 更改影响可以可视化和自动化
- 模型可以与基础元模型保持兼容
- 模型可以版本化和基线化
- 需求可追溯性可以轻松管理
- 模型可以轻松操作和更改
- 该模型可用于生成代码和标准
- 可以模拟模型产生丰富的可视化
- 模型可以从一个抽象级别转换到另一个抽象级别
- 模型的部分可以很容易地重复使用创造效率

该图显示了如何在工具中可视化和管理可追溯性，允许您查看模型部分互锁的方式以及元素如何形成连接图，帮助您描述和理解您的模型。



Enterprise Architect利用 SysML 的强大功能，以及为系统工程经理、系统工程师和其他利益相关者构建的大量工

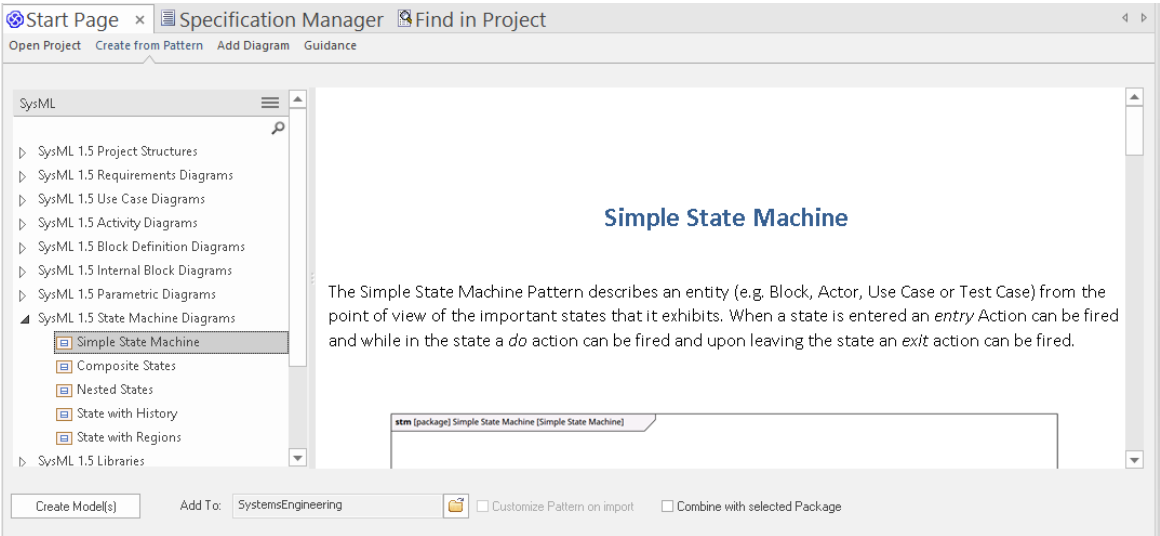
具，提供简单而有效的方法来利用基于模型的系统工程方法。

转向基于模型的方法还可以获得其他深远的好处，包括使用鼓励卓越和协作的工具确保项目和工作计划以严谨、高效和高效的方式执行。有关更多信息，请参阅Why Enterprise Architect?帮助主题。

决定从哪里开始

对于刚接触基于模型的系统工程的工程师来说，建模过程可能相当艰巨。显然，最重要的是从else开始建模的问题——工程师相当于艺术家的 空白画布”惯性。

Enterprise Architect为这个问题提供了一个受欢迎的解决方案，它提供了一系列可用于为倡议或项目创建起点的模式，包括所有 SysML 图表类型，每种类型都有许多模式。有关更多信息，请参阅The Model Wizard帮助，

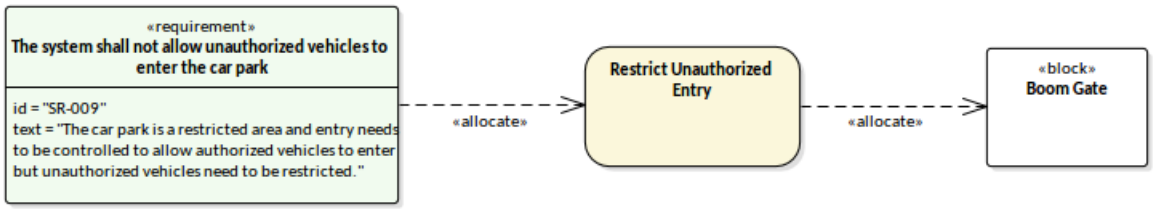


教科书通常描述了应按规定顺序执行的一系列步骤，但实际上这些方法不起作用，因为项目比书中描述的通用方法复杂得多，而且复杂的项目和资源依赖性意味着任务不能按规定的顺序进行。

起点通常由项目使用的工程方法或过程确定，可以是瀑布式、迭代式、两者的组合或其他类型的过程。无论使用哪种类型的流程，清楚地了解任务通常是一个很好的起点，定义受影响的利益相关者及其关注点和要求通常是一个很好的下一步。

连接部分模型

系统建模语言鼓励工程师创建一系列模型，对于基于模型的系统工程的新手或新手来说，这似乎是对系统视图的碎片化。实际上，SysML 描述了一个模型网络，每个模型都解决特定的问题，但连接在一起以将系统及其部分描述为一个整体。



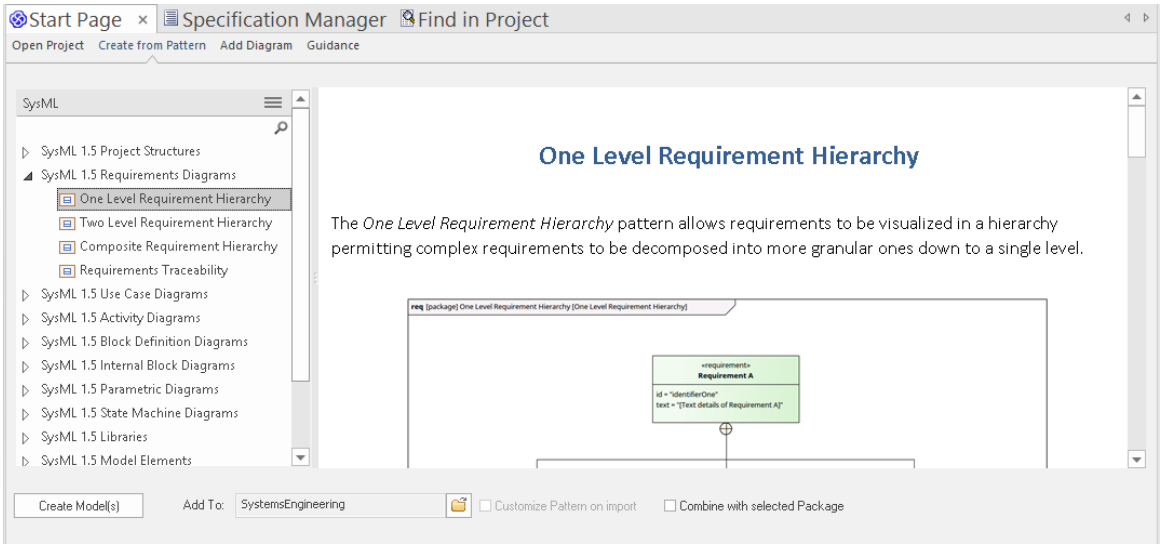
在这个插图中，我们看到了一个引人注目的Enterprise Architect图表，它使用专门的元素和连接器来描述模型各部分之间的连接，即需求、活动和使用分配关系的块。这些元素可以在任意数量的图表中重复使用，并且在一个位置更改它们的属性将在每个上下文更新它们。可以使用许多特征快速图表地创建图形，并且可以通过多种方式可视化它们，例如作为列表、库表和电子表格。可以过滤图表，并且可以用图形图标替换元素，从而为非技术观众创造更多兴趣。有关更多信息，请参阅Model Diagrams帮助主题。

这是基于模型的系统工程方法的真正威力，因为它允许以多种方式查看系统，从完成和高级视图到多层分解或层次结构。每个级别都是相互关联的，模型中的间隙或中断可以很容易地识别出来，并且可以找到补救措施。

确保模型的质量

模型的质量最终将反映在它所代表的系统的质量上。 Enterprise Architect旨在为创建和管理高质量的协作模型提供一个平台。有许多特征可以帮助建模者达到所需的质量水平，包括以下功能：

- **元模型**——可由用户定义以有效地为模型创建语法，确保用户创建 "合规"的建模语句；更多信息参见 Developing Profiles帮助主题
- 模型验证- 允许检查模型是否符合基础元模型
- 讨论和聊天- 允许建模者协作处理问题或解决方案
- 评论- 允许内部或外部专家查看和评论模型
- **模型模式**——提供专业创建的模型包，为建模提供起点
- **搜索** -帮助查找模型中的特定问题



此图显示了正在运行的模型模式，新手和有经验的建模者都可以通过按下一个按钮，使用行业最佳实践模型的高效语料库创建格式良好的模型和图表——所有这些都符合 SysML。还有关于如何使用该模式、在哪里获得进一步帮助等的详细解释和讨论。有关详细信息，请参阅The Model Wizard帮助主题。

系统建模语言（SysML）

系统建模语言（系统工程工件或程序的语言和解决方案。

SysML 旨在为各种系统工程问题和解决方案建模提供简单但有效的构造。它可用于多种用途，但在指定系统属性的要求、结构、行为、分配和约束以支持工程分析（包括参数分析和仿真）方面特别有效。 SysML 可以与多种流程和方法一起使用，例如结构化、面向对象、迭代、瀑布和许多其他方法。

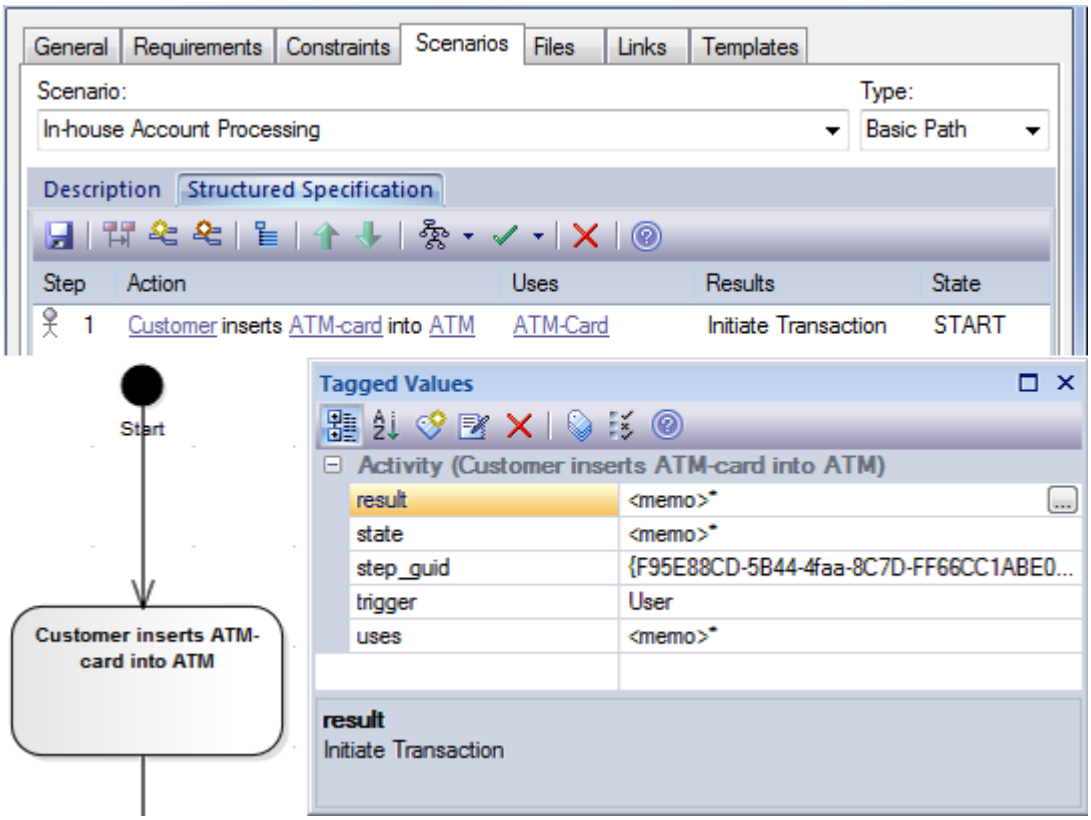
该语言经过十多年的设计和扩充，适用于日益复杂的建模系统。这些变化使相对紧凑和简洁的语言变得更广泛和多样化；尽管如此，大多数系统工程项目仍然可以使用语言的一小部分进行建模，我们可以将其称为 "Core SysML"。有关更多信息，请参阅 Enterprise帮助主题Modeling Systems in Enterprise Architect。

Enterprise Architect建模工具

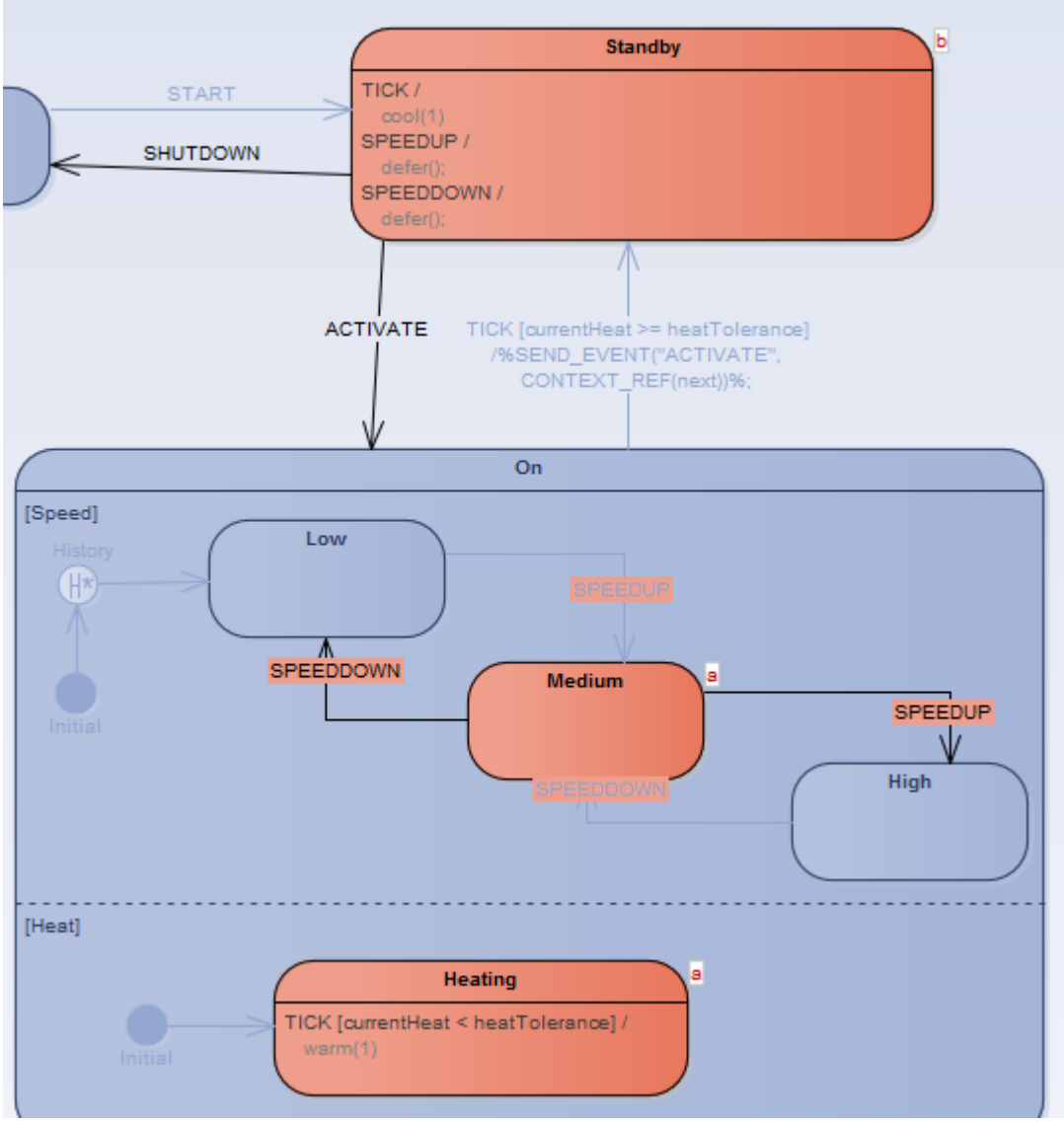
Enterprise Architect既是模型存储库又是模型协作平台，是系统工程项目的有效工具。它使团队成员（包括项目发起人、工程经理、客户和工程师）能够在严格且富有成效的环境中就项目进行协作。使用WebEA和Prolaborate ，协作可以在移动设备上继续进行，例如移动电话、平板电脑和笔记本电脑。

在信息和创新时代，除了存储信息或允许用户查看图表和模型之外，还需要一个工具来做更多的事情。Enterprise Architect已经接受了这一挑战，并通过以下工具将其系统工程产品提升到了另一个层次：

- *Scenario Builder* ，它自动从用例步骤创建活动图，并从 Scenario 生成测试案例；有关更多信息，请参阅帮助Scenario Builder帮助主题

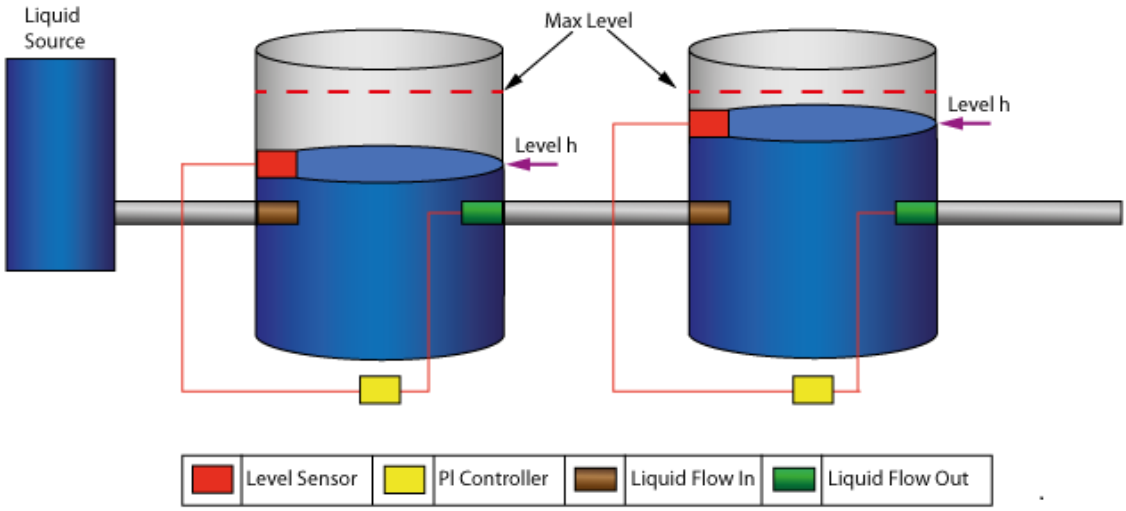


- 可执行状态机，允许从状态机自动生成编程代码；更多信息请查看Executable StateMachines帮助

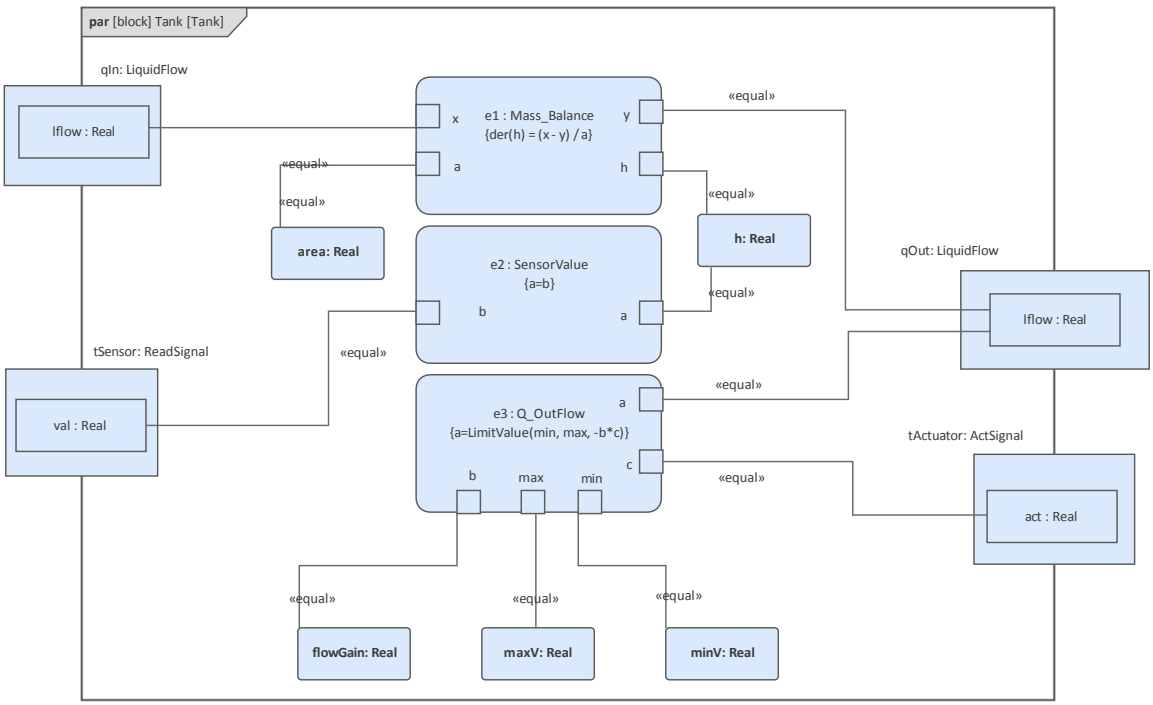


- 使用 OpenModelica 或 Simulink 进行参数仿真，使模型栩栩如生，并允许对复杂且通常难以处理的问题进行可视化和分析，以支持权衡分析和工程调查；有关更多信息，请参阅Simulation帮助 话题

在此示例中，定义了控制两个罐之间的流体流动的因素之间的关系：

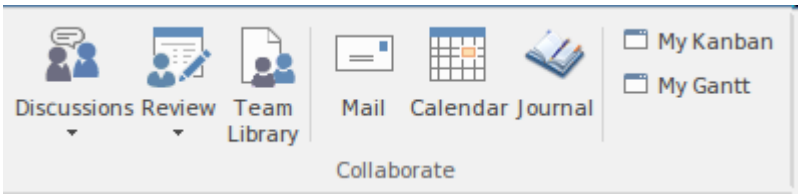


然后可以使用高级 OpenModelica 模拟特征对模型进行模拟。



协作平台

几乎在我们睡觉的时候，信息时代已经转变为创新时代，现在比以往任何时候都更需要团队以新的和有凝聚力的方式合作。在磁盘上共享文档和文件以及使用静态图表是我们现在期望在博物馆中看到的东西。响应式、稳健和创新的解决方案只能由团队使用卓越的工具来实现，这些工具不仅可以构建模型并促进协作，还可以执行工作。 Enterprise Architect是一个多功能工具包，它允许团队协作，汇集来自广泛互锁学科的最优秀的头脑和最有经验的双手。为更新和查看模型做出贡献的人可能在地理上分散，在不同的时区工作，来自不同的组织，甚至说不同的自然语言。

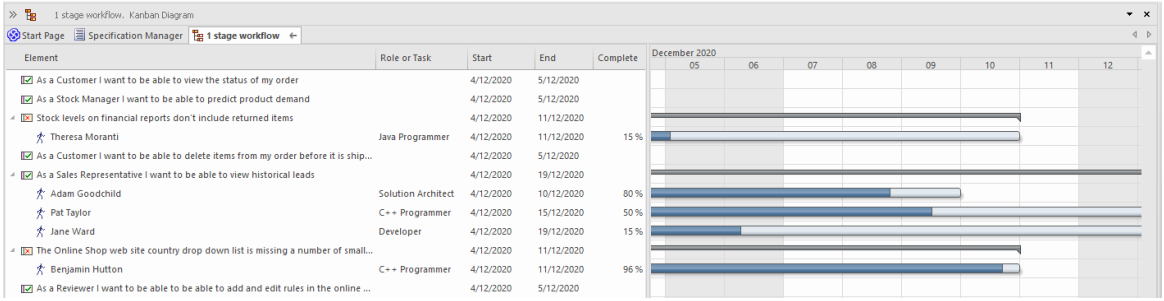


此图像显示了 开始"功能区中提供的一些有用的协作特征。 WebEA和Prolaborate也提供讨论和评论，允许建模和非建模人员协作，从而产生更强大和适合用途的解决方案。有关详细信息，请参阅Teams & Collaboration帮助。

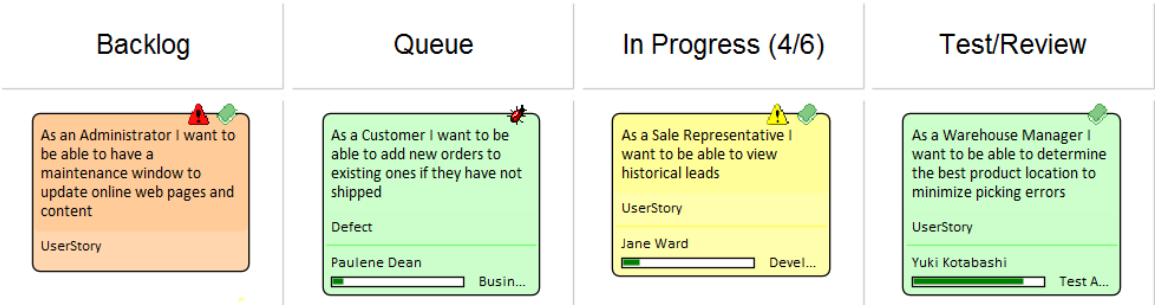
项目管理工作台

Enterprise Architect提供了广泛的工具来管理基于模型的系统工程项目。通过这种方式，它可以充当可用于管理工程项目的项目管理工作台。整个系统生命周期可以在该工具中建模，从业务需求的概念化到设计、实施、利用、支持，最终到系统的报废或处置。

有甘特图、日历、团队图书馆、风险、缺陷、任务、工作量和指标寄存器，仅举几例。路线图是另一个有价值的特征，它允许项目经理随时间可视化项目从当前状态到任意数量的过渡或未来状态的开发。



团队A可以使用内置的看板板协同工作，这些看板可以让需求、用户案例、缺陷和修改等项目在积极工作时可视化。资源分配和属性，如优先级和状态可以通过板项目查看，并显示溢出限制。



这种源自日本汽车行业的成熟技术已在Enterprise Architect中实施，其方式将大大提高您团队的生产力及其项目管理。有关更多信息，请参阅Project Build & Deploy 帮助话题。

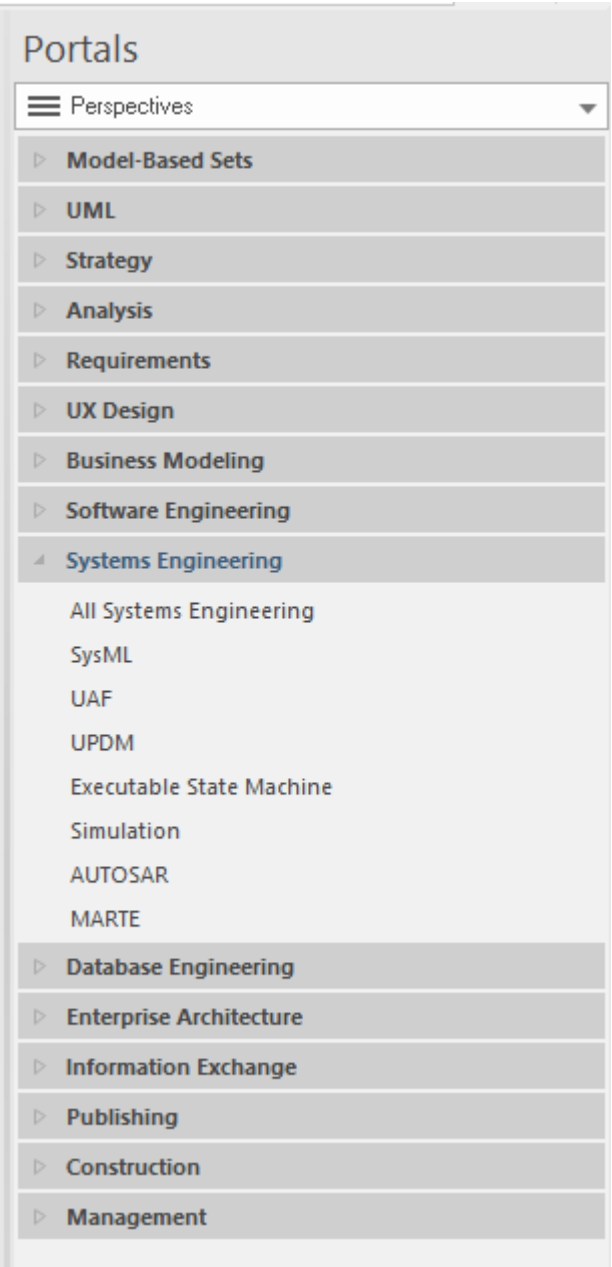
模型存储库

Enterprise Architect主要是一个模型存储库，它允许模型从创建到退役进行管理。存储库存储在关系数据库中，该数据库可以托管在客户端服务器配置中或作为云服务功能的一部分，无论是在云环境中的内部还是外部。因此，即使建模者将使用图表和可视元素，这些图表都已编码并存储在存储库数据库中。存储库可以包含任意数量的模型，并且可以组织起来以供重用以及用于企业和项目模型。有关详细信息，请参阅The Model Repository主题。

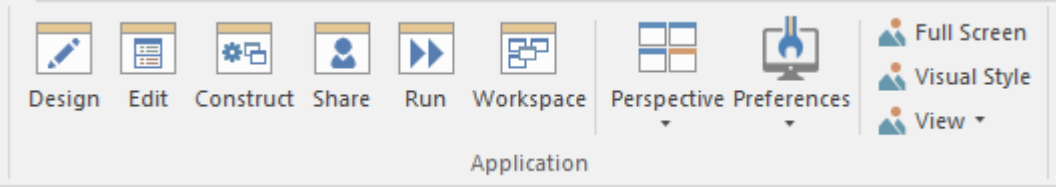
开始

开始使用新工具通常是最困难的挑战之一，但Enterprise Architect提供了许多功能来帮助新手使用该工具，从而使这一切变得容易。 Enterprise Architect是一个大型、多功能的应用程序，它的覆盖范围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。

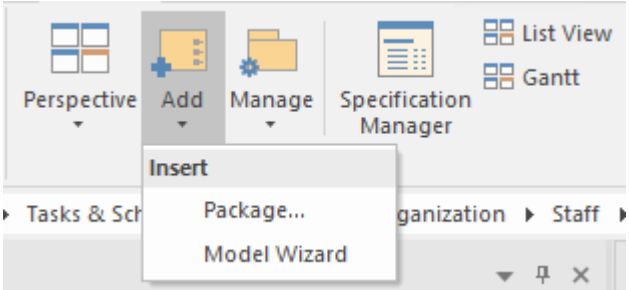
蓝图可用于将功能限制在特定领域，例如系统工程，让系统工程师或经理轻松系统。 A仍然可以使用其他可能有用的功能，例如策略建模、思维导图、代码工程等，只需更改蓝图，无需打开其他工具。值得注意的是，蓝图存在于Enterprise Architect支持的广泛建模学科中。更多信息请参见Model Perspectives帮助。



A在通过设置偏好和选择工作空间和视觉风格来定制他们自己的环境和用户界面方面也具有极大的灵活性。有关更多信息，请参阅Advanced Customization帮助主题。



使用模型模型（附带文档）可以直接创建一个新项目，该模型向导可用于自动创建模式项目结构以帮助您入门。 模型向导（首页）选项卡 从模型创建"然后可以用于创建任意数量的模式图，并随着问题和解决方案空间的发展而充实。

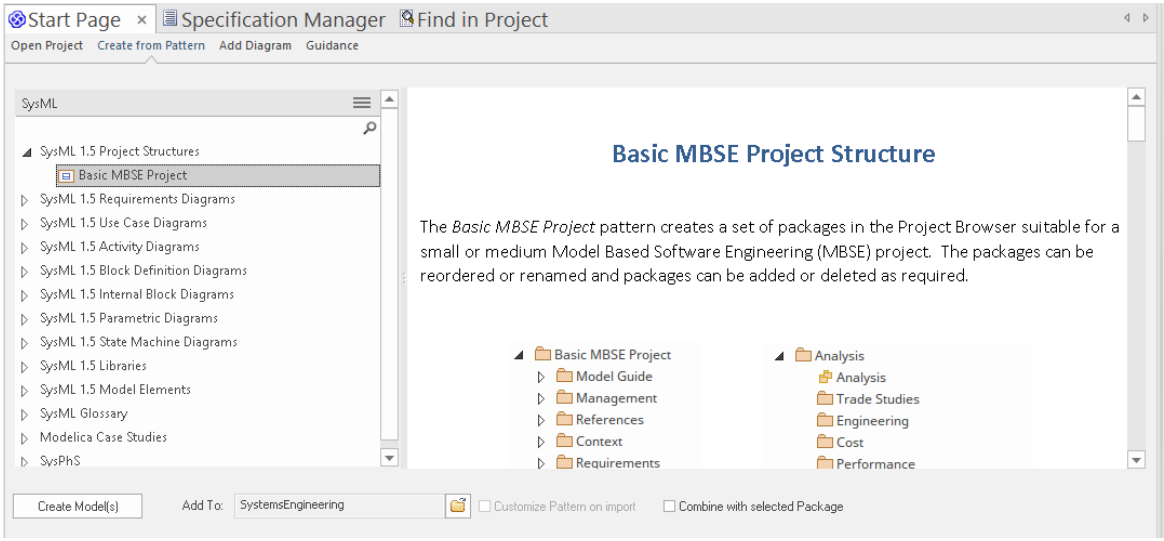


所有这些功能使新人上手变得容易，帮助他们成为团队中富有成效的成员，并迅速、毫不拖延地开始为模型做出贡献。与A基于文本或其他更基本的建模工具相比，新手工程师会惊讶于他们的生产力。在将自己和工具推向新极限的过程中会遇到挑战，但深入的帮助系统、庞大的用户社区、综合论坛、社区站点和一流的支持服务将使旅程变得轻松且内容丰富.

设置模型结构

Enterprise Architect已被设计为从头开始的生产力工具，并且设置模型结构 - 有时对于初学者来说是一项艰巨的任务，对于有经验的用户来说是一项繁琐的任务，在Enterprise Architect模型向导通过使用模型模型（首页 从模式创建”选项卡）。

可以使用模型向导包为导入的项目创建一个新的计划（项目）结构，该项目将创建一个可以定制的整个项目结构，提供所有准备启动



存储库的结构是稍后将探讨的主题，因为它对于基于模型的系统工程方法的成功系统工程。稍后我们将了解到，包是组织和维护模型库的重要单元，并且有一个完整的主题专门讨论使用包构建库的主题。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，将工具转换为适合用途，这对我们来说是在基于模型的系统工程项目上进行协作。

我们将研究一些可以将应用程序从通用建模工具更改为系统工程工具的地方。我们将研究这些主题。

选择蓝图

选择蓝图类似于在光学镜头上放置滤镜。它允许建模者只看到与该功能相关的应用蓝图——在我们的例子中是系统建模语言（SysML）。

选择工作空间

选择工作区很重要，因为它允许用户确保窗口、功能区、工具栏和其他可视元素提供高效的工作环境并轻松访问所需的重要功能。

设置视觉样式

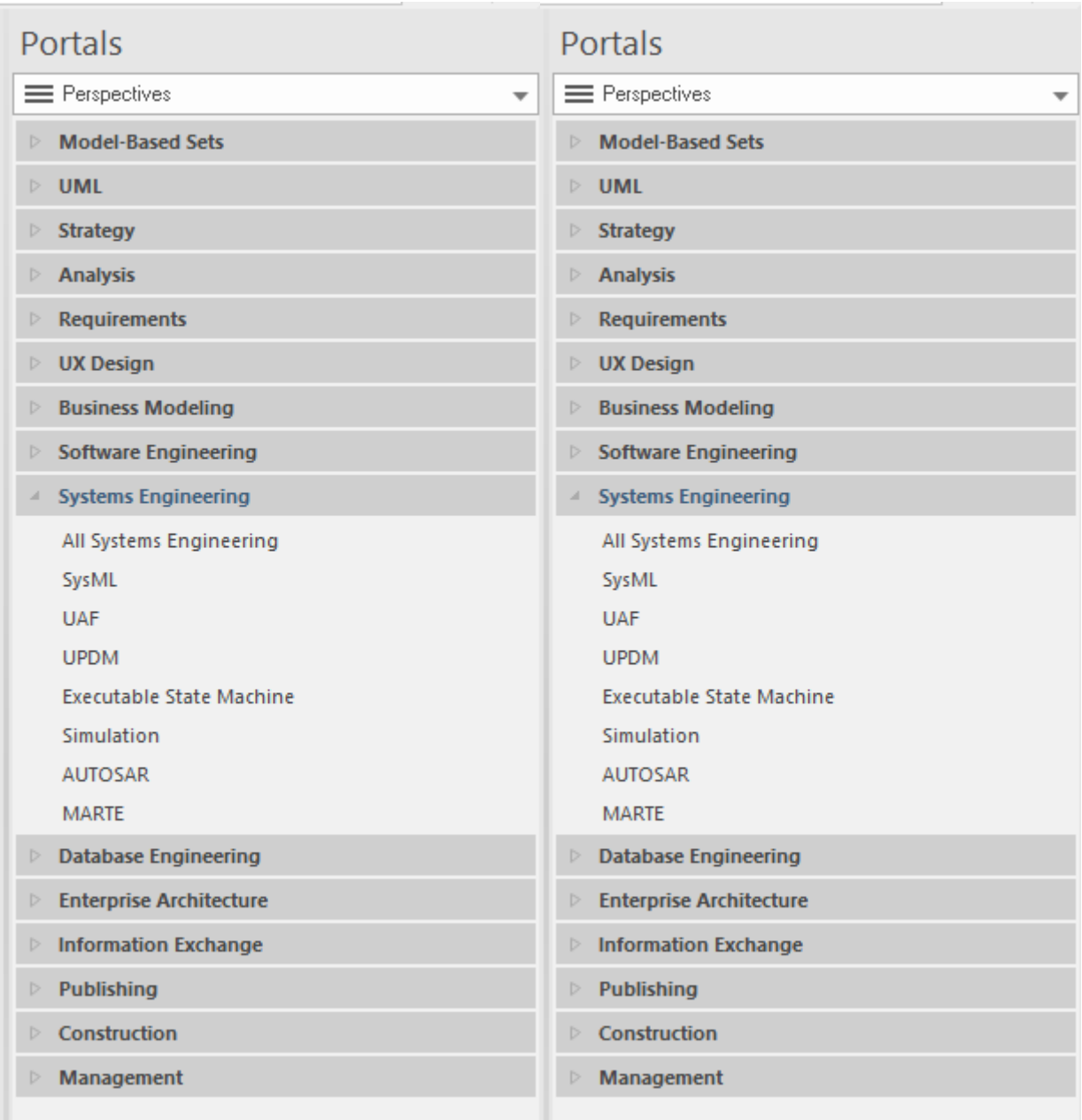
视觉样式为应用程序的外观和感觉提供了一系列选项，包括颜色和选项卡位置等。

设置首选项

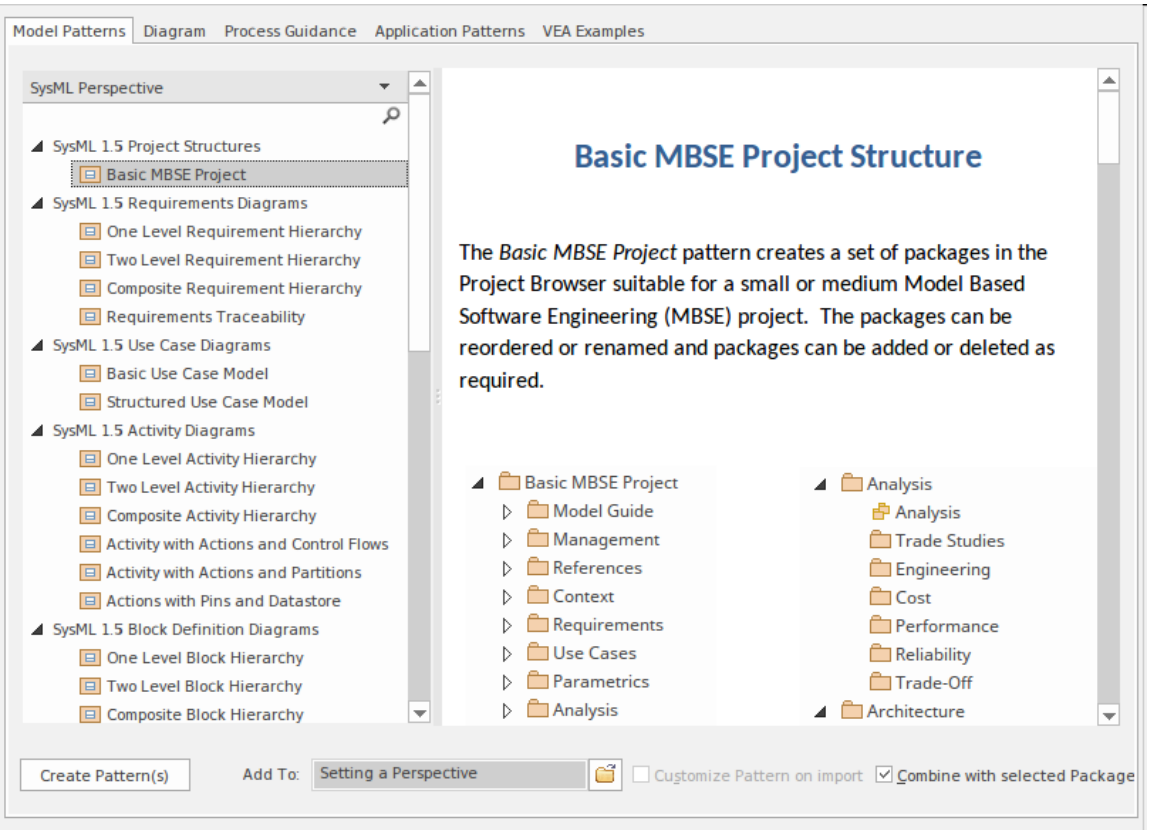
首选项提供了广泛的选项，用于定制Enterprise Architect的外观和功能，从常规设置（例如浏览器窗口选项）到图表、对象和工程选项。许多选项适用于单个用户，而其他选项则适用于整个存储库。

设置蓝图

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的特定子集特征和功能.蓝图的系统工程组为系统工程师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，您只需更改蓝图，工具就会更改为在选定区域提供聚焦。



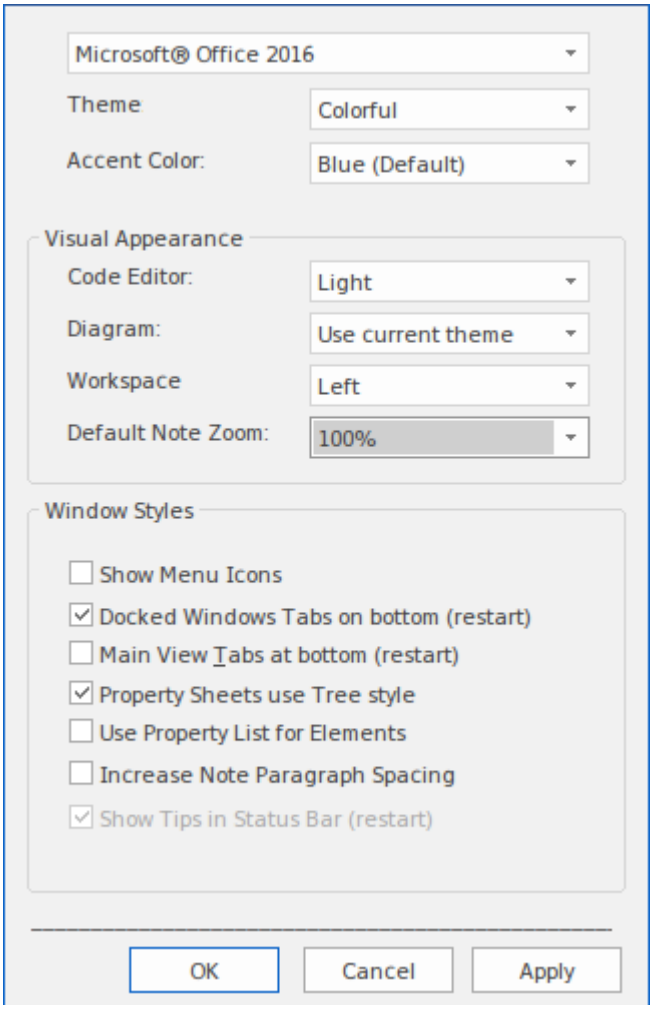
选择其中一个系统工程蓝图将更改工具以聚焦于系统工程的选定方面。例如，选择 SysML蓝图将显示一系列模型模式，通过能够加载标准模型片段或图表的模式，为用户提供快速入门。 新图表"对话框也将只显示 SysML 图表类型。



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 SysML 图的建模者添加其他功能，例如战略模型、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。

选择视觉风格

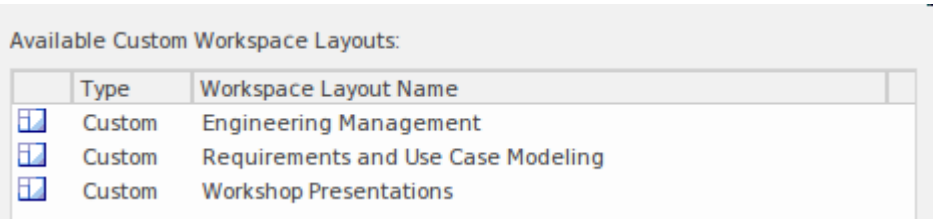
每个建模者都会对用户界面的配色方案和风格有自己的偏好，企业架构师允许为每个用户设置和保存这些，使应用程序更具吸引力。例如，一些建模者会想要深色方案，而其他人会喜欢浅色或彩色方案。



这里有一系列选项，包括设置主窗口选项卡的位置、注记窗口中文本的大小等等。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。

选择工作空间

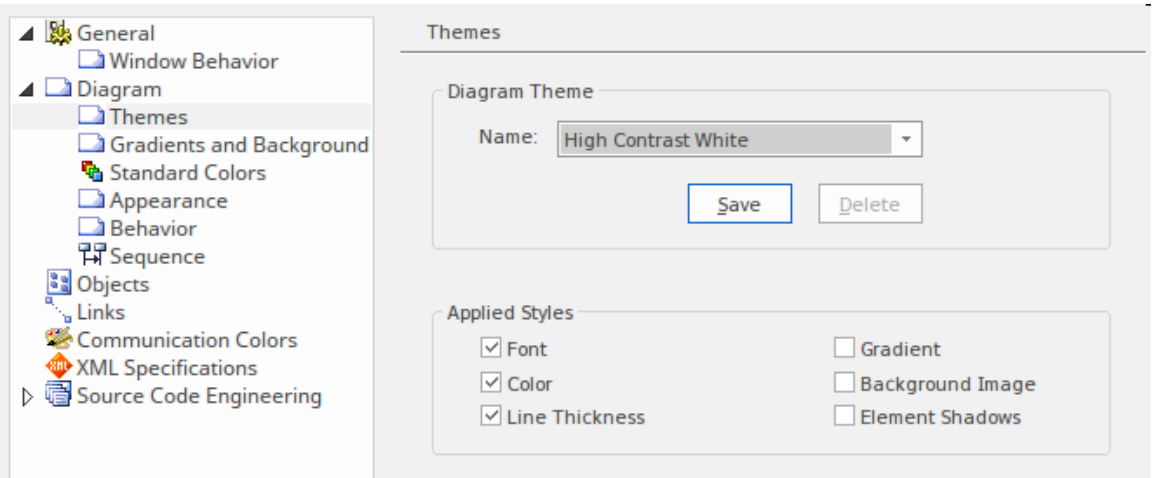
Enterprise Architect有一种快速更改用户接口布局的有用方法，以促进特定任务或工作方式。这是通过简单地选择将更改可见窗口和工具的工作区来实现的，以提供适合任务的最有效的工作方式。例如，为系统工程模拟定义了一个工作区，一个用于用例建模，另一个用于测试。您还可以定义任何数量的您认为有用的工作区布局，方法是打开窗口和工具并将它们放置在有助于处理特定任务或一组任务的排列中，然后保存它们。在此示例中，建模者定义了三个自定义工作区布局。更多信息请参见Workspace Layouts主题。



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工程师或整个团队。有关详细信息，请参阅User Preferences主题。

该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。



导入现有材料

当Enterprise Architect已经设置好供使用时，您可能仍会拥有一些现有的项目工件，这些工件以图表、文档、电子表格和其他格式的项目的形式存在。其中许多可以方便地导入Enterprise Architect或从工具中引用。该工具还通过DOORS需求管理专业云服务器（单独许可的服务器端组件）。

导入电子表格

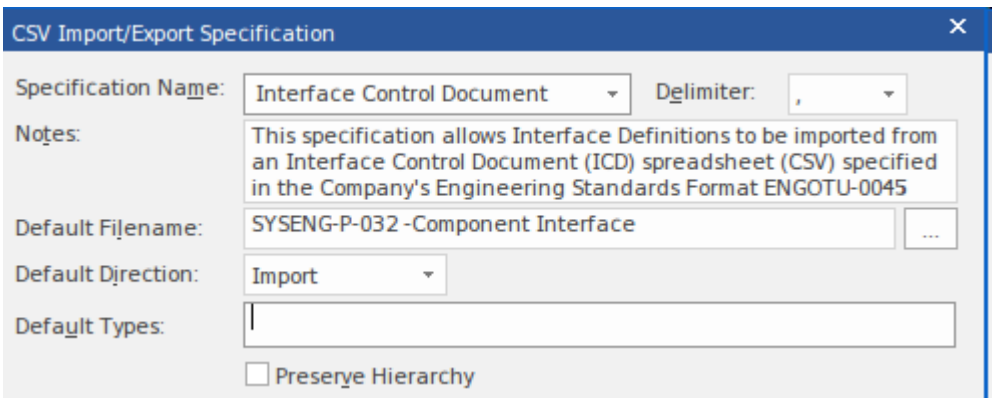
电子表格通常用作各种一般和文本项目数据的通用容器。用途包括：

- 需求
- 利益相关者分析
- 规划
- 路线图
- 子系统
- 成分
- 接口定义
- 任务管理

虽然电子表格是一种非常熟悉的工具，但它缺乏Enterprise Architect等信息管理平台的许多严格和有用的特征，包括：

- 协作，
- 绘图
- 可追溯性，
- 基线
- 可视化，
- 模拟
- 版本等等。

Enterprise Architect内置了对所有这些和许多其他形式的信息的支持，这些信息通常存储在电子表格中。该工具还方便地附带使用功能文件格式导入和导出电子表格数据的功能。



最典型的场景是将电子表格中的信息导入Enterprise Architect ，然后可以停用电子表格，然后可以在Enterprise Architect中管理信息资产。

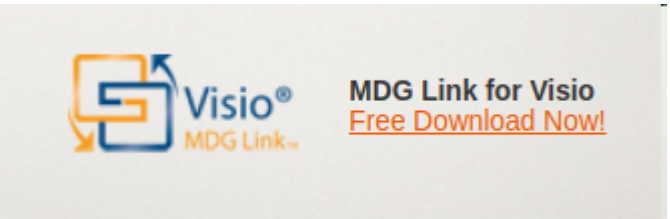
在某些情况下，使用电子表格的数值分析方面会很有用， Enterprise Architect方便地提供了一个工具来将信息导出为 CSV 文件格式以导入电子表格。有关详细信息，请参阅Import and Export Spreadsheets主题。

导入 Visio图表

Microsoft Visio 经常被工程团队使用，通常是因为没有其他更复杂的工具可用，它的目的是创建一般图表。对于采用Enterprise Architect作为他们选择的工程平台的团队来说，拥有一系列预先存在的 Visio 图表是很常见的。所有这些图表都可以导入Enterprise Architect ，但是当这些 Visio 图表以一致性或使用标准行业调色板而不是没有共同意义的自由形式的几何形状（如正方形和圆形）构建时，结果会更有效。导入后，可以对图表进行按摩和更新以形成存储库的一部分，并且可以停用原始 Visio 图表。

Enterprise Architect提供了一个免费工具，可用于连接到 MS Visio 引擎并将选定的图表导入存储库。

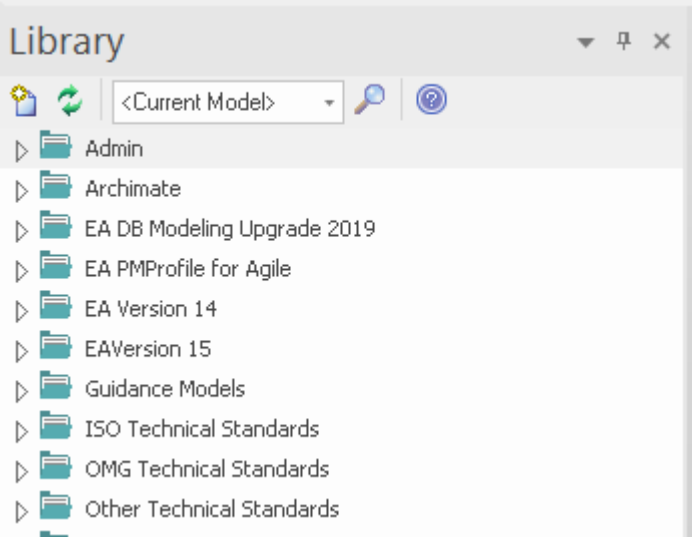
A决定是否停用 Visio 中的图表并允许Enterprise Architect从此时起管理图表。将图表放在存储库中提供了强大的功能，因为图表上的元素可以与存储库中的其他元素相关联。有关详细信息，请参阅Extensions - MDG Technologies主题。



使用团队图书馆

即使工程团队已经过渡到基于模型的系统工程，很可能仍然会有一系列对工程解决方案的管理和开发至关重要的文档和基于 Web 的材料。 Enterprise Architect提供了一种实用的方法来满足这种需求，方法是结合团队图书馆特征，可以将文档和 Web 资源（本地和远程）收集在一起作为参考。

任何编目在团队图书馆工件在图表中。但也可以导入或更方便地引用它们有关更多信息，请参阅The Model Library帮助主题。



Microsoft Office集成

Enterprise Architect能够使用 Microsoft Office 的MDG Link 与 Microsoft Office 应用程序套件集成，从而可以轻松地在任何Enterprise Architect模型和 MS Powerpoint、 MS Word 和 MS Excel 之间交换信息。有导入、导出和同步内容的选项。

Microsoft PowerPoint

PowerPoint 集成提供了对 PowerPoint 演示文稿中Enterprise Architect模型库的轻松访问。您可以插入对模型的引用、使用超链接的模型元素名称、插入图表作为图像并在幻灯片上列出包内容。

Microsoft Excel

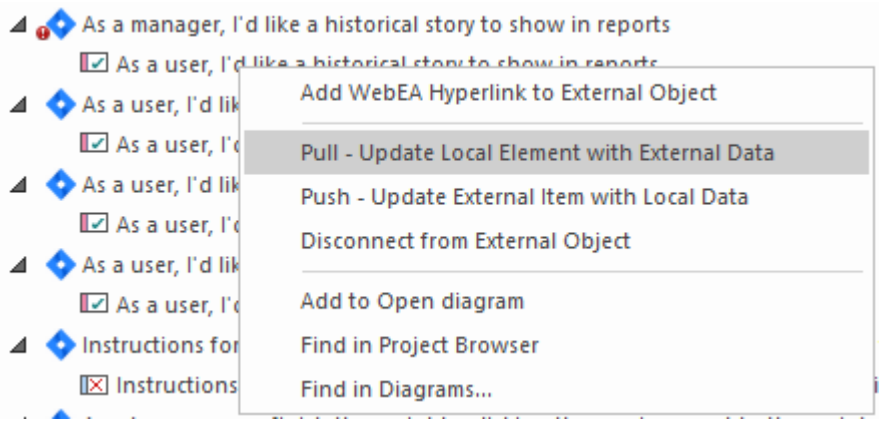
Microsoft Excel Importer 工具允许您将Microsoft Excel工作簿中的内容作为模型元素导入Enterprise Architect 。这包括将电子表格数据作为UML元素、连接器、属性和操作导入。

Microsoft Word

Microsoft Word Importer 工具将Microsoft Word文档中的需求、使用案例、流程、类和其他数据作为模型元素引入Enterprise Architect。 Microsoft Word Importer 提供了一种循序渐进的方法，可帮助您将部分、库表和分隔的名称-值对等项目映射到Enterprise Architect属性，包括定义自定义标记值。

与外部工具集成

Enterprise Architect提供了一个接口（作为专业云服务器的一部分），用于将您的模型存储库连接到外部工具。这使Enterprise Architect能够将外部工具中的元素与Enterprise Architect中的元素视图同步，如果Enterprise Architect和另一个工具对特定类型的信息有共同兴趣，这将特别有用。一个例子是与DOORS Next Generation (NG) 产品的集成，其中DOORS中建模的需求可以在Enterprise Architect中查看，元素的本地代理可以放置在图表上并与任意数量的其他建模元素相关，包括策略、贸易研究、使用案例和组件。（还有一个模型内功能可用于连接到旧版本的DOORS。）



有广泛的集成可用，团队可以使用作为专业云服务器的一部分提供的生命周期开放服务 (协作)功能创建自己的集成。有关更多信息，请参阅Integrate Data from External Providers帮助主题。

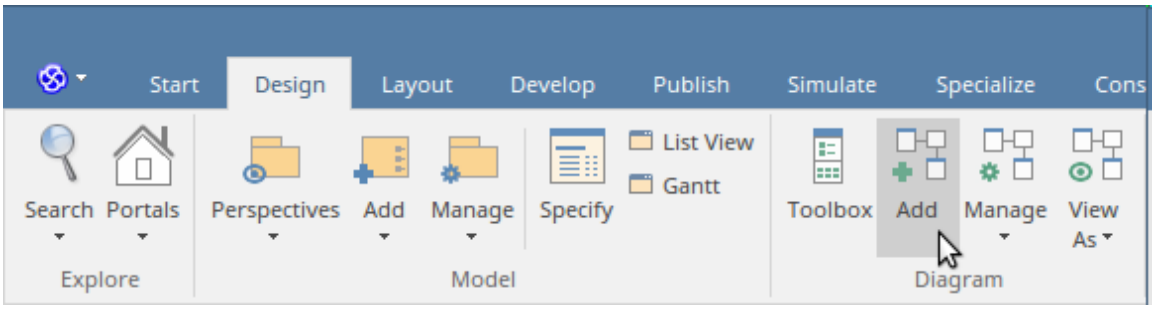
创建图表元素和关系

一旦设置了模型并根据您的需要定制了应用程序，包括选择 SysML蓝图和适当的工作区，您就可以开始创建元素和图表了。虽然可以在不首先创建图表的情况下创建元素，但通常的做法是首先创建一个图表作为元素如何可视化的画布。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，您可能正在定义系统的基本架构并定义了一个名为 子系统"的包。通过选择这个包，您是在告诉Enterprise Architect您希望在此处插入新的子系统图。

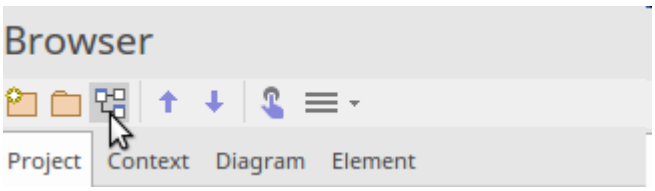
创建一个新的图表

Enterprise Architect是一个灵活的工具，提供了多种插入新图表的方法，包括：

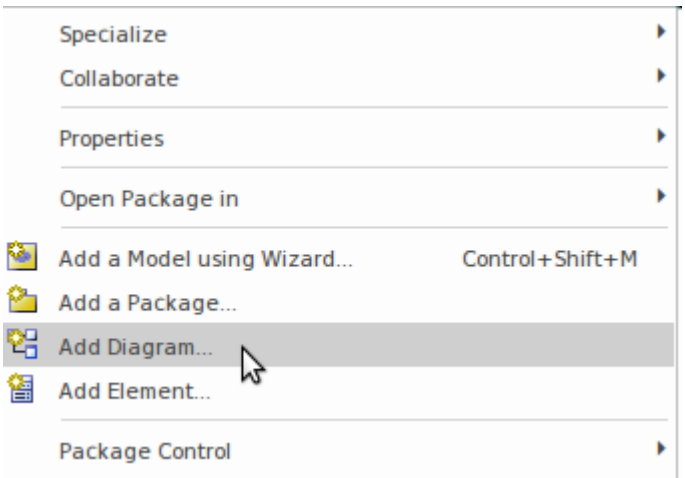
从丝带中选择：



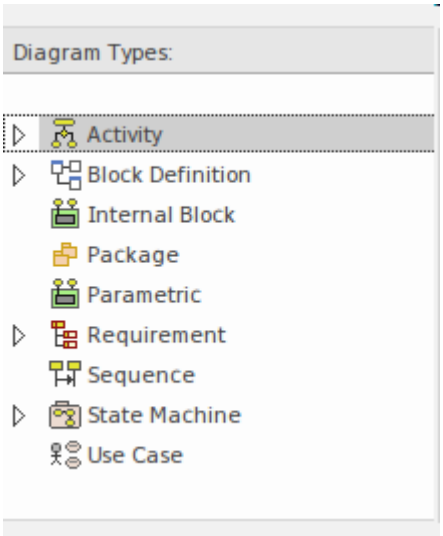
从浏览器窗口标题栏中选择：



从上下文（右键）菜单中选择：

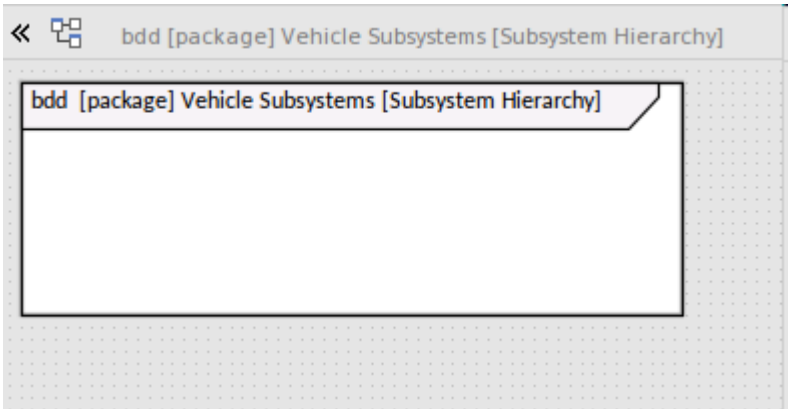


无论您选择哪种方法，您都可以从 新建图表"对话框的 图表类型"面板中选择图表图表类型。



让我们继续创建一个块定义图来表示子系统。选择块定义图作为图类型并输入适当的名称。单击确定按钮后，将创建一个新的（空白）图表图，并显示块定义工具箱，供您或您的团队成员创建元素和关系。

Enterprise Architect将创建一个图表画布，其中包含一个表示图表边界的可见框架。包含图表框架是因为一些用户更喜欢看到它，但它可以被隐藏而不会失去意义或合规性；一旦隐藏，画布就会变成框架，并且标题信息包含在画布的顶部。通过在 "首选项"对话框中选择此选项，可以将框架包含在已保存或已发布的图表中。



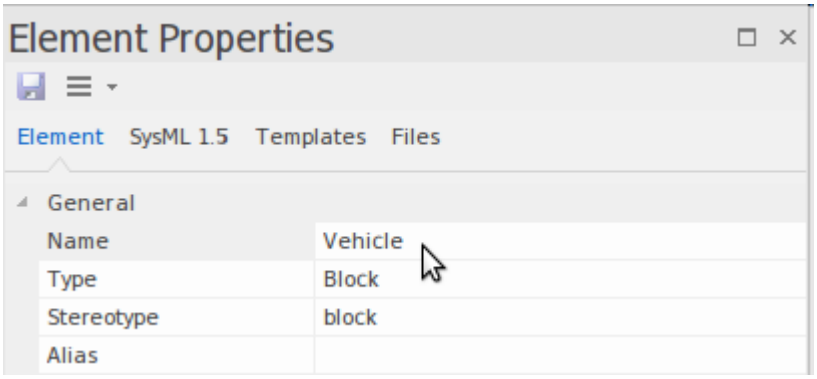
将元素添加到图表中

打开新图表后，您就可以开始创建元素和关系来描述子系统了。基本上有两种类型的物件可以添加到图表中：

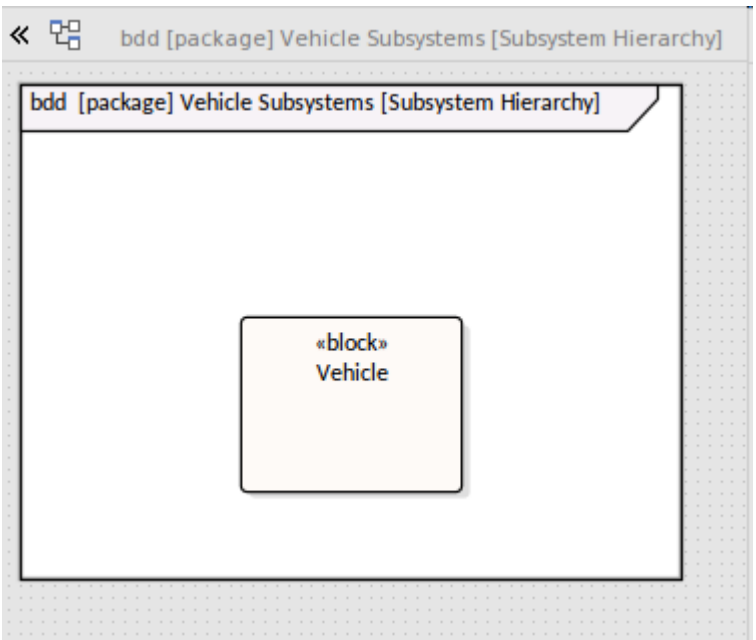
- 新元素 -通过从工具箱中拖动一个项目并将其放到图表画布上来创建
- 现有元素 -通过从浏览器窗口拖放元素放置在图表上

如果您正在开始一个新项目并且刚刚设置了存储库，那么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，使用第二个选项并从浏览器窗口中拖动现有元素将变得越来越普遍。

我们将创建许多块。首先，我们需要一个块来表示整个车辆，所以我们将一个块项从工具箱到图表画布上。该工具将调整框架的大小以包含块，无论您将其放置在画布上的什么位置。该元素将被赋予默认名称 "Block1"。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 "Block1"将元素的名称更改为 "Vehicle"。



这将更改浏览器窗口和图表中元素的名称。返回图表，您将看到新添加的块，其名称为 车辆”，包含在图表框架中。



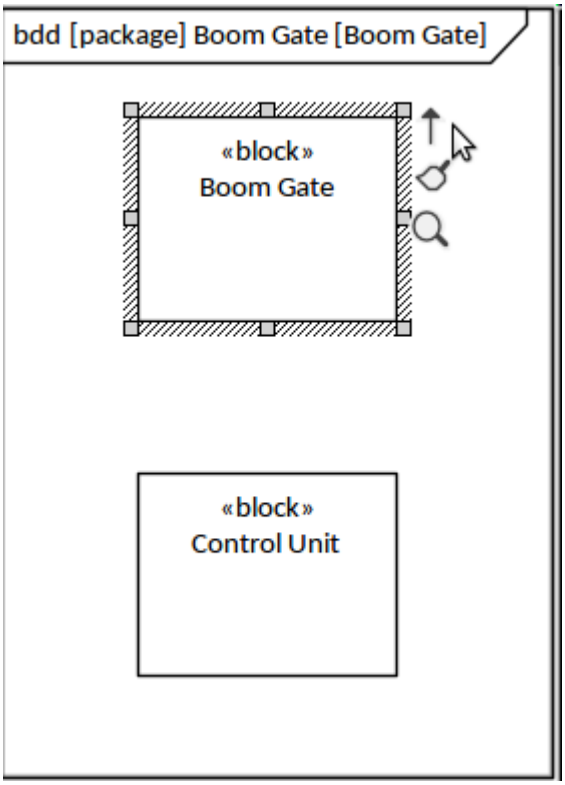
我们现在可以使用相同的方法添加一系列块来表示每个子系统。

将关系添加到一个图表

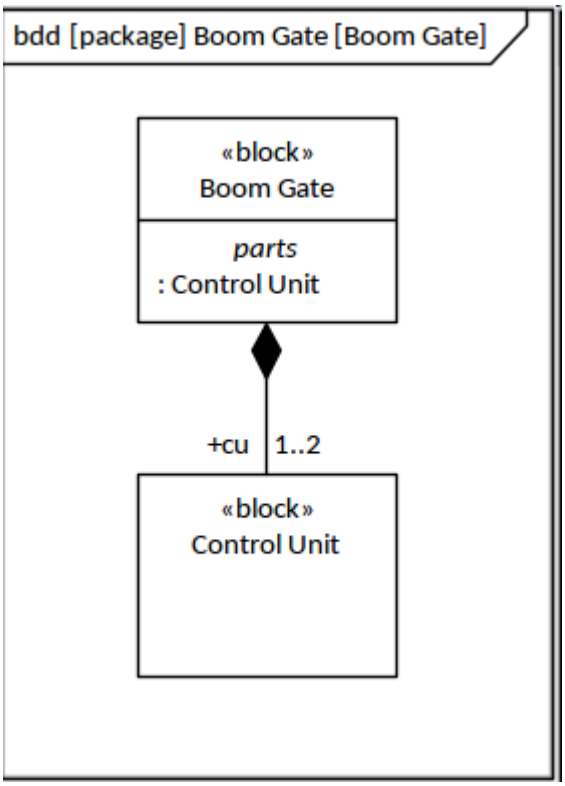
添加两个或更多元素后，您可以将它们与关系连接起来，这提供了模型中不同元素之间的语义粘合。例如，一个块元素可以使用部件关联关系连接到另一个块元素。有两种主要方式可以将连接器添加到图表中：

1. Quick Linker -通过拖动 *Quick Linker* 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。需要小心谨慎，以确保您朝着正确的方向前进；**例如**，部件关联关系应该从位于**部分末端**的块拖动到位于整个末端的块。这将确保关系末尾的小菱形标记位于正**确的末尾**，指示整个部分的关系。



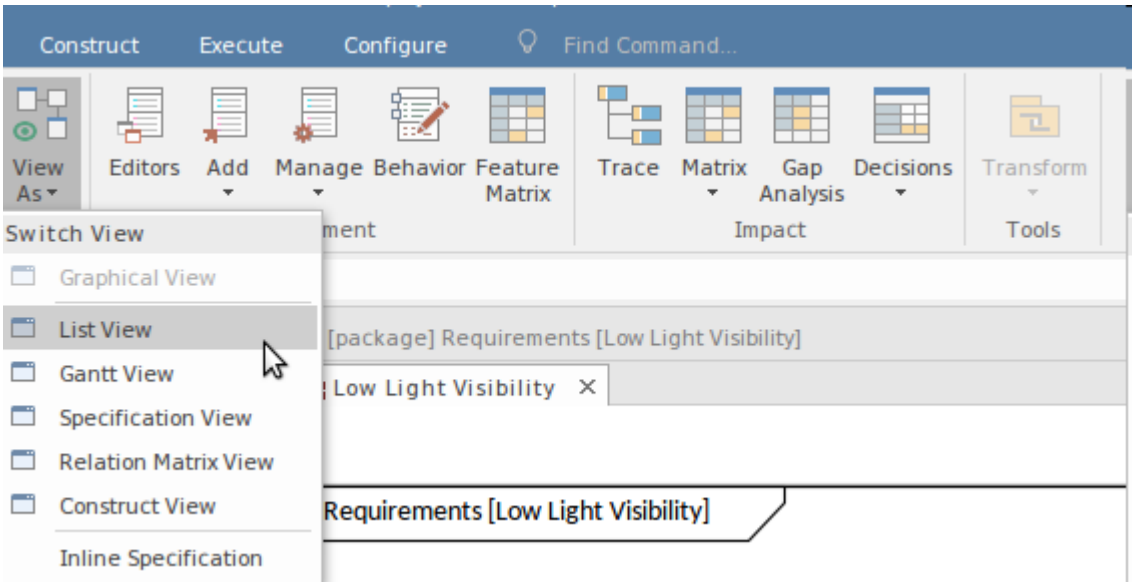
无论使用哪种方法，结果都将是连接两个块的信息需求关系。可以更改连接器的方向和样式，并且可以添加任意数量的路径点，以便随着模型的开发以不同的方式对其进行布线。此图显示了添加的关系，其中建模者还添加了角色名称 (+cu) 和多重性 (1 ..2)，表明 Boom门必须至少有一个控制单元，但最多可以有两个。如果建模者无意中将连接器添加到错误的方向，则可以通过访问连接器上下文菜单的高级子菜单中的选项来方便地反转它。



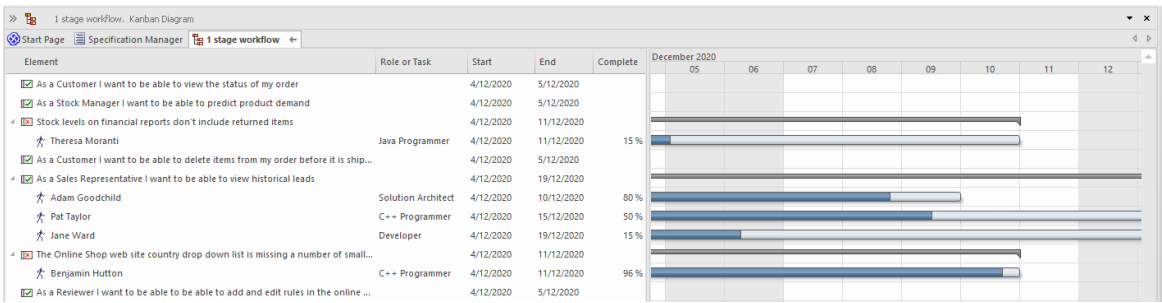
可视化模型

您可能是通过阅读教科书或作为高等教育资格的一部分或在线页面中的讲座注记中的介绍性材料而了解 SysML。大部分内容将 SysML 呈现为一种图表语言，但Enterprise Architect通过提供多种方法来可视化存储库中的信息，从而扩展了该语言的视野 - 以及一般的建模。图表仍然发挥着重要作用，但在许多情况下，有更引人注目的方式来可视化或使用存储库内容，包括：

- 规格视图- 为更熟悉这种方式工作的人提供电子表格或文档视图
- 列表视图- 类似于电子表格的基于表格的视图，可以查看多个元素的属性
- 可追溯性视图——可以查看到任何层次的元素关系的图形嵌套
- 差距分析矩阵- 显示架构模型中的源和目标元素和差距
- 关系矩阵- 基于网格的视图，具有包含元素集的两个轴
- 状态库表视图- 可用于状态机图表的视图，用于在表中显示状态转换
- 甘图表图视图- 查看资源分配后的时间线
- 搜索 - 查看符合内置或用户定义的一组标准的元素列表
- 出版物 - 使用内置或用户定义的模板查看出版物中的元素
- 图形视图- 以图表形式查看元素（这是默认视图）



此图显示了一个甘图表图，它是根据输入的资源信息自动创建的，该信息针对在看板图上可视化的元素。有关更多信息，请参阅Gantt View帮助。



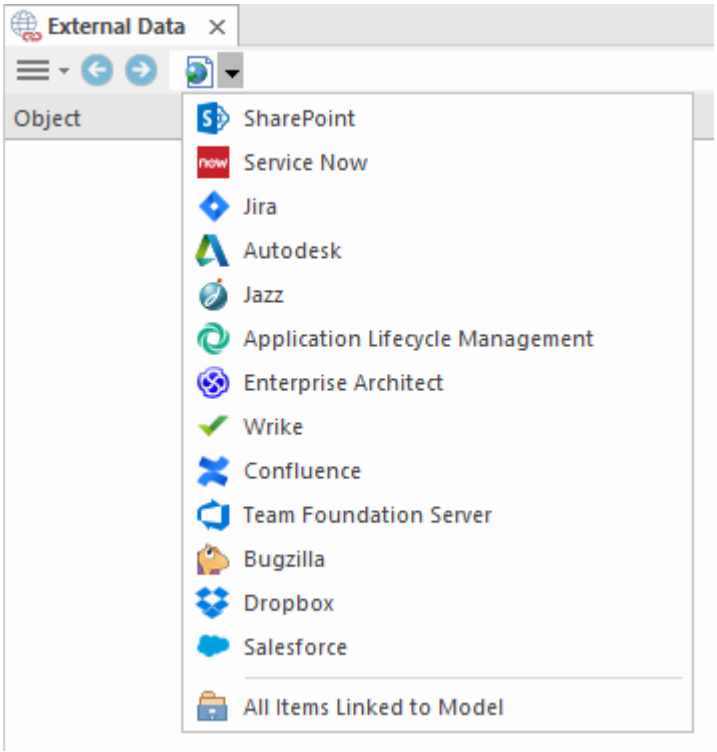
与外部数据同步

虽然Enterprise Architect是管理基于模型的系统工程项目的中心工具，但工程办公室可能会拥有或将获得许多其他工具，以确保努力实现其结果。这些可能包括项目管理工具、可视化工具、需求管理目录、配置管理系统和问题跟踪软件。

Enterprise Architect通过专业云服务器（一个单独许可的服务器端组件）提供与大量工具的双向集成，包括：

- Doors Next Generation - 用于需求管理
- Wrike - 用于一般目的的项目管理
- Jira - 用于问题跟踪
- ServiceNow - 用于配置管理
- Share Point - 用于文档管理
- 团队基础服务器——用于软件学科的版本控制

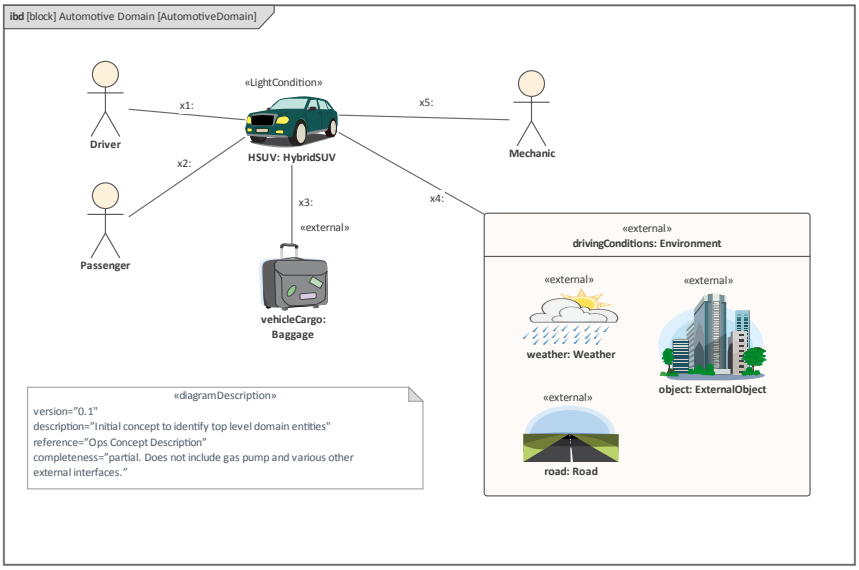
来自这些外部数据源的元素的表示可以包含在图表中并与存储库中的其他元素相关。这使Enterprise Architect成为一个工程信息中心，连接和集成各种工具以提供项目的单一视图。此图像显示了当前的提供者列表。



我们前进的方向

系统工程的学科可以追溯到 1900 年代初期；该术语被认为是贝尔实验室在 1940 年代初期首次创造的。 60 多年过去了，直到 2001 年才意识到对建模系统的专用语言的需求，但直到 2006 年，系统建模语言（SysML）才被物件管理组采用并成为首选语言用于描述系统。

在系统工程领域， Enterprise Architect凭借其灵活、可扩展和实用的复杂系统建模方法以及对系统建模语言规范的高度遵从性，已成为众多行业领导者的首选工具。

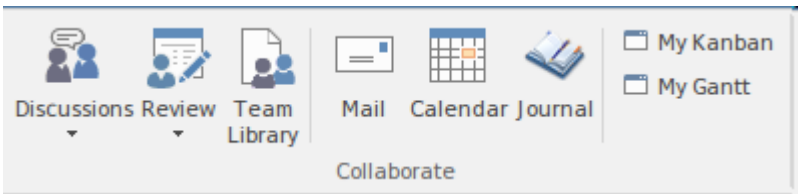


- 战略思想家，
- 高级管理人员，
- 工程管理，
- 系统设计师和架构师，
- 软件设计师和架构师，
- 系统工程师，
- 软件工程师，
- 制造团队，
- 实施团队，
- 技术支持人员。

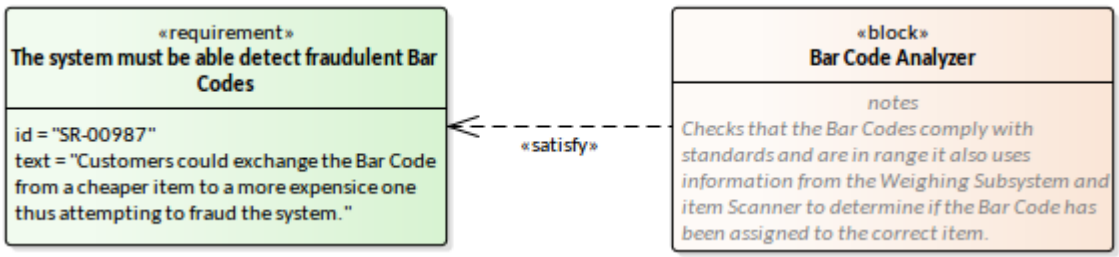
这种工作方式和使用Enterprise Architect作为工具的真正力量在于跨学科团队的统一。因此，允许来自广泛学科、可能在地理上分散并在从战略到支持的广泛层面工作的人们分享对正在开发的产品或系统的共同看法。

你会学到什么

本指南将教您如何使用Enterprise Architect特征丰富特性来开发和管理基于模型的系统工程计划、编写和阅读系统建模语言(SysML)、创建文档以及作为团队成员使用正式或非正式的工程过程。



您将了解可用的工具功能以及应该使用哪些工具来执行特定技术，并使用链接来帮助主题如何使用它们。例如，无论采用何种流程或框架，在某些时候都需要执行结构元素和行为元素之间的分配；本主题将描述该技术以及如何通过采用Enterprise Architect使用触手可及的工具和功能来最好地实现它。



文档概述

此表提供了本作品中包含的主题的列表和描述，并提供了材料概述。

具有四个变量的方程	本主题从数学角度将基于模型的系统工程描述为一个具有四个变量的方程，即：建模、 *Enterprise Architect*选择的工具、系统建模语言和工程进程。初等数学表明我们需要四个联立方程来解决这样的问题。本主题将介绍这四个变量，并建议您作为工程师或经理无需同时求解方程即可精通该学科的方法。 我们将看到Enterprise Architect如何在方程式的所有四个部分中发挥重要作

用，并且旨在帮助您的工程计划取得成功。

开始 本主题向该工具的新手介绍应用程序的最重要方面，包括设置模型结构、定制应用程序、使用图表、其他可视化内容的方式以及与其他工具的集成。它还包括关于用户界面的讨论，包括功能区和蓝图，这是使用该工具的基础。

该工具有一个广泛的帮助系统，该系统已经开发了二十多年，可以为工程师或其他利益相关者在使用该工具时可能遇到的几乎所有问题提供答案。

我们前进的方向 本主题描述了指南中的主题列表，并对每个主题进行了概述。

了解 SysML图表 本主题介绍图表作为可视化模型和存储库内容的首选和最常用方法。它对每个图表进行了简要介绍，并允许读者建立一个关于图表是什么以及如何使用它来表达自己的想法和解释他人的想法的思维导图。

系统建模语言概述 本主题提供对系统建模语言的解剖学和生理学的深入了解。它提供了有用的符号概要并描述了语言架构。它还介绍了基本概念，包括：元素、图表、模型和视图。虽然可以从规范本身收集到这些信息，但在本主题中总结了足够的细节，以保护新手无需涉足规范。

Enterprise Architect提供了示例模型和广泛的帮助系统，它们将作为有用的资源来帮助新手和经验丰富的从业者获得使用该语言的经验。

作为工程团队进行协作 本主题介绍了强大的协作工具，系统工程师、经理、客户、咨询工程师、监管机构和标准机构都可以为模型做出贡献。这包括讨论、聊天、模型邮件、团队库、评论等。

使用包构建存储库 本主题涵盖存储库结构的基本方面以及如何将其用作组织原则以确保存储库适合用途。它还涵盖了如何将包用作容器，既允许添加内容，又提供管理添加到每个包中的元素、图表和视图的属性的机制。

需求定义与管理 本主题向工程师和其他利益相关者介绍这一所有重要且核心的学科，这些学科构成了建模工作的所有其他方面的基础。 Enterprise Architect拥有一套强大且无与伦比的工具，用于开发、管理、可视化和记录需求，并介绍了这些工具，并给出了如何使用该工具的实际示例。

使用案例描述用户目标 本主题描述了一种以用户为中心的表达需求的方式，其中用户试图实现的目标是编写需求的基础。用户的目标在使用案例中进行了编码，这些案例仅在图表上表示，但其细节在Enterprise Architect中使用 Scenario Builder 进行了充实。此功能允许将描述、约束（如前置条件和后置条件）以及场景的步骤写入引人注目且高效的用户界面中。它还描述了如何从该工具自动生成行为图，以及诸如活动之类的元素可以链接到诸如需求之类的上游建模元素和诸如组件之类的下游元素——创建有效且有用的可追溯性。

使用模块构建和约束模型 本主题介绍的块是语言中结构的基本单位，也可用于模型约束。正是交互中可以有属性特征和属性点的原子语言和原子点来描述细节。本主题还描述了块与其他模型元素的关系，包括将块与活动相关联的最重要的分配关系。分配将 SysML 的两个重要支柱联系在一起：结构和行为。它还介绍了用于建模数量和值类型的设备，这些设备可用于对物理世界中的模型进行建模。

使用属性和部件来模型块用途 本主题继上一主题之后介绍内部块图，用于可视化在给定上下文。这些图显示了一个块的部分属性如何连接在一起。拥有块表示为图框，并且出现在块定义图上的部件连接部分关联的部件作为部件元素出现在内部块图上

协调行为与活动 本主题介绍活动和更原子的单元行动，它们都用于描述系统在不同级别的行为方面。这些最重要的元素等同于我们自然语言中的动词，并且与函数一样，在模型中具有组织功能。 Enterprise Architect有许多有用的设备，例如Simulations，它们可以使这些模型栩栩如生，并使复杂的现实世界问题得以

简化和可视化。

使用参数和模拟进行可视化

本主题探讨参数图与块定义图的结合使用，定义图定义模型数学方程的约束块及其使用的参数。该主题描述了如何在参数图上表示这些约束和参数，它是内部块图的表亲。还介绍了参数图的仿真，您将学习如何安装和使用Open内部Modelica 界面。您将了解如何在无需离开Enterprise Architect环境的情况下可视化高级方程图。

用状态机建模更改

本主题介绍状态机作为描述实体（如块）可以表现的离散条件（状态）的方法。这种行为装置可以为模型带来极大的清晰度并解决其他棘手的问题。Enterprise Architect有许多工具特征可以将这些模型变为现实，即状态库表和可执行状态机，它们允许状态和转换在引人注目的可视化中处于活动状态。

作为消息序列的交互

在本主题中，序列图被介绍为一种建模基于消息的行为的方法。您将学习如何使用此图来模型一系列工程概念。图表模型了系统内部的块之间的交互，或者系统本身与其环境之间的交互，它们可以用来模型用例的步骤发生的通信。研究生命线和激活，并使用片段介绍信息编排。

A一个示例SysML模型

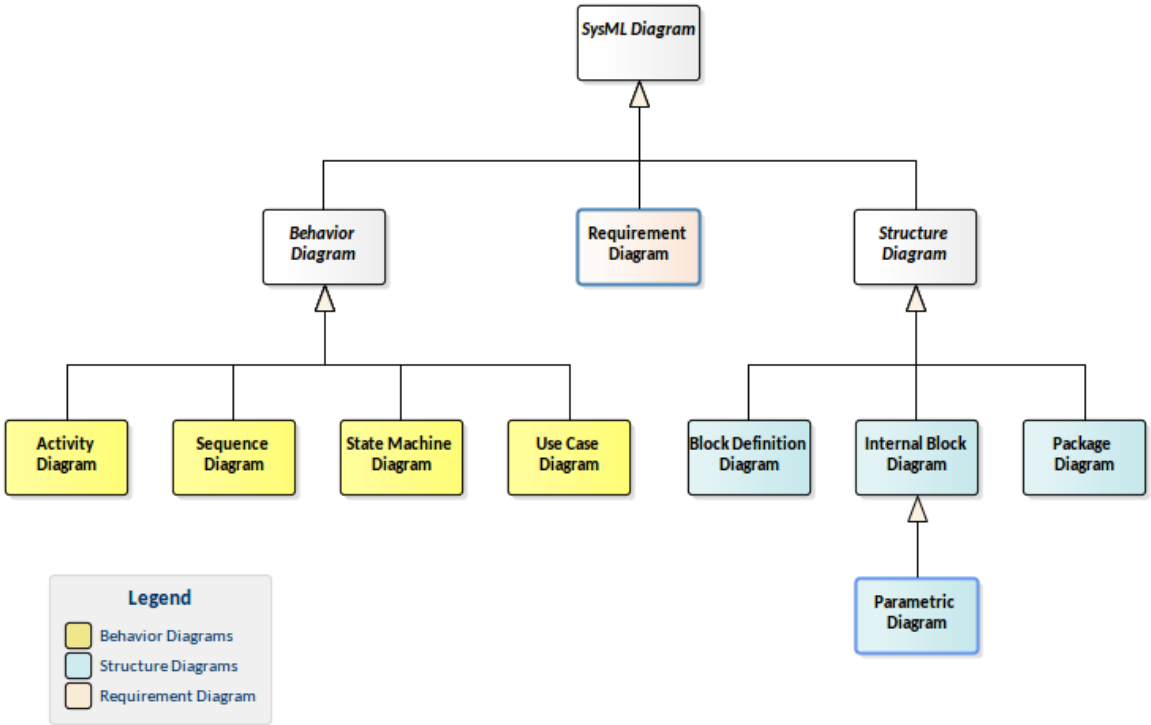
在本主题中，我们探讨了指定、设计和测试混合动力 SUV 车辆的图表，该车辆利用燃料和电力作为动力源，具有再生制动等机制以满足其燃油效率要求。示例模型发布在 SysML 规范的附件中；在本主题中，我们将探索在Enterprise Architect中创建的图表，解释重要的语言和工具特征。

认识系统工程工具

本主题介绍了使用系统工程模型的一些最重要的工具，描述它们是什么、在哪里可以找到它们以及如何使用它们，包括选项和在哪里可以找到更多帮助。还有许多其他建模者可能会发现有用的工具。这些在文件的相关点进行了介绍。

了解 SysML图表

SysML 的图表可以被视为画布类型，工程师将在其中创建构成模型一部分的工程概念的可视化表示。有九种 SysML 图类型，每种都专注于问题或解决方案的特定方面。虽然图表类型通常包含不同类型的元素，但它们都符合由以下组成的标准表示：包含的框架、标题和内容区域。



本节描述了在处理图表时可以使用的一些最有用的工具和特征，但还有许多其他有用的工具。有关更多信息，请参阅Model Diagrams 帮助话题。

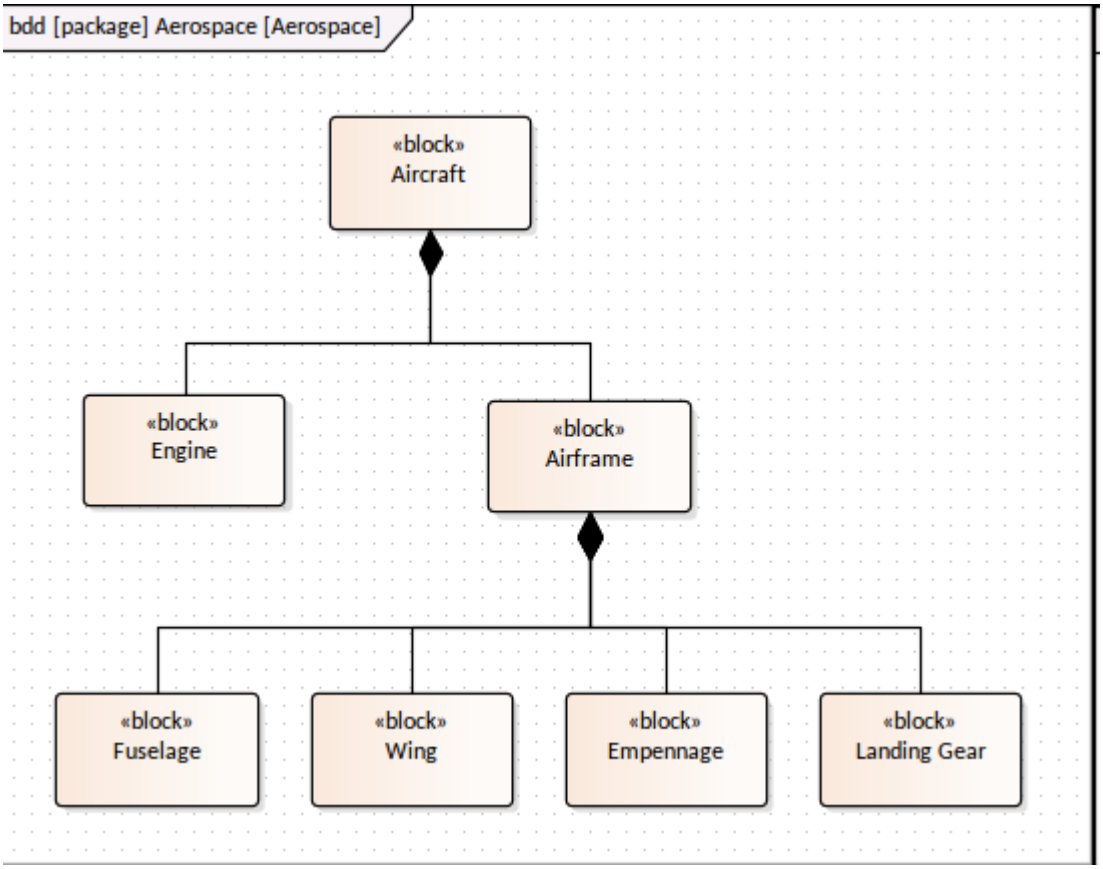
图表框架

图框是A在图表上包含元素和关系的视觉设备。框架有两个部分：

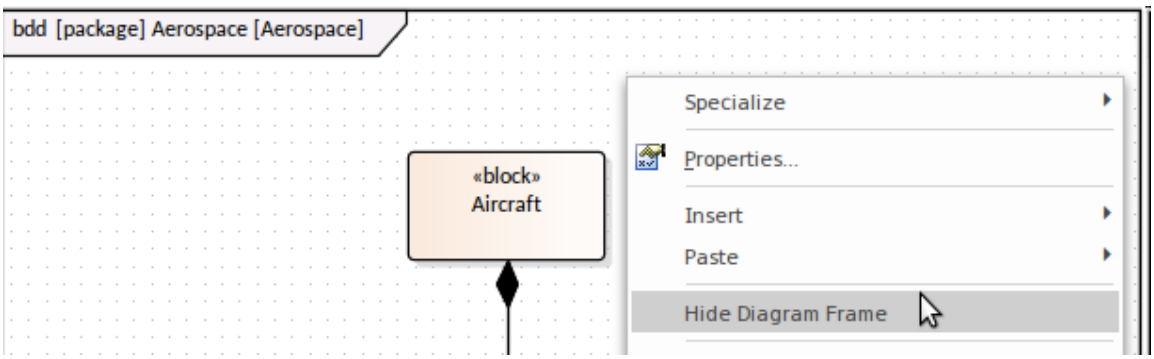
- 包含框架内模型元素的限定名称A标题区域，如果它不包含在与框架关联的默认命名空间中，则提供该名称；它具有以下形式：

diagramKind [modelElementType] modelElementName [diagramName]

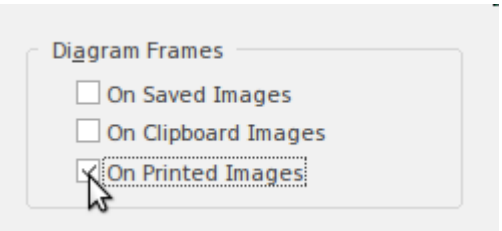
- 包含构成图表的视觉元素A内容区域



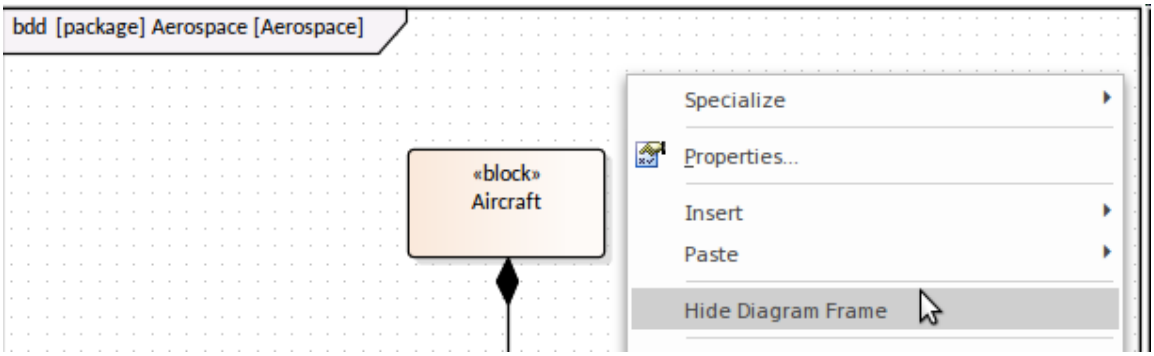
默认情况下， Enterprise Architect在图表窗口中显示框架，但根据 SysML 规范，可以抑制框架以创建不那么混乱的图表界面。（请记住， Enterprise Architect可以方便地在图表标题中显示框架标题信息）。



A在需要时重新打开框架，并且可以将发送到剪贴板或打印机的图表配置为显示框架，无论它们是否已隐藏在用户界面中。



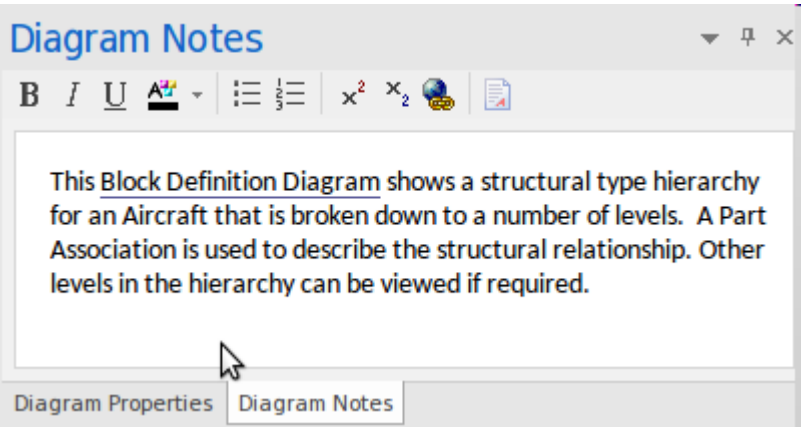
如果设置为不可选择，框架将自动调整大小以适应图表的边界，从其默认大小扩展但不会缩小。但是，它们可以进行选择和调整以适应建模者的偏好。



注记显示使用图表或更高版本的Enterprise Architect应用的图表将在使用早于 14.0 版本的Enterprise Architect打开时在图表上绘制父object 。

图表（注记）

除了 Header 中包含的元信息之外，图表还可以包含有助于新手理解图表目的和意图的描述。描述可以在图表的注记窗口中添加、查看和维护，



A创建图表来描述模型或系统的各个方面。虽然图表本身及其包含的元素和连接器讲述了一个故事，但通常需要使用描述性文本形式的一些额外信息来注释图表。例如，此文本可能：

- 描述图表的目的，
- 强调如何解释图表，
- 包含指向模型中其他信息的链接，
- 解释所使用的符号。

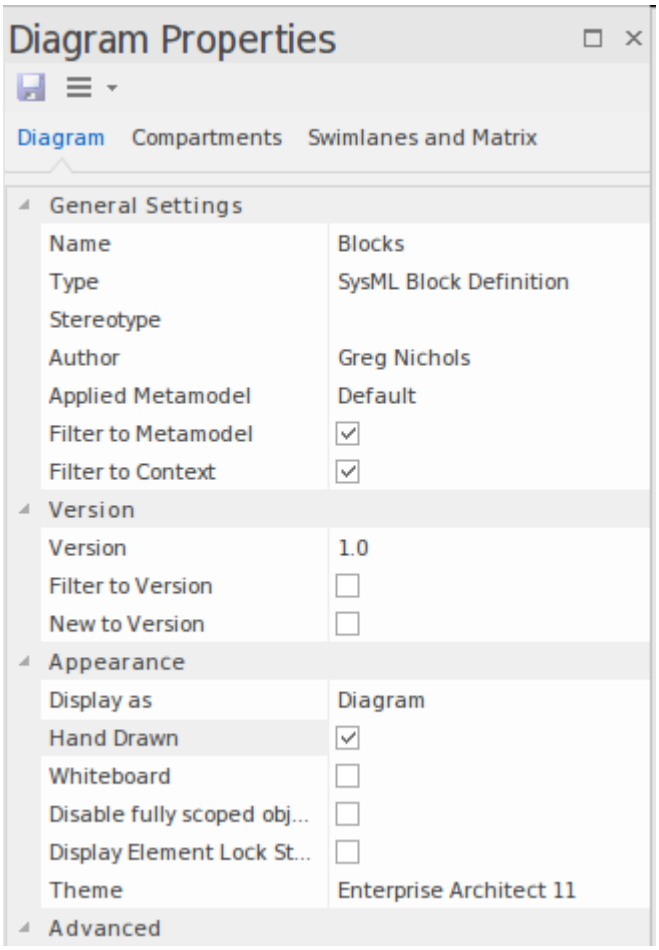
注记将生成为文档并通过WebEA界面可见。

图表属性

每个图表都有一系列在元级别描述图表的属性，包括以下项目：

- 创建图表的日期，
- 图表修改日期，
- 创建图表的模型作者，

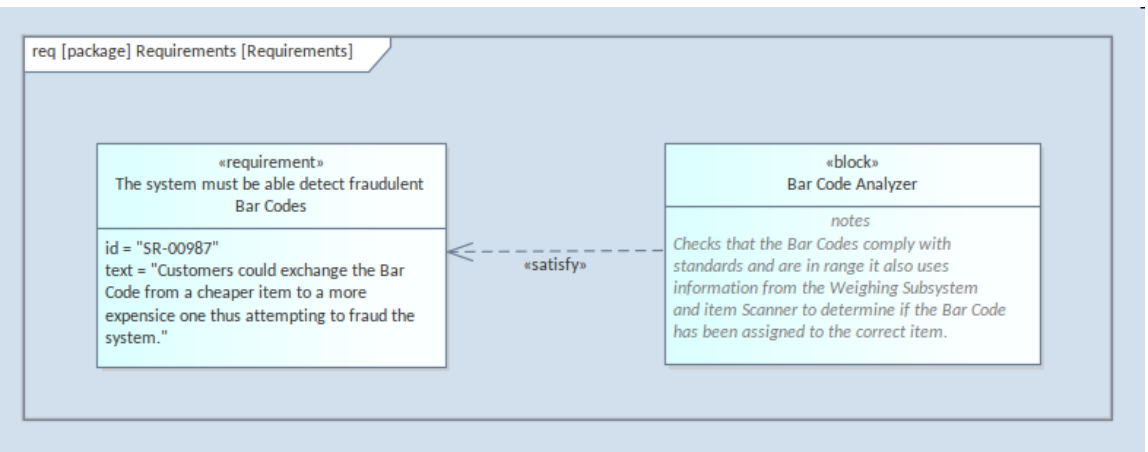
- 手绘和白板模式
- 视觉主题等等。



此外，还有一个选项卡专用于指定图中哪些隔间可见，另一个选项卡用于指定矩阵和泳道。

不断变化的主题和外观

Enterprise Architect提供了一种功能，允许您将选定的主题应用于设备上显示的所有图表。您可以使用它来创建特定样式的图表，它可以影响颜色、字体、渐变、线条粗细和背景图像（平铺）。这是一种有用的机制，可以使您的图表更具吸引力或统一它们的外观。

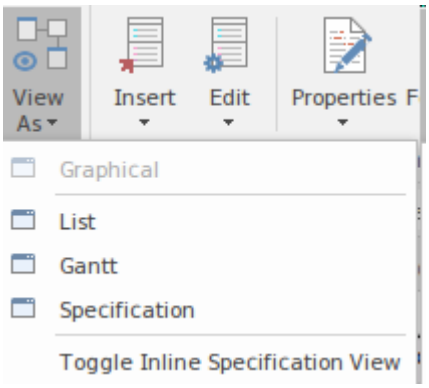


也可以为每个图表元素单独设置颜色和样式，或者作为默认值（元素出现在每个图表中）或仅为当前图表上的元素设置。

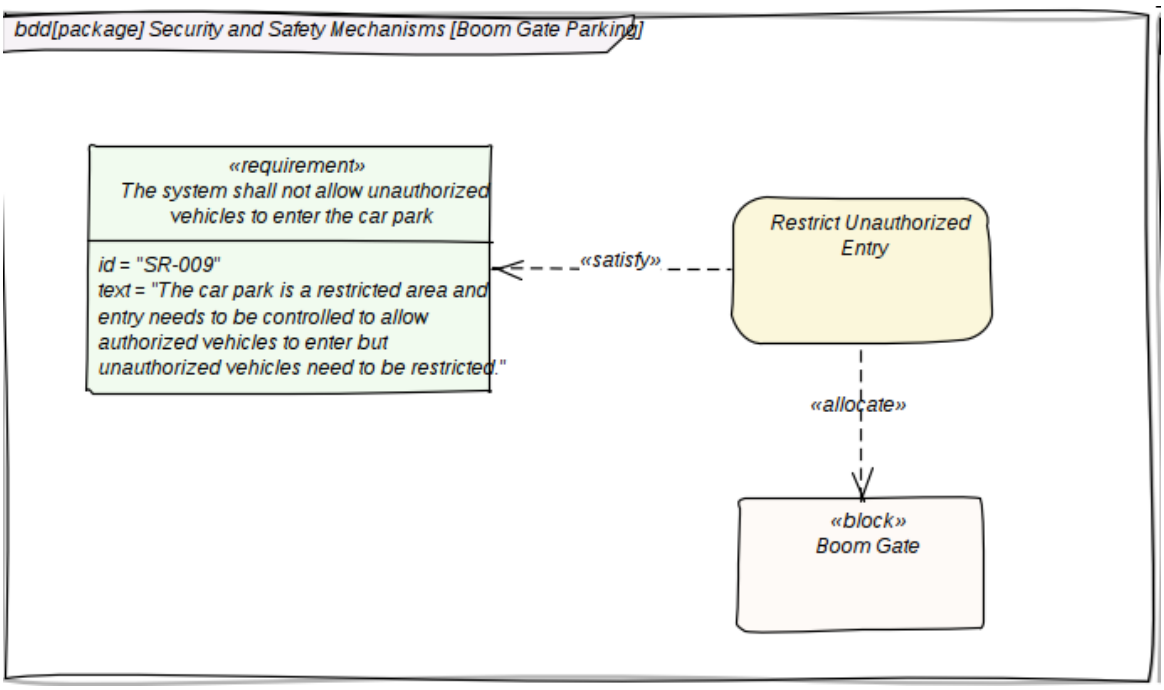
选择图表的可视图表

在符合 SysML 规范并提供建模工具中预期的所有图表功能的同时， Enterprise Architect为建模者提供了多种方式来以不同的方式查看图表，从而带来图表及其元素的丰富可视化。这些包括：

- 规范视图- 以熟悉的文字处理器或电子表格格式显示元素，允许更新元素和文本
- 列表视图- 显示列表中可以排序和分组的元素，并更新字段
- 内联规范视图-在类似于规范视图的叙述视图旁边显示视图
- 甘特视图- 在甘特视图中显示元素，显示资源分配和其他时间信息



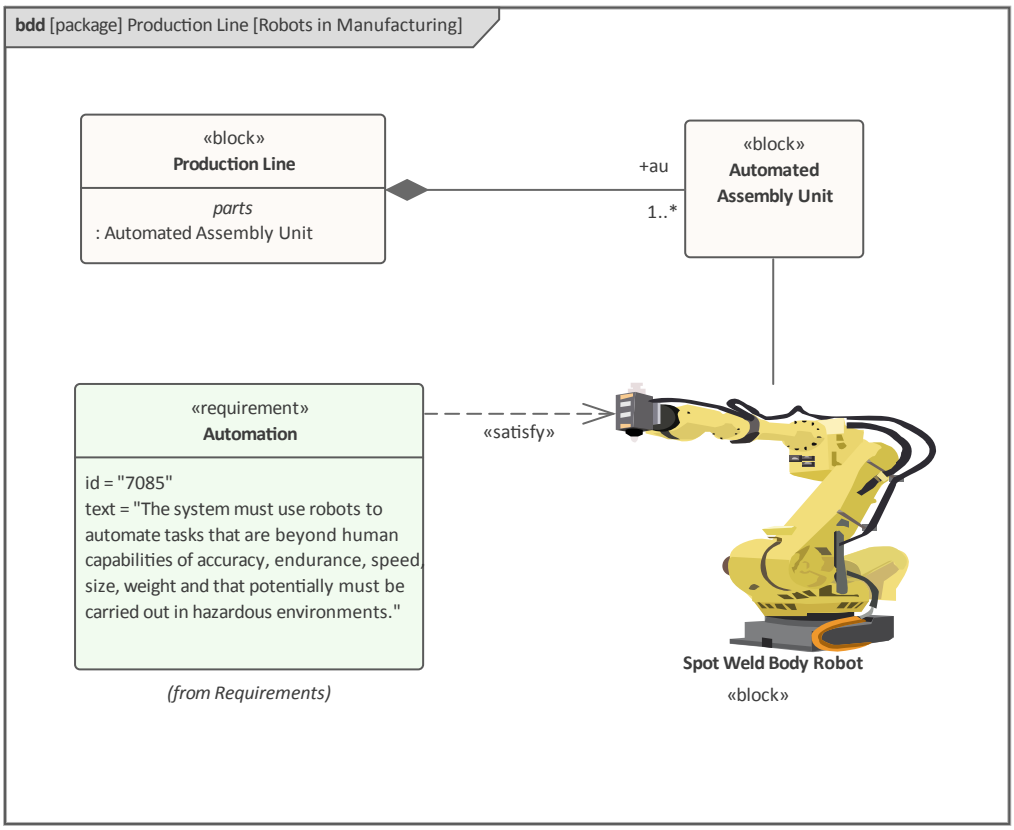
此外，可以以两种模式查看图表，改变图表的外观以适合演示，增加吸引力并减弱对底层建模语言严格性的聚焦。



用选择图像创造吸引力

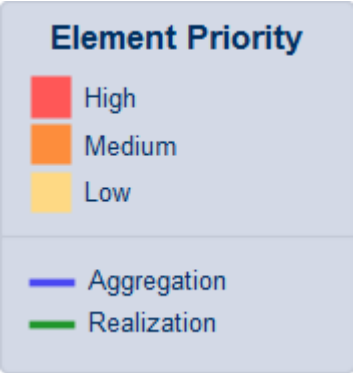
作为模型的一部分创建的图表旨在向观众传达想法，并且图表的意图通常通过使用图像更好地传达给非技术或

业务观众。 Enterprise Architect提供了一种机制，可以用各种格式的图像（包括基于矢量的图像）替换系统建模语言的普通（有时）不吸引人的图形表示法。



图表图例

图表Legends 提供了一种描述图表中使用的元素和连接器的方法。图例可以动态改变图表中元素和连接器的视觉方面，例如根据元素属性或标记值改变填充颜色、线条颜色和线条宽度。



可以创建任意数量的图例，并将它们应用于一个或多个图表。

图形的公共图表

图表作者是可视化模型内容的最重要的方法之一，并表示他们认为重要的内容的图表表达。它引人入胜的视觉吸引力和作为叙事讲述关于被建模系统的某些方面的故事的能力使该图成为最重要的视图之一。 SysML 中的每个图表都有共同的方面（或特征），包括：

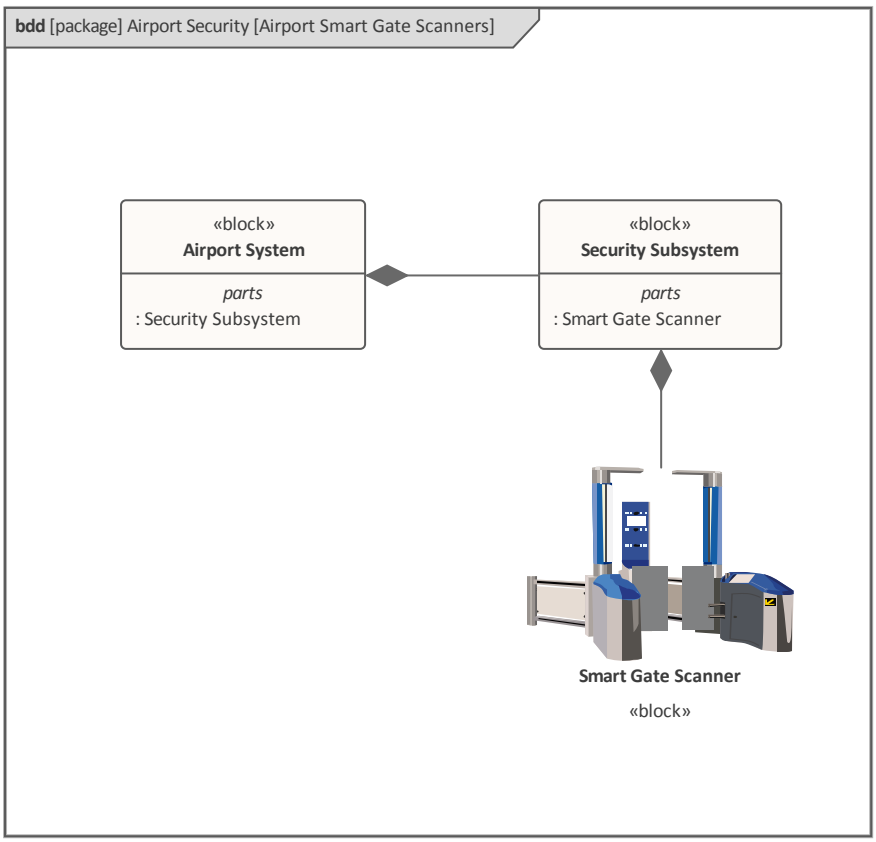
- 图表框架
- 图表标题
- 图表内容区（画布）
- 图表描述（注记）
- 图表属性Sheet
- A图例
- 和更多

还有很多其他功能可以帮助工程师处理图表，包括：

- 可追溯性窗口
- 窗口关系
- 平移和缩放
- 图表布局
- 图表过滤器
- 路线图
- 看板
- 缩放选项
- 外观、对齐和样式工具

图框

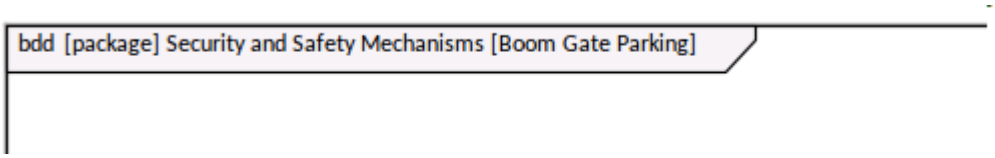
图框类似于一个窗帘，将构成图表的一部分的元素包裹起来。虽然在查看与书面文本一致的图表时更为重要，但一些建模者更喜欢在建模时让图表可见。


bdd [package] Airport Security [Airport Smart Gate Scanners]
«block»
Airport System
parts
: Security Subsystem
«block»
Security Subsystem
parts
: Smart Gate Scanner
Smart Gate Scanner
«block»

该框架在左上角包含一个标题，其中包含有关图表的有用信息。此语法描述标头的内容。

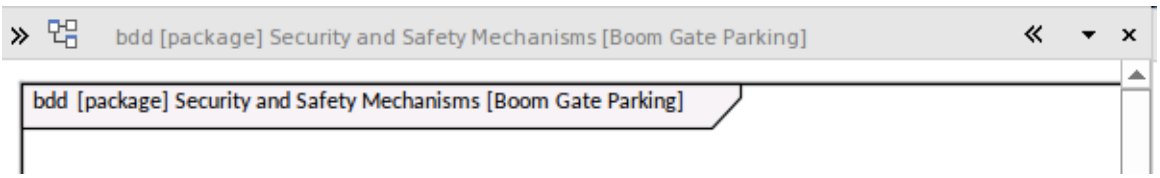
diagramKind [modelElementType] modelElementName [diagramName]

1. diagramKind - 是图表类型的代码，例如 bdd（块定义图），
2. *modelElementType* -元素的类型，作为图的命名空间，
3. *modelElementName* - 是命名空间元素的名称，
4. *diagramName* - 图表的名称，由用户提供。



图表标题栏

图表标题栏提供有关图表的有用信息以及用于处理所有打开图表的工具。即使框架不可见，标题也会显示图表详细信息。

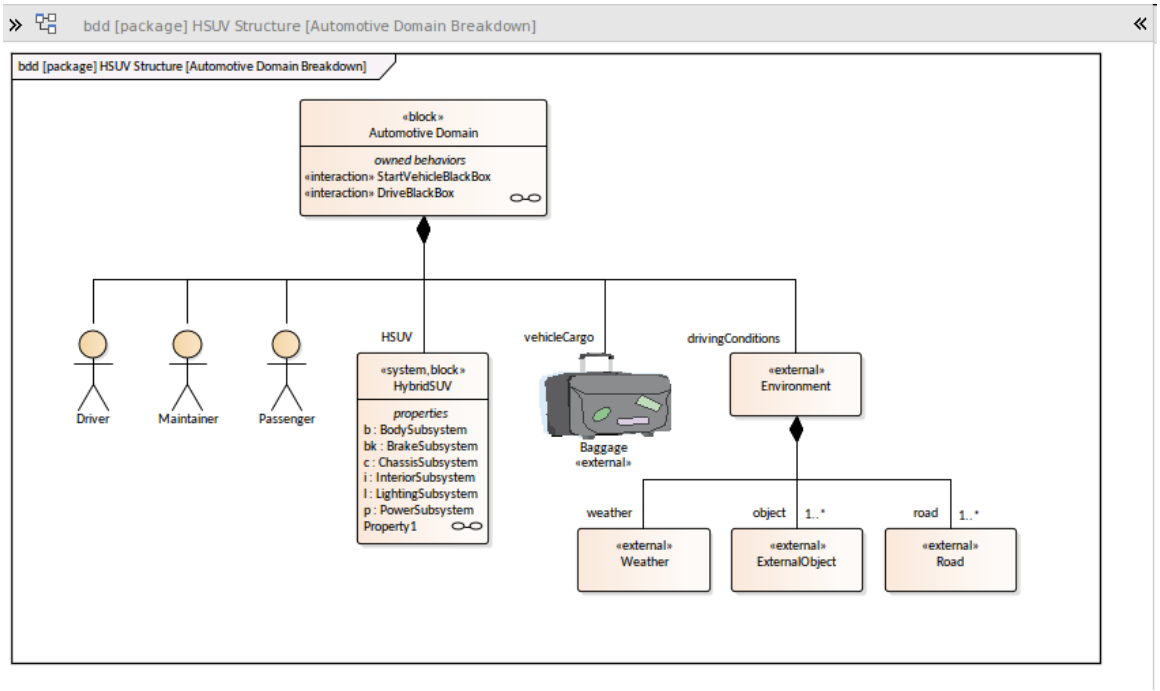


它还有许多其他有用的图标，用于控制显示的各个方面，包括：

- 工具箱*Chevron* - 隐藏或显示所有显示图表的工具箱，
- *Document Chevron* - 显示或隐藏图表的 Inline Specification 视图，
- 图表- 显示打开图表列表，指示未保存更改的图表，
- *Close*图表- 允许关闭图表。

图表内容

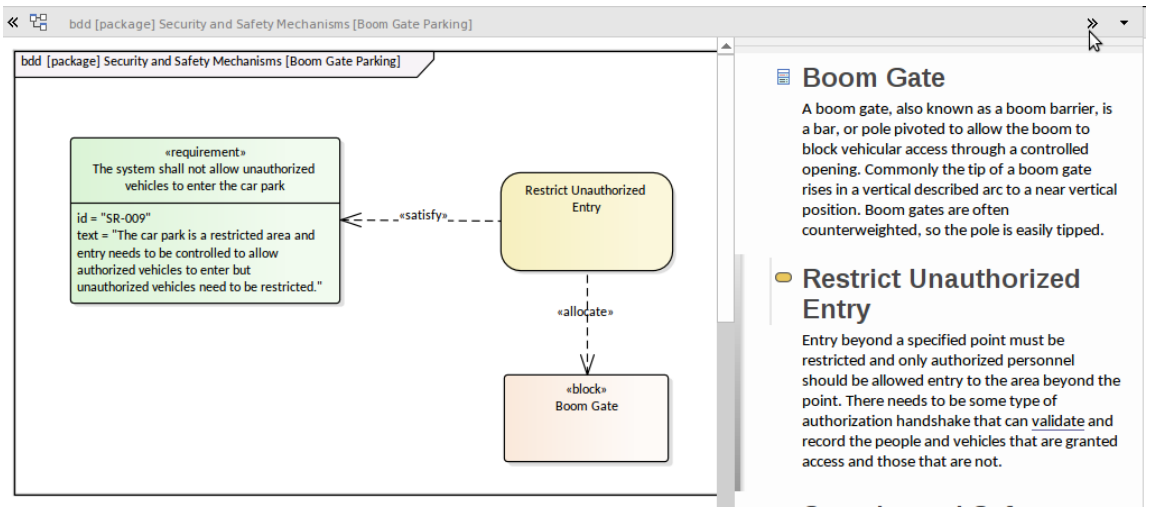
图表内容是您查看和处理图表的画布。这是一个高度灵活的工作区，有许多有用的工具可用于创建和可视化现有图表。可以添加图片并将其与标准几何 SysML 元素混合，从而提供富有表现力和引人入胜的图表，帮助向技术和非技术受众传达重要的工程概念。



Enterprise Architect使用多种可视化技术扩展了查看图表的方式。这些将为您提供处理图表内容的替代方法，并受到可能更熟悉使用电子表格、列表和文档中的元素的新手的欢迎。另请注意，在此图中，图像可以用作普通 SysML 形状的替代品。

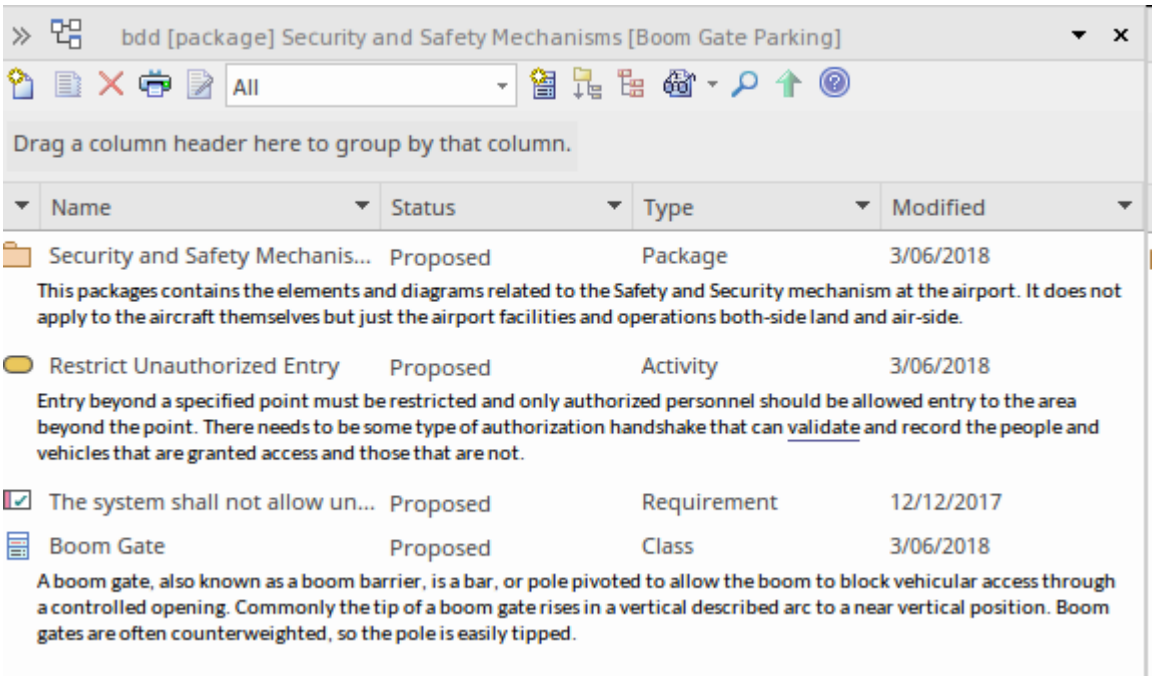
文档视图

这是一个方便的视图，可以在文档中以叙述形式显示图表上的元素。它也称为 "内联规范"面板，不是单独的视图，而是与所选图表相关的面板。该文档在图表的右侧很方便地打开，因此可以同时查看图表和文档。图表中出现的每个元素在文档中都有一个标题，元素的注记（**描述**）显示在标题下方的位置。在此视图中呈现的元素及其注记可以方便地编辑，同时查看文档视图和图表，并且两者保持同步。默认情况下，元素按字母顺序列出，但此排序顺序可以从上下文菜单更改为从上到下或从左到右。



列表视图

这是一种查看标记值中属性的属性方式，同时允许在熟悉的类似电子表格的注记中查看、更新和创建元素。具有离散值列表的属性可用，提供了一种跨多个元素查看这些属性的受欢迎的方式。



规格视图

规范管理器类似于内联文档查看器，但提供更多功能并在单独的可停靠窗口中打开。它是完美的工具，专为更熟悉使用电子表格或文档的工程师和其他利益相关者而设计。它本质上允许建模者或查看者将图表（或包）的内容可视化为文档或电子表格。文档视图类似于熟悉的文字处理器文档，可以在线编辑以创建新元素及其描述。可视化可以更改为类似于电子表格，其中属性显示在列中。

Item

1 REQ019 - Manage Inventory

The system MUST include a complete inventory management facility to store and track stock of books for the on-line bookstore.

1.1 REQ122 - Inventory Reports

Inventory reports are required that detail the available stock for each item including back orders. Future stock level reports should be able to predict the quantity of stock at a specified future date.

1.2 REQ023 - Store and Manage Books

A book storage and management facility will be required.

1.2.1 REQ022 - Order Books

A book order facility will be required to allow on-line ordering from major stockist's.

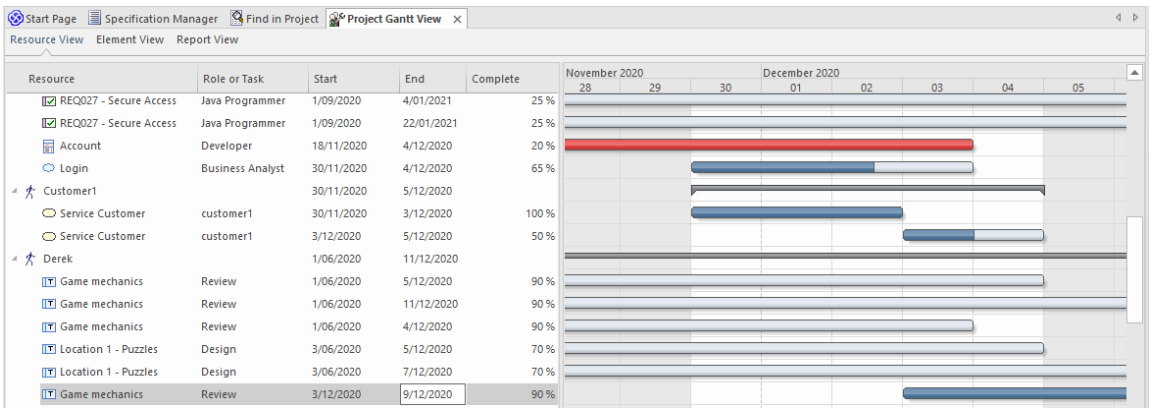
1.2.2 REQ021 - List Stock Levels

A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.

对于那些从基于文档的系统工程过程过渡的人来说，这是一个非常受欢迎的视图，也是工程师输入基于测试的信息（例如一组需求、活动、组件和其他元素）的最爱。

甘图表图

这提供了图表中元素的典型甘特图视图，其中资源分配已应用于元素，这可以被可视化，以便为工程或项目经理提供模型中的进度视图。



绘图工具

还有许多其他工具可用于显示存储库中的信息，这些工具将帮助工程师处理图表中的元素，包括但不限于：

可追溯性窗口

用于查看模型中的元素如何在元素及其关系图中连接。有关详细信息，请参阅The Traceability Window主题。

窗口关系

用于查看选定元素与模型中其他元素的关系；这些关系是不可见的，但可以在这个单独的窗口中方便地可视化。有关详细信息，请参阅The Relationships Window主题。

平移和缩放

用于通过使用代表视口的小矩形在大图上移动，并使用滑块控件放大。有关详细信息，请参阅Pan and Zoom主题。

图表布局

用于创建有吸引力的图表布局，使用选定的视觉布局模式，例如有向图和弹簧。有关详细信息，请参阅Layout Diagrams主题。

图表过滤器

用于从图表中的视图中过滤元素，方法是隐藏它们、将它们更改为灰度或淡化它们。您还可以反转行为并选择要包含的元素。有关详细信息，请参阅Diagram Filters主题。

路线图

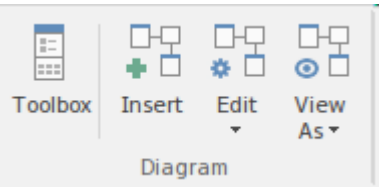
用于创建基于时间的元素表示，具有可配置为适合项目或建模者的应用时间刻度，以显示用于战略和开发目的的路线图。有关详细信息，请参阅Roadmap Diagram主题。

看板

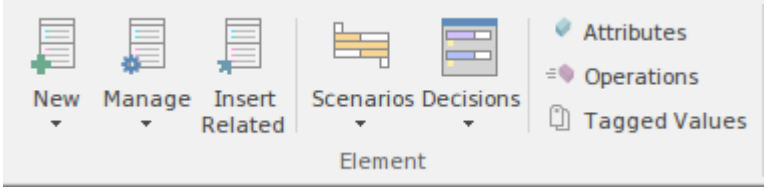
用于管理传统看板图中的项目，其中元素在表示它们在分阶段过程中的顺序的列之间移动；可以可视化处理项目的资源，为团队管理其模型或产品开发提供有用的方法。有关详细信息，请参阅Kanban Boards主题。

图表功能区和菜单

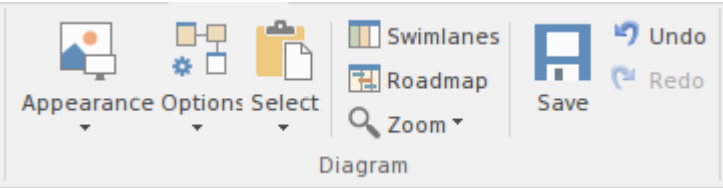
在使用图表时，有许多功能区很有用。如前所述，起点是设计功能区的 图表”面板，它允许您插入新图表以及编辑和更改现有图表的视图。



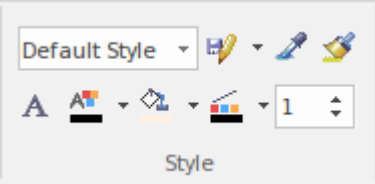
设计功能区上的 元素”面板在处理图表上的元素（或在浏览器窗口中选择）时也很有用，允许您插入新元素、编辑现有元素并管理元素、属性和特征。有关详细信息，请参阅Design Ribbon帮助主题。

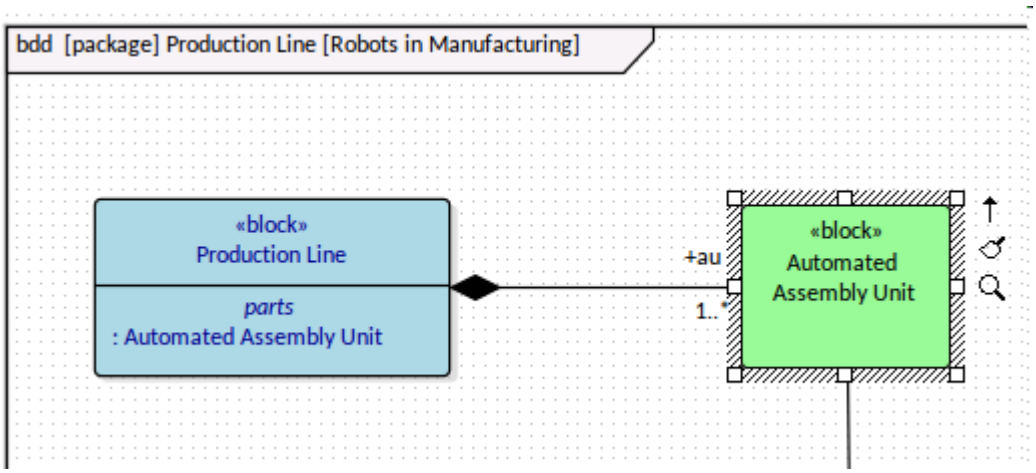


另一个重要的功能区是布局功能区，其中包含许多可用于处理图表的面板。这包括 图表"面板，其中包含设置主题和更改图表模式的选项，例如手绘。有关详细信息，请参阅Layout Ribbon主题。



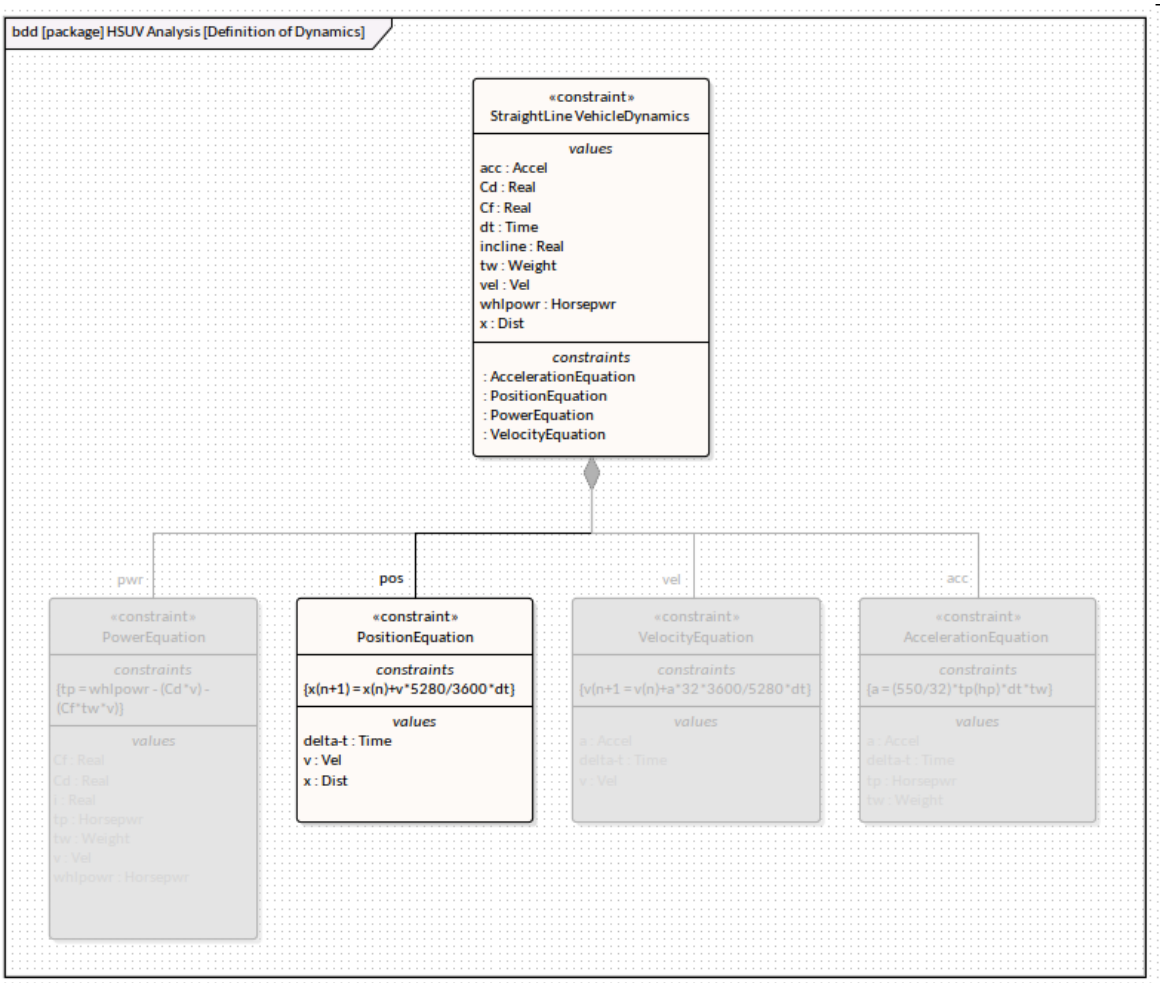
样 "面板允许将图表元素的视觉样 属性设置为组或单个元素。





对齐"面板提供了一组丰富的工具来对齐图表中的元素。这些非常有用，并提供了大多数绘图包中通常不提供的一系列对齐选项，例如，允许元素水平或垂直均匀分布。

"工具"面板提供了一系列用 处理图表的工具，允许过滤图表中的内容 - 例如，允许建模者仅显示在指定日期之后创建的具有指定状态的约束元素 - 平移和缩放以及布局选项。过滤器面板提供了一种快速且临时的方式来过滤图表中的内容，而无需创建图表过滤器。



A助手也可用于处理图表及其元素，控制诸如水平和垂直保持以限制元素对角线运动。

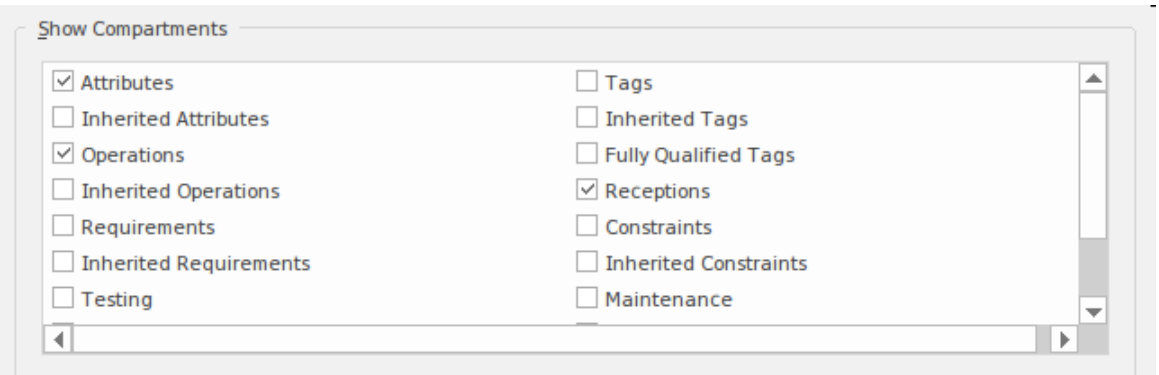
图表属性

Enterprise Architect中A图表有一组丰富的属性，**一些是描述性的** - 例如名称 - 而另一些是说明性的，指定图表应该如何显示以及哪些元素、隔间和其他特征应该是可见的，包括线条样式。

其中许多属性可以在图表级别和元素或连接器级别设置，从而允许图表的各个部分以不同的方式显示。

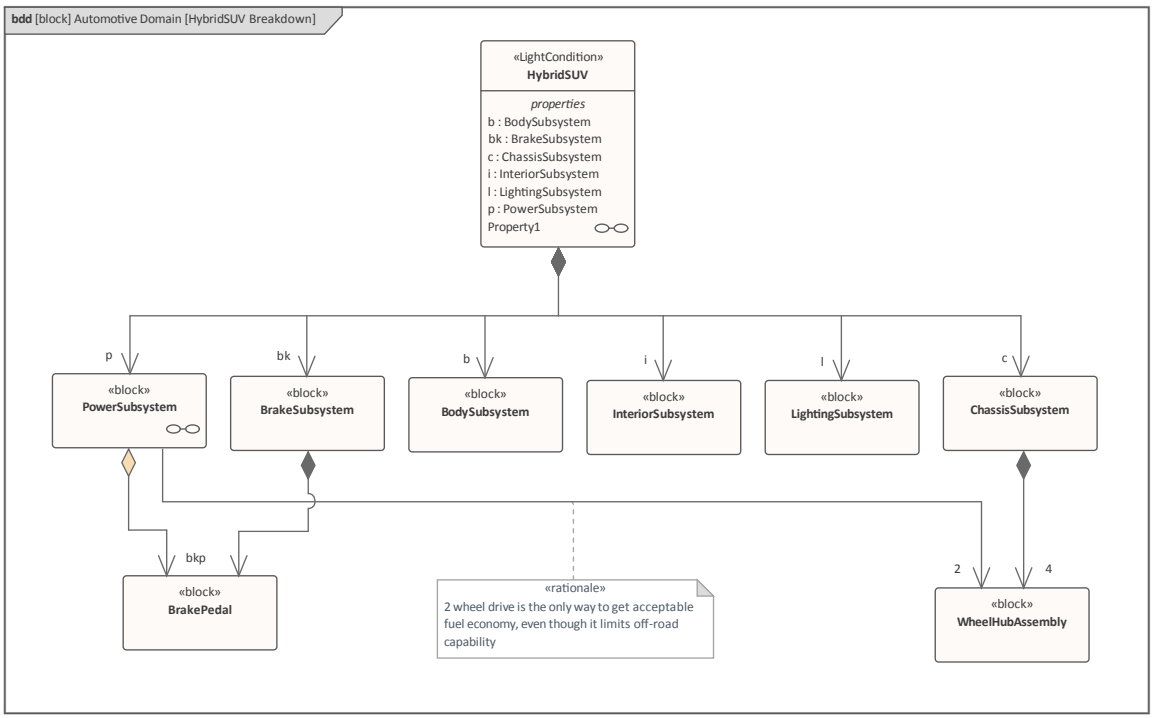
设置元素隔间可见性的能力对于工程图特别有用，因为 SysML 为各种项目提供了丰富的隔间集。这些隔间如果**不加以管理**，可能会使图表变得混乱并削弱读者理解图表含义的能力。

与其他属性一样，隔间可以设置在图表级别或元素级别，这允许建模者选择要为特定元素显示的隔间。



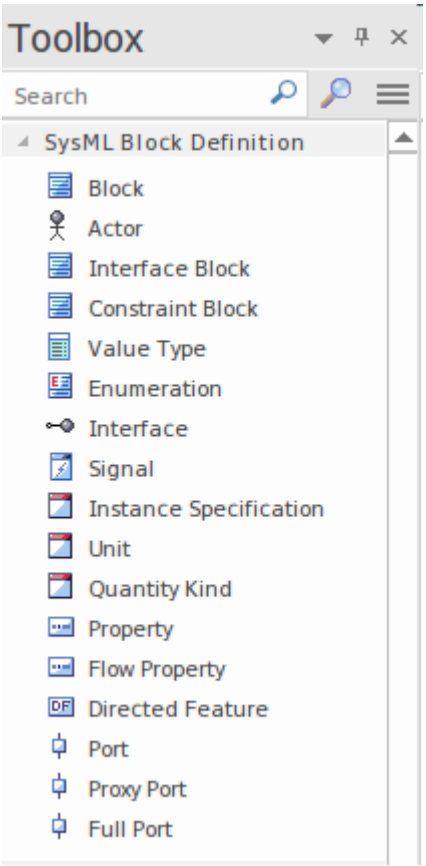
块定义图表

块定义图是 SysML 图中使用最广泛的；它用于模型操作块，它们与其他元素（包括其他块）的关系及其属性形式的特征，以及信号接收。块是系统描述的模块化单元，并提供一种将系统建模为模块化单元的图形或树的方法。其他元素，例如其它和属性，也可以出现在图表上并帮助描述正在建模的系统。有关更多信息，请参阅 Block Definition Diagrams (BDDs)帮助主题。

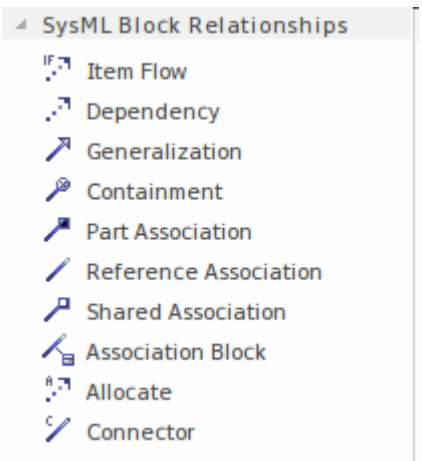


元素

可以出现在块定义图中的主要元素由图表工具箱为这种图表类型提供：



块图中可以出现的主要连接器如下图所示：



工具

A工具可用于结构建模和块定义图，包括：

- 图表- 允许用户从图表中过滤元素以实现更具体的聚焦
- 平移和缩放 - 允许建模者或查看者轻松地在大型图表中移动
- 电子表格 (CSV)导入和导出- 允许从模型中导入或导出电子表格中的内容
- 文档 - 允许从模型生成各种格式的正式或非正式文档

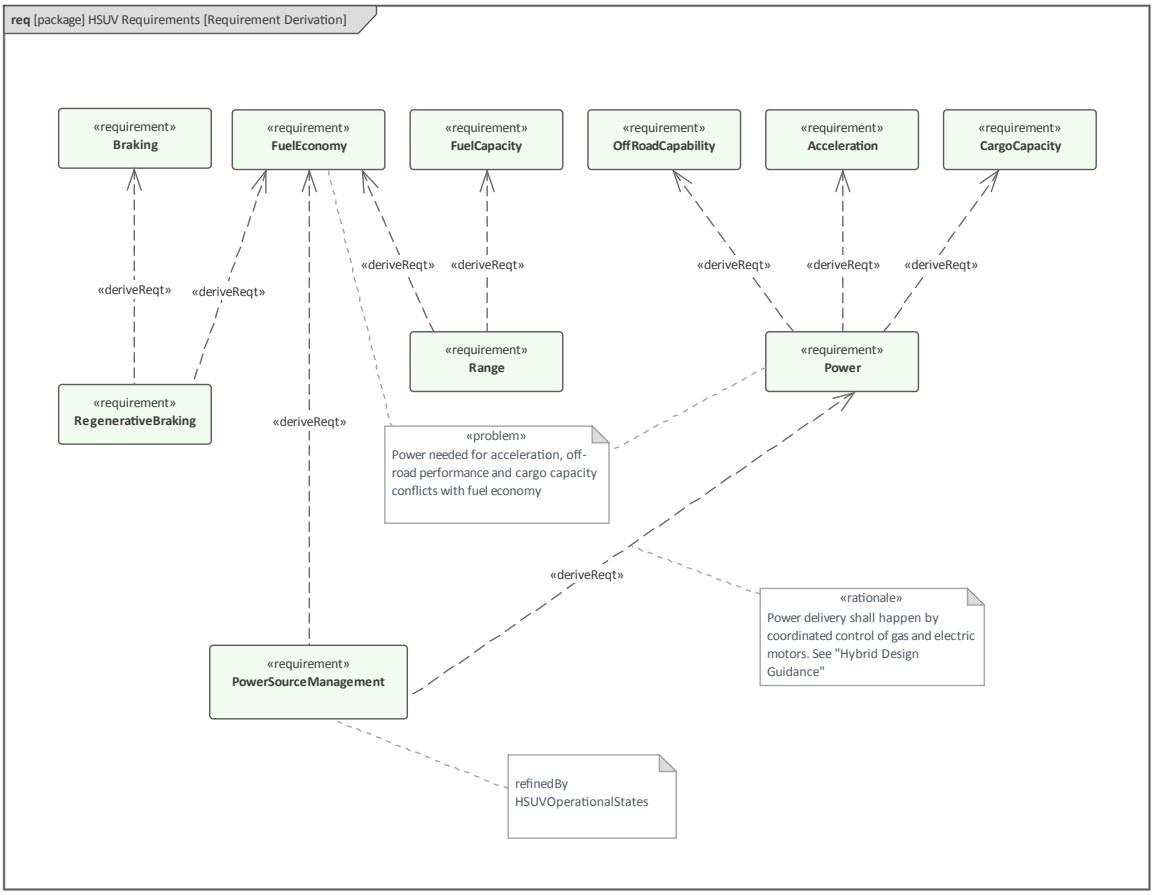
- 可追溯性——提供一个元素与其他模型元素的层次关系
- 责任窗口 - 提供元素重要责任的综合视图，包括约束、需求和场景
- 关系矩阵- 允许块（**或其他元素**）与其他元素（**例如需求和使用案例**）之间的连接在矩阵中可视化

用途

块定义图表是一个通用图表，可用于描述系统的结构方面。

需求图表

需求图用于创建和查看需求及其与其他元素的关系，包括其他需求。需求可以在任何级别指定，从战略性企业或业务需求到利益相关者需求，再到低级工程甚至软件和过渡要求。有关更多信息，请参阅SysML Requirements Modeling帮助主题。



可以通过多种不同方式查看此图中包含的元素，包括：

- 规范视图- 允许元素及其注记以文字处理器或电子表格格式显示在单独的可停靠窗口中
- 内联规范视图- 允许以叙述形式并排查看图表及其元素列表
- 列表视图- 允许在可以排序的列表中查看图表元素，并按属性分组元素
- 甘特视图- 允许在甘特图上表示图表元素，显示资源随时间的利用情况

元素

需求图中可以出现的主要元素有：

- 需求
- 测试用例

需求图中可以出现的主要连接器有：

- 遏制
- 跟踪
- 复制
- 派生
- 核实

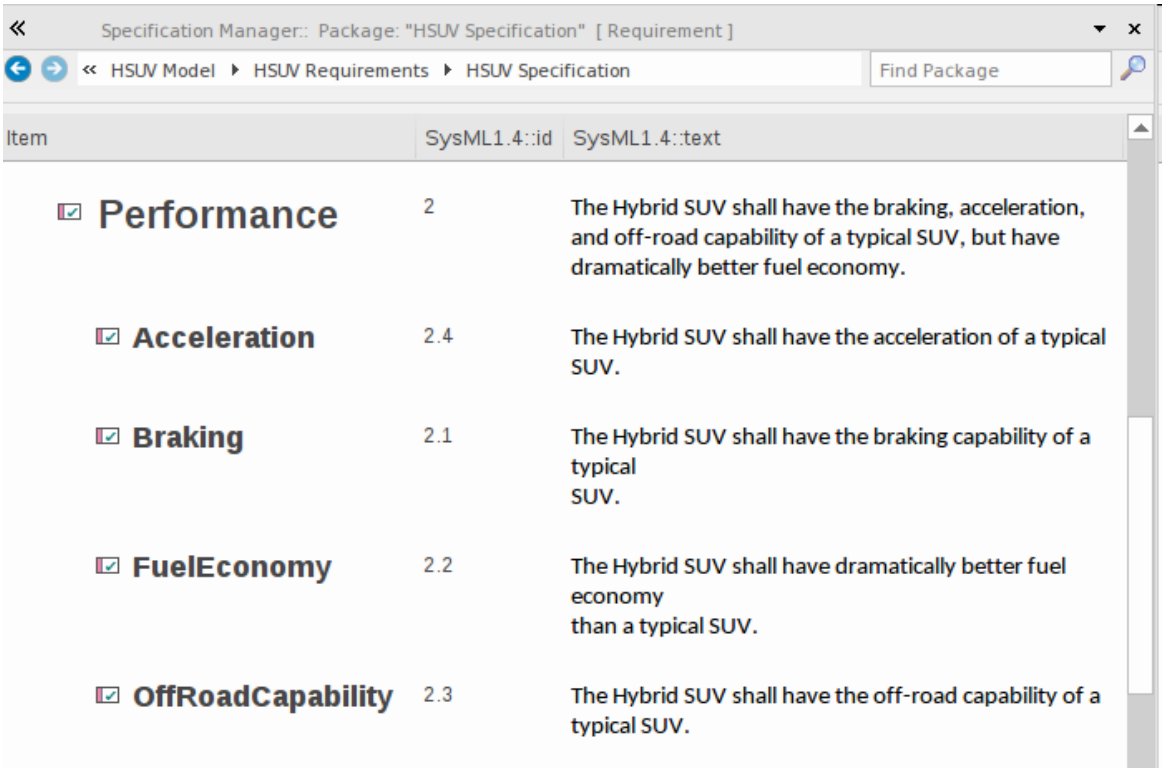
- 细化
- 满足

工具

A工具可用于需求建模，包括：

- 规范管理器-允许用户以电子表格或文字处理器类型的格式处理要求
- 电子表格 (CSV)导入和导出- 允许从模型中导入或导出电子表格中的内容
- 文档 - 允许从模型生成各种格式的正式或非正式文档
- 可追溯性——提供一个元素与其他模型元素的层次关系
- 责任窗口 - 提供元素重要责任的综合视图，包括约束、需求和场景
- 关系矩阵- 允许需求（或其他元素）与其他元素（如利益相关者）之间的联系需要在矩阵中可视化
- 思维导图 - 它提供了一种记录会议进度的方法，从而记录 - 例如 - 利益相关者的需求

这些需求工具中使用最广泛的可能是规范管理器，它将提供一种受欢迎且熟悉的方式来处理诸如需求或约束之类的文本规范。规范管理器可用于查看包或图表中包含的元素列表。

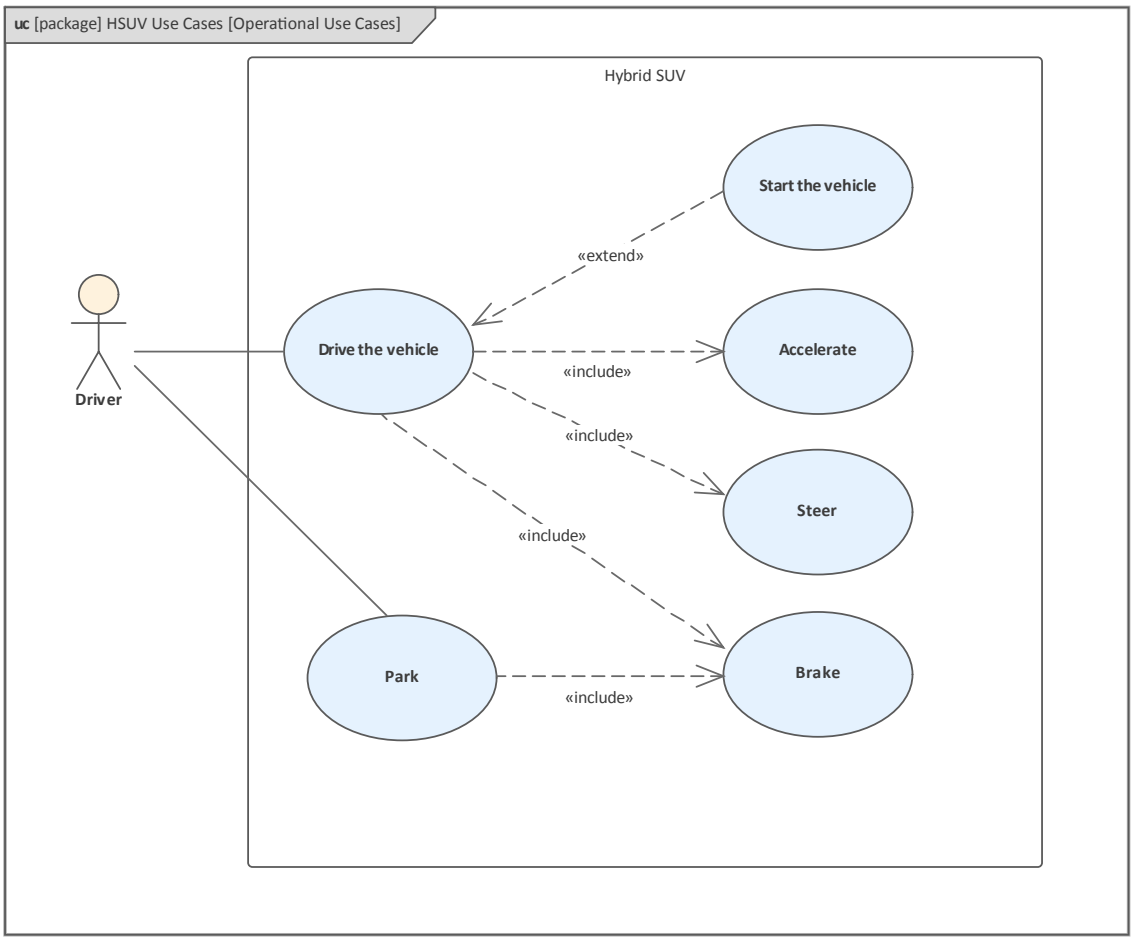


用途

需求图可用于使用包含关系显示需求层次结构，从而允许查看者查看需求的结构关系。然而，当需求与使用其他关系（包括其他要求）的其他元素一起在图表中查看时，这是最引人注目的。这方面的一个例子是需求与测试案例或需求与解决方案组件之间的关系。

用例图表

用例图用于定义和查看使用案例和从系统中派生价值的参与者。用例图描述了Actor和使用案例之间的关系，将用例封装在一个边界系统边界的地方；根据定义，Actor 位于此边界。虽然用例图可能看起来很简单，但它是一种有用的通信工具，它描述了外部角色从与系统交互中获得的价值或目标。用例可以详细描述、约束和任何场景，其中包含由每个参与者和系统交替执行的步骤集，以实现所需的目标。



元素

可以出现在用例图中的主要元素是：

- 边界
- 参与者
- 用例
- 设想

可以出现在用例图中的主要连接器是：

- 通讯路径
- 概括
- 包括
- 扩展

工具

除了用例图本身之外，还有多种工具可用于处理使用案例。这些包括：

- 场景窗口- 提供了一种详细说明每个场景的描述、约束和步骤的方法
- 文档生成器- 允许以各种格式创建公司、报告或临时文档，包括 docx、pdf 和 rtf
- 可追溯性——提供一个元素的关系视图到其他模型元素
- 责任窗口——提供元素重要责任的综合视图，包括约束、需求和情景
- 关系矩阵- 允许需求（或其他元素）与其他元素（如利益相关者）之间的联系需要在矩阵中可视化

用于处理使用案例的主要工具是场景窗口，它是一个综合的、专门为处理使用案例和场景而构建的功能。该工具可用于定义用例的详细信息及其场景和约束，这为定义使用案例的传统基于文本文档的方法提供了一种有效的替代方案。这确保了使用案例及其场景和约束的用例图和文字细节都包含在同一模型中并且可以追溯。

如果出于合同或流程原因需要文档格式的使用案例，则可以使用内置文档引擎从模型中自动生成用例报告。

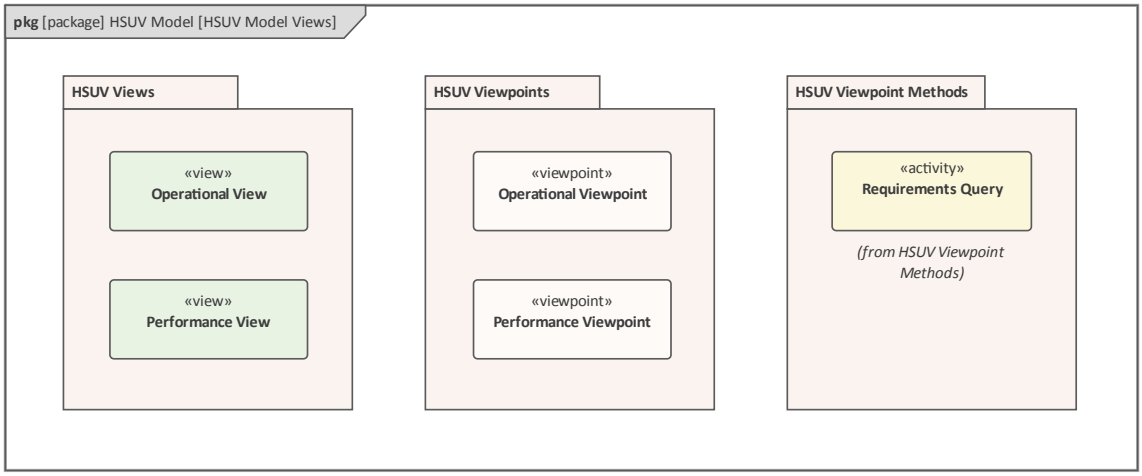
Scenario Builder

用途

用例图可用于定义用例的详细信息及其情景和约束。对于通常用于定义使用案例的传统基于文本文档的方法，这是一种受欢迎的替代方法。这确保了使用案例及其场景和约束的用例图和文字细节都包含在同一模型中并且可以追溯。如果出于合同或流程原因需要以文档格式呈现使用案例，则可以使用内置文档引擎从模型中自动生成用例报告。

包图

SysML包图用于定义或查看提供存储库基本组织的包。这些可以包括名称空间及其子包和其他不太正式定义的元素组。图表中出现的包也可以在浏览器窗口中查看，并且可以通过展开和折叠树来导航它们的层次结构。



包图中表示的主要元素是包本身，以及它包含的元素。包之间有许多重要的关系，包括表示一个包依赖于另一个包的依赖关系。包可以组织成许多不同类型的层次结构。

元素

包图中可以出现的主要元素有：

- **模型**
- 团队图书馆
- **包**
- 视图
- 视图
- 利益相关者

包图中可以出现的主要连接器有：

- **符合**
- 依赖
- 导入
- **遏制**
- 实现
- 细化
- **暴露**

工具

A工具可用于结构建模和块定义图，**包括**：

- 文档 - 允许从模型生成各种格式的正式或非正式文档
- 可追溯性——提供一个元素与其他模型元素的层次关系

- 责任窗口 - 提供元素重要责任的综合视图，包括约束、需求和场景
- 关系矩阵- 允许包（**或其他元素**）与其他元素（**例如需求和使用案例**）之间的联系在矩阵中可视化

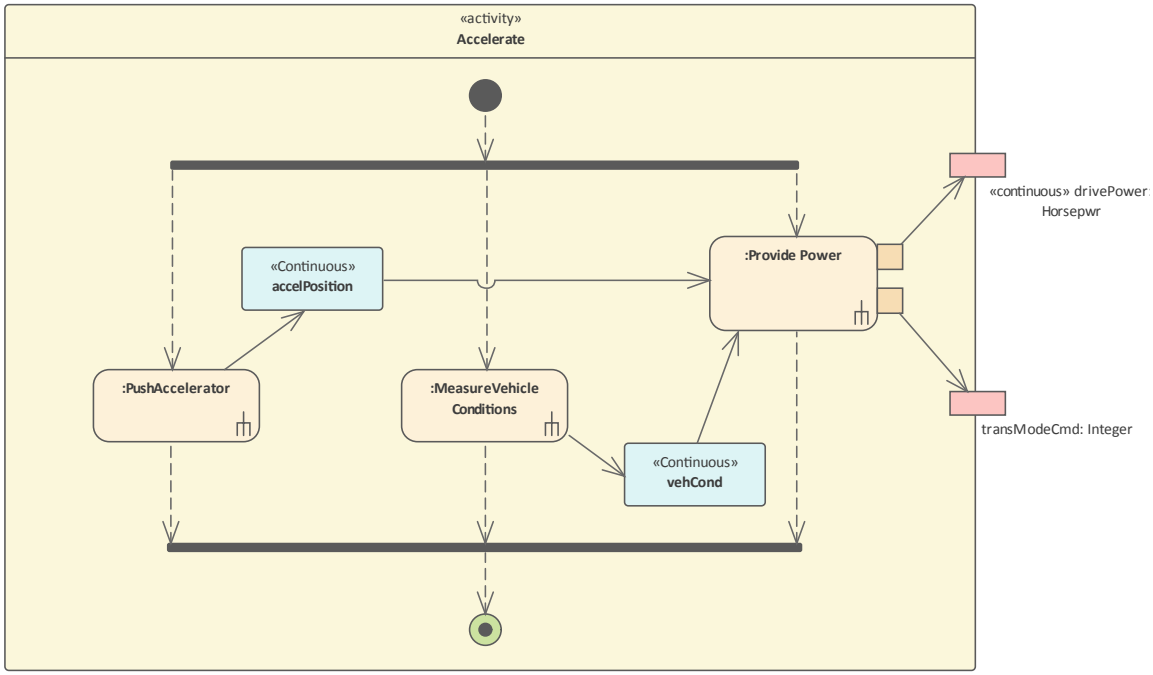
用途

包图可以用来描述包和它们所包含的元素之间的关系。虽然结构信息在浏览器窗口中可见，但在包自身之间以及包与元素之间可能存在一系列关系，但在浏览器窗口中无法可视化。包图也可以包含在文档中并有助于通过以类似于在出版物中提供内容表的方式向观众提供架构或设计部分的概述来引导他们。

活动图表

活动图是最重要的行为图，**可用于模型基于流**（离散或连续）的行为，其中通过序列对项目执行工作的一系列动作将输入转换为输出。它们类似于常见的流程图，但具有更复杂的语义，并且还允许活动和行动与块、需求和使用案例等元素相关。

出现在活动图上的行动可以包含输入或输出引脚，这些引脚表示输入被馈入动作并发出输出的交互点。



用途

活动图可用于模型基于流的行为，并且类似于在设计 SysML 规范之前已广泛使用的广泛可用的流图表或功能流图。它们通常用于显示系统各部分的行为方式，包括项目和信号的输入和输出。

元素

可以出现在活动图中的主要元素是：

- 活动
- 行动（各种）
- 行动销
- 分区
- 物件节点
- Central Buffer节点
- 数据存储
- 决策

- 合并
- 最初的
- 终点

可以出现在活动图中的主要连接器是：

- 控件
- 物件流
- 中断流
- 依赖

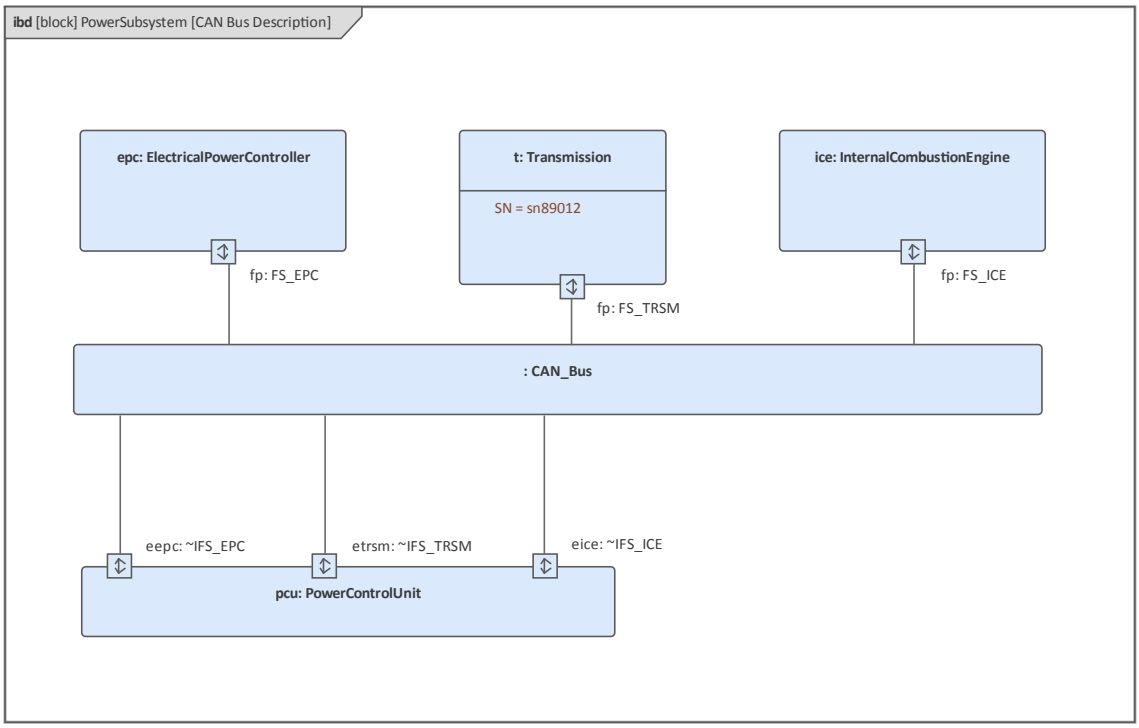
工具

A工具可用于行为建模和活动图，包括：

- 图表- 允许用户从图表中过滤元素以实现更具体的聚焦
- 平移和缩放 - 允许建模者或查看者轻松地在大型图表中移动
- 电子表格 (CSV)导入和导出- 允许从模型中导入或导出电子表格中的内容
- 文档 - 允许从模型生成各种格式的正式或非正式文档
- 可追溯性——提供一个元素与其他模型元素的层次关系
- 责任窗口 - 提供元素重要责任的综合视图，包括约束、需求和场景
- 关系矩阵——允许活动（或其他元素）与其他元素（如块、需求和使用案例）之间的连接在矩阵中可视化

内部块图表

内部块图提供了一种查看块组成的方法，该方法使用使用端口和连接器连接在一起的部件属性。该图对于显示一个块（由图框表示）的组成以及组成块的各个部分之间的输入和输出流很有用，当需要时，可以在连接器上指示流向。



元素

可以出现在块定义图中的主要元素有：

- 属性
- 连接器属性
- 分布式属性
- 流属性
- 有界参考
- 结束路径多重性
- 信号
- 端口

可以出现在块定义图中的主要连接器是：

- 项目流
- 连接器
- 捆绑连接器
- 依赖

工具

A工具可用于结构建模和内部块图，包括：

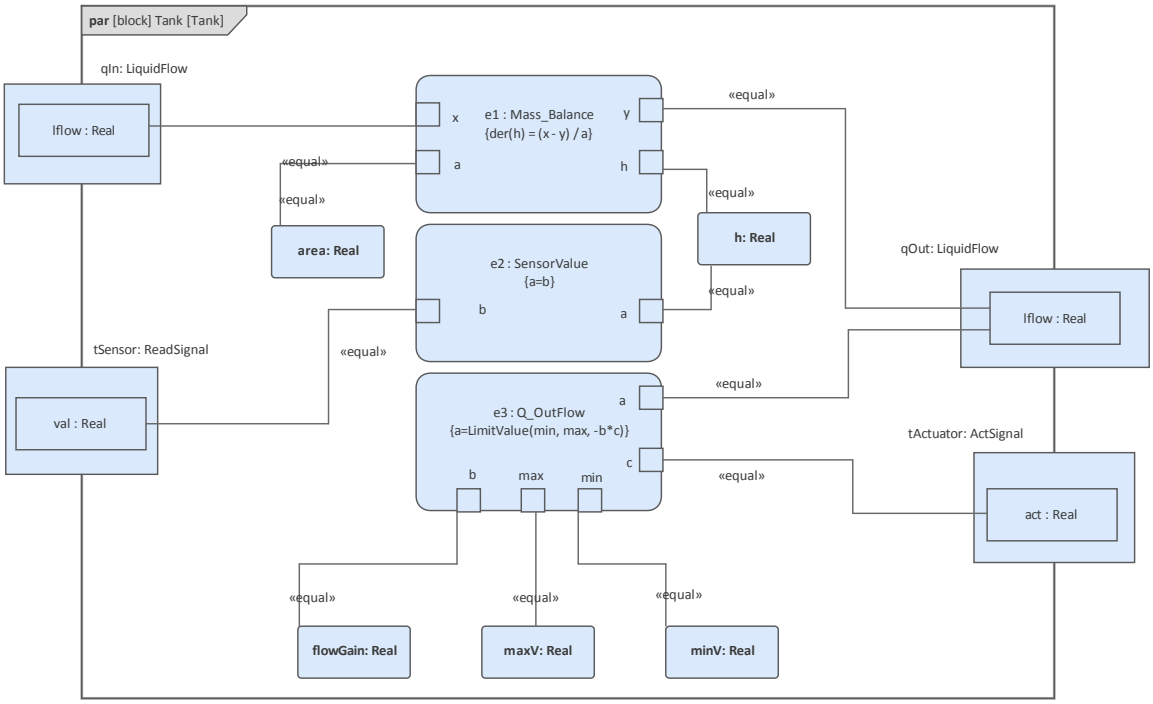
- 图表- 允许用户从图表中过滤元素以实现更具体的聚焦
- 平移和缩放 - 允许建模者或查看者轻松地在大型图表中移动
- 电子表格 (CSV)导入和导出- 允许从模型中导入或导出电子表格中的内容
- 文档 - 允许从模型生成各种格式的正式或非正式文档
- 可追溯性——提供一个元素与其他模型元素的层次关系
- 责任窗口——提供元素重要责任的综合视图，包括约束、需求和场景
- 关系矩阵- 允许块（或其他元素）和其他元素（例如需求和使用案例）之间的连接在矩阵中可视化

用途

内部块图用于模型块的内部结构，包括其组成部分以及这些关系之间的关系。

参数图表

图表参数图是一种内部块图表（有一些限制），用于模型带参数的方程。它们是可用于描述方程及其参数的重要工具，并且在进行权衡分析和评估设计备选方案时非常重要，因为它们可以组合成方程组并与有效性度量 MOE 相关。



参数图描述了约束块的使用，并提供了一种将工程分析（例如性能和可靠性以及其他感兴趣的因素）与其他 SysML 模型和图表集成的机制。

参数图定义了约束块用于约束另一个块的属性的方式。约束的使用被称为将约束的参数（例如 F=m*a）（例如 F、m 和 a）绑定到块的特定属性（例如质量和加速度），这些属性提供以下值参数。

元素

可以出现在参数图中的主要元素是：

- 约束属性
- 属性
- 目标函数
- 有效性测量

可以出现在参数图中的主要连接器是：

- 连接器
- 捆绑连接器
- 项目流
- 依赖

工具

A工具可用于结构建模和内部块图，包括：

- Modelica集成——它提供了一种模拟机制，
- 图表- 允许用户从图表中过滤元素以实现更具体的聚焦，
- 平移和缩放 - 允许建模者或查看者轻松地在大型图表中移动，
- 电子表格 (CSV)导入和导出- 允许从模型中导入或导出电子表格中的内容，
- 文档 - 允许从模型生成各种格式的正式或非正式文档，
- 可追溯性——提供一个元素与其他模型元素的层次关系，

用途

参数图可用于通过指定表示数学表达式（例如 {F=m*a} 和 {a=dv/dt}）的约束网络来显示系统的物理属性如何受到约束。

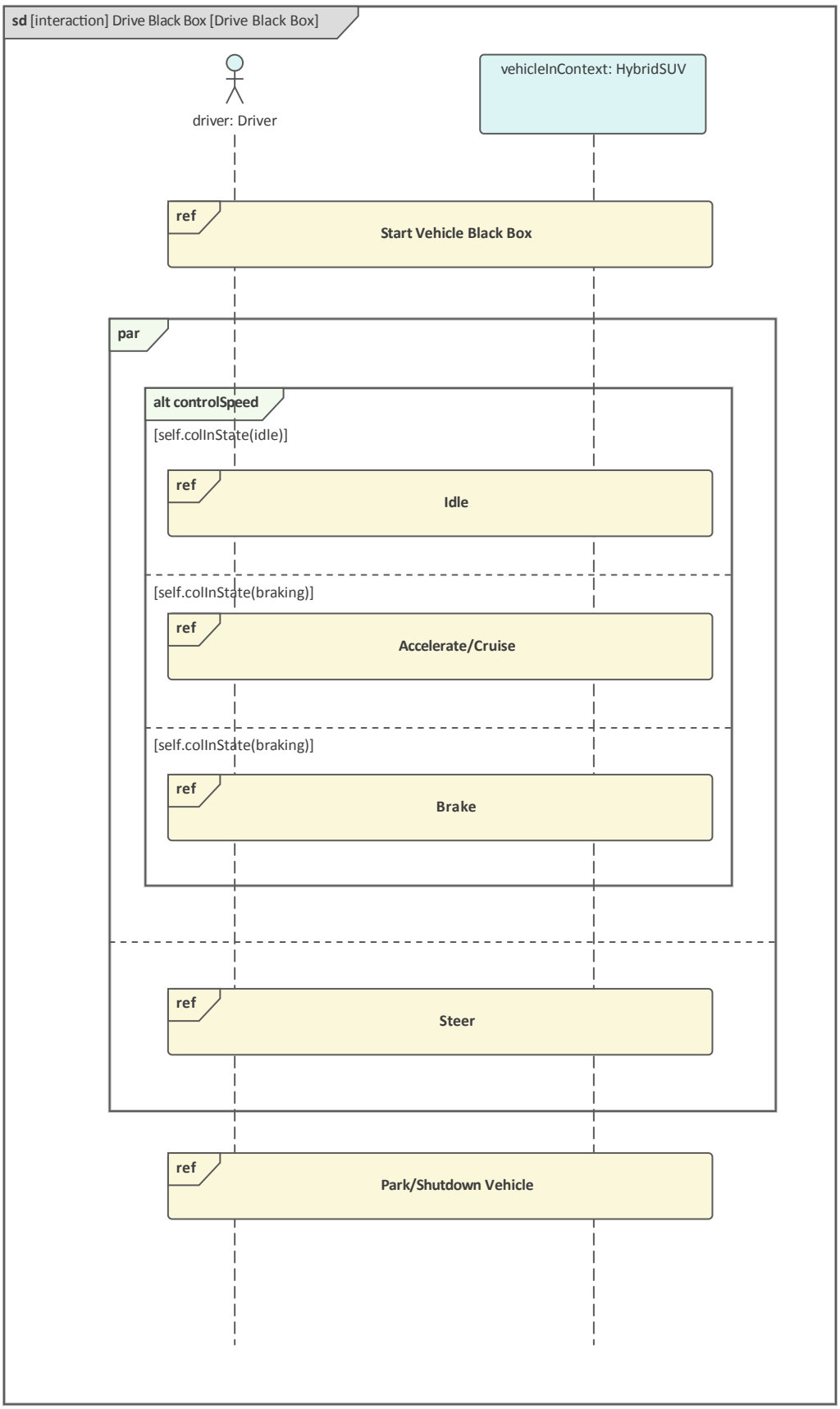
它们还可以用于权衡分析，其中约束块可以定义一个目标函数，用于在替代解决方案之间进行比较。

可以对关键性能参数及其与其他参数的关系进行建模，然后在整个系统生命周期中对其进行跟踪。

序列图表

序列图是A显示时间有序交互对象之间的交互图。该图有两个轴；纵轴代表时间，横轴代表参与交互的对象，通常以最能说明交互的方式排序。这些图起源于软件交互的建模，但它们可以与系统工程一起使用，以规定元素（例如块）应该如何交互，或者在实践中描述它们如何交互。

此序列图显示了驾驶员和车辆之间的消息流交互和序列。该图表达了 驾驶车辆”用例的必要交互。该交互由“AutomotiveDomain”块拥有。


sd [interaction] Drive Black Box [Drive Black Box]
driver: Driver
vehicleInContext: HybridSUV
ref
Start Vehicle Black Box
par
alt controlSpeed
[self.colInState(idle)]
ref
Idle
[self.colInState(braking)]
ref
Accelerate/Cruise
[self.colInState(braking)]
ref
Brake
ref
Steer
ref
Park/Shutdown Vehicle

元素

可以出现在参数图中的主要元素是：

- 序列
- 分段
- 端点
- 图表门
- 状态/延续

可以出现在参数图中的主要连接器是：

- 信息
- 自我信息
- 递归
- 依赖

工具

A工具可用于行为建模和活动图，包括：

- 图表- 允许用户从图表中过滤元素以实现更具体的聚焦，
- 平移和缩放 - 允许建模者或查看者轻松地在大型图表中移动，
- 电子表格 (CSV)导入和导出- 允许从模型中导入或导出电子表格中的内容，
- 文档 - 允许从模型生成各种格式的正式或非正式文档，
- 可追溯性——提供一个元素与其他模型元素的层次关系，
- 责任窗口 - 提供元素重要责任的综合视图，包括约束、需求和场景

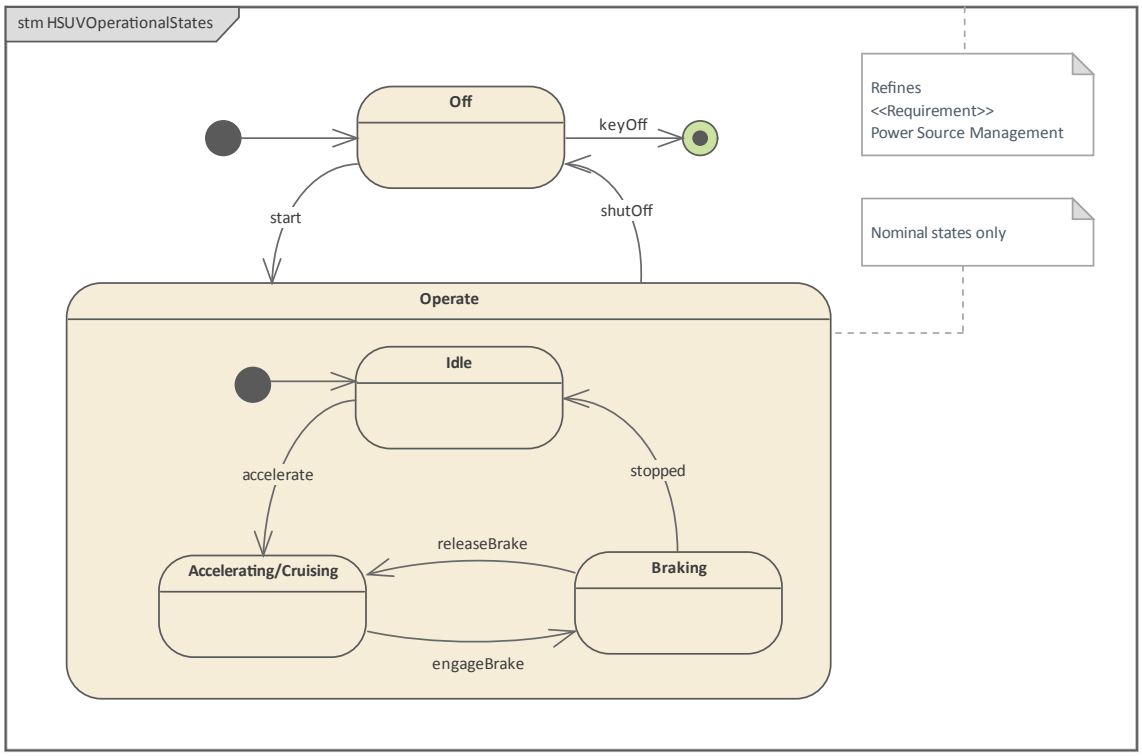
关系矩阵——允许活动（或其他元素）与其他元素（如块、需求和使用案例）之间的连接在矩阵中可视化。

用途

我们正在对系统建模这一事实意味着它具有一定程度的复杂性，如果不使用工具就无法管理。序列图可用于表示系统场景，显示系统的各个部分如何相互作用以实现某些特定的结果.据说消息在代表object生命周期的生命线之间交换，消息代表对象发送和接收的操作或信号。

状态机图表

状态机A是显示系统元素（如块）生命周期信息的有效方式。它可以用来描述一个实体的生命周期或周期中发生的重要条件（状态）。**通常**，只有在其生命周期中具有重要阶段的实体才使用状态机图进行建模。实体被称为从状态机指定的一个状态转换到另一个状态。可以描述允许状态转换触发器的事件，并且可以定义限制状态变化的状态。每个状态都可以定义在进入、期间和退出状态时发生的行为。



元素

可以出现在参数图中的主要元素是：

- 状态
- 状态机
- **最初的**
- 终点
- 选择
- 历史
- 分叉和汇合

可以出现在参数图中的主要连接器是：

- 转移
- 依赖

工具

除了状态机图本身之外， A多种工具可用于处理状态机图。这些包括：

- 状态库表表器 - 它允许状态机图表以表格的形式可视化，对于某些分析师来说，这种表格比图表更容易理

解；它包含与图表相同的信息，可以通过多种不同方式查看

- 动态仿真-允许状态机可视化，显示实体如何从一个状态到另一个
- 可执行状态机- 以及利用仿真引擎并允许状态机可视化，提供完成语言特定的实现，可以形成多个平台上的多个软件产品的行为 "引擎"

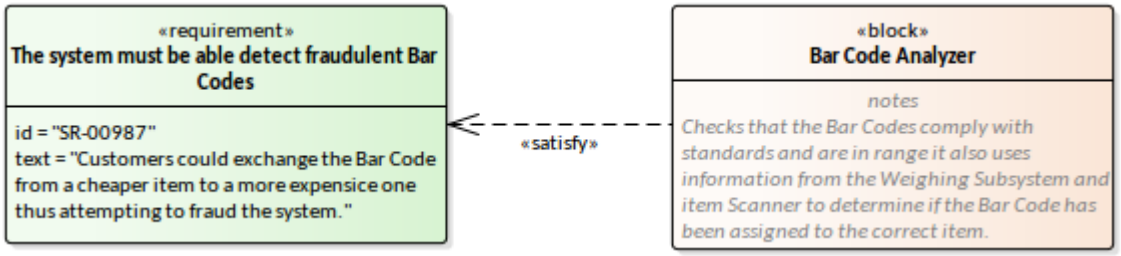
用途

复杂系统通常由具有复杂行为并且可能具有难以理解的生命周期的块等实体组成。

状态机可以用来描述一个实体的生命周期或周期中发生的重要条件（States）。通常，只有在其生命周期中具有重要阶段的实体才使用状态机图进行建模。这些图表提供了对实体从状态到状态转换方式的状态，忽略了对分析不重要的条件。

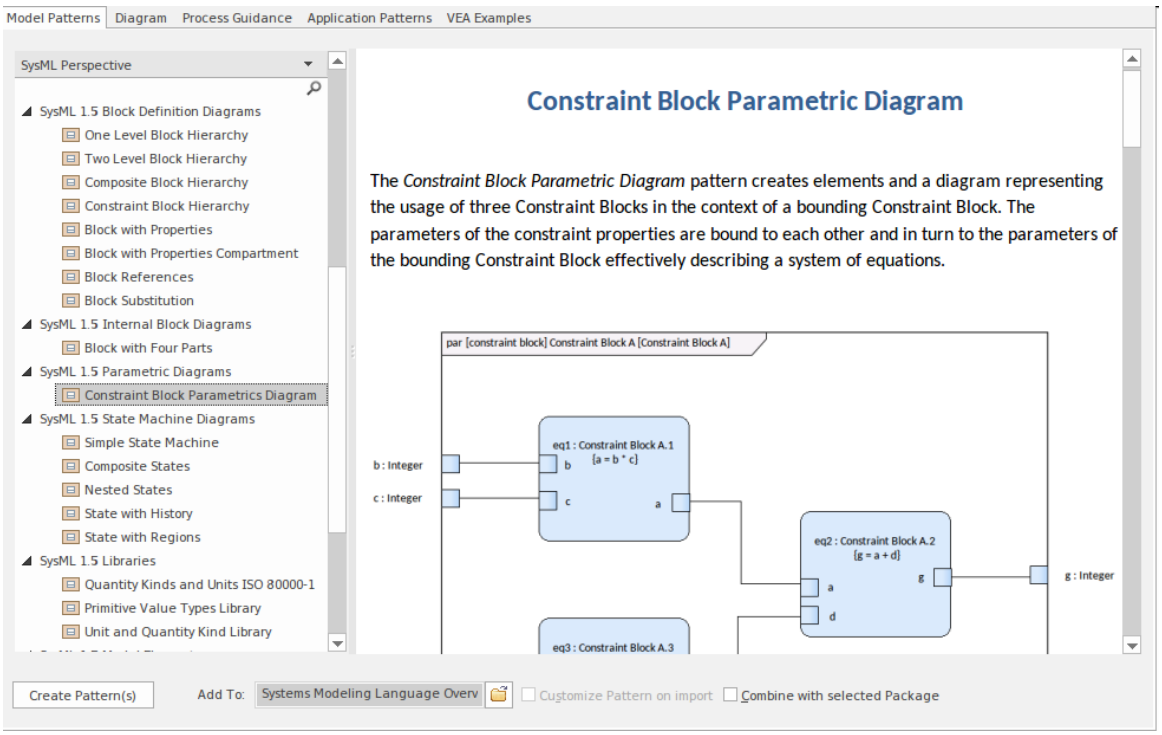
系统建模语言概述

基于模型的系统预示着通信和实时模型协作的新时代。它带来了作为机器而不是装满文件的柜子的模型的概念。 A台机器可以完成诸如验证需求、生成复杂数学和物理方程的参数模拟、使可执行机器状态机、模拟业务和决策逻辑、响应审查而发展以及创建文档等工作的机器，仅举几例。这些好处是通过Enterprise Architect的力量实现的，并且因为使用标准和共享语言来创建模型 -系统建模语言，通常缩写为*SysML*。

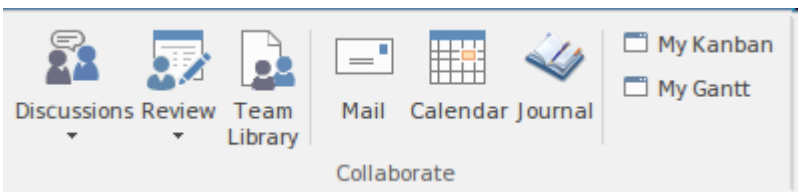


SysML 允许人类和机器理解模型——人类增加独创性、工程和设计，机器执行繁琐且容易出错的任务，例如验证、执行繁重的工作（例如生成参数模拟和执行假设分析），并执行更普通的任务，例如搜索和报告生成。

语言的习得不是免费发生的事情，而是可以在没有痛苦或挫折的情况下发生的事情，也没有一些怀疑论者所说的 “语言天赋”。 Enterprise Architect也将成为帮助您学习语言的朋友，通过提供许多工具内设备来协助学习，以及丰富而完整的模型模式库，帮助您入门，确保您创造行业最佳- 实践模型。



当您开始使用Enterprise Architect时，您会立即轻松地成为广泛的国际用户和从业者社区的一部分，他们日复一日地使用该工具来指定、设计、实施和支持系统工程模型，这些模型用于解决现实世界的问题。这些问题和机会中的许多都是复杂的，并且通常看似难以处理，但可以通过应用 SysML 来表达和解决问题的建模者的协作来解决。



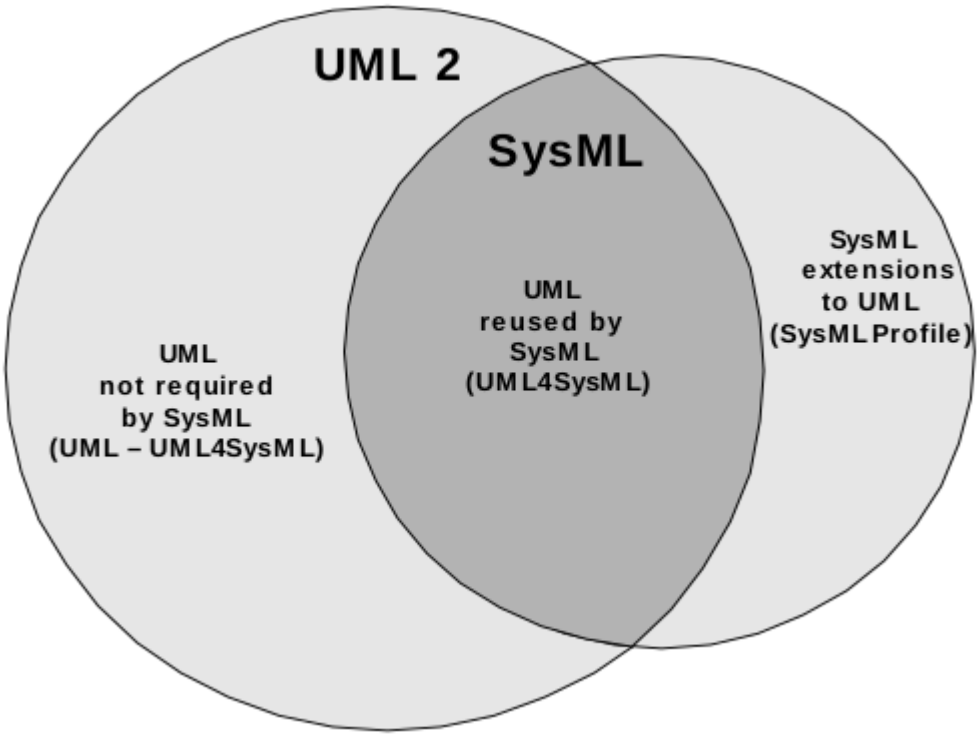
Enterprise Architect通过其丰富的桌面和云平台无缝地促进了这种协作，这有助于确保模型既健壮又富有表现力，并且不仅是一个而是任意数量的工程师和其他利益相关者的结果，无论他们说什么自然语言都可以一起工

作，他们正在使用什么设备或他们在世界上的位置。

语言架构

系统建模语言（SysML）从根本上说是一组允许人类和工具就系统工程进行交流的常规符号。它是一个国际标准，定义和描述了用于系统工程的通用建模语言。 Enterprise Architect是实现此标准并允许系统工程师应用被称为基于模型的系统工程的方法的世界领先工具之一。此外， Enterprise Architect提供了支持工程实践和管理的广泛辅助方面的工具特征。我们将在本指南中探索这些有用且富有成效的工具特征。

SysML 基于另一个标准， Unified Modeling Language (UML)，自 90 年代后期以来，软件工程师已采用和使用该标准。这一点很重要，因为许多系统工程项目都涉及系统和软件方面，因此系统和软件工程师都能够理解彼此的模型，从而提高透明度、减少错误机会和相互理解的语言结构，从而形成一个系统不太可能失败或出现故障。这个维恩图显示了两个标准之间的关系。 SysML 重用了用例、活动和序列图表。



需求驱动

系统建模语言（SysML）的创建是由用户需求驱动的； SysML 的设计响应了系统工程*Unified Modeling Language*提案请求中列出的需求。本文档为系统工程(SE) 指定了UML的定制，并要求这种定制应支持广泛系统的建模，其中可能包括硬件、软件、数据、人员、程序和功能。该文件指出：

'用于系统工程的UML的定制应该支持复杂系统的分析、规范、设计和验证：

- 以精确和有效的方式捕获系统信息，使其能够在更广泛的时间进行集成和上下文
- 分析和评估指定的系统，识别和解决系统需求和设计问题，并支持权衡
- 在各种利益相关者和参与者之间正确、一致地传达系统信息

Enterprise Architect的设计者已经阅读了这些文档和由此产生的细节规范，并创建了一个复杂且高度可用的工具来实现所有这些要求，并添加了一组丰富的附加特征，以确保组织的工程和业务成功。

-

OMG Systems Modeling Language™

Version 1.5

为了使一种语言有用且相关，它必须根据其用户社区的需求而发展。为此，SysML 规范会定期更新， Sparx Systems的团队还更新和扩展Enterprise Architect ，以确保它符合不断发展的标准，更根本地，满足其用户社区的多样化需求。

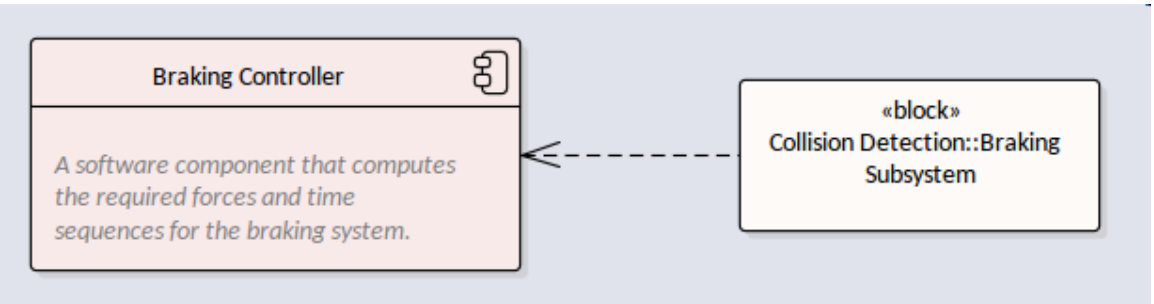
Unified Modeling Language重用和扩展

系统建模语言（SysML）建立在Unified Modeling Language （ UML ）之上。 UML已被物件管理组 (OMG) 批准和采用，他们继续作为规范的保管人。 2005 年，国际标准化组织 (ISO) 也发布了UML作为批准的 ISO 标准。该语言为以软件为中心的系统建模提供了规范。 SysML 语言可以追溯到 2001 年，起源于一个源规范，但是当国际系统工程委员会 (系统工程) 开始与 OMG 合作时，OMG 于 2006 年采用了 SysML 的最终版本。

在许多方面，SysML 在理论上是更原始的语言，因为它是一种一般建模语言，而UML更专业，旨在为以软件为中心的系统建模。然而，历史和语言的起源颠倒了这一立场。实际上，SysML 是使用UML分析系统创建的，并且是UML的扩展子集。这意味着 SysML 没有采用所有的UML ，它还定义了一些额外的语言结构。我们在前面部分中看到的维恩图在数学上描述了两组相交的语言结构。

SysML 规范的Enterprise Architect实现高度兼容，开发人员与规范密切合作，并与各行各业的行业专家、思想领袖和系统工程社区保持持续沟通。这导致了一个世界级的工具，它不仅实现了规范，而且还提供了广泛的附加工具，例如可执行状态机、参数模拟、甘特图、看板、思维导图、战略模型以及数百个其他特征.

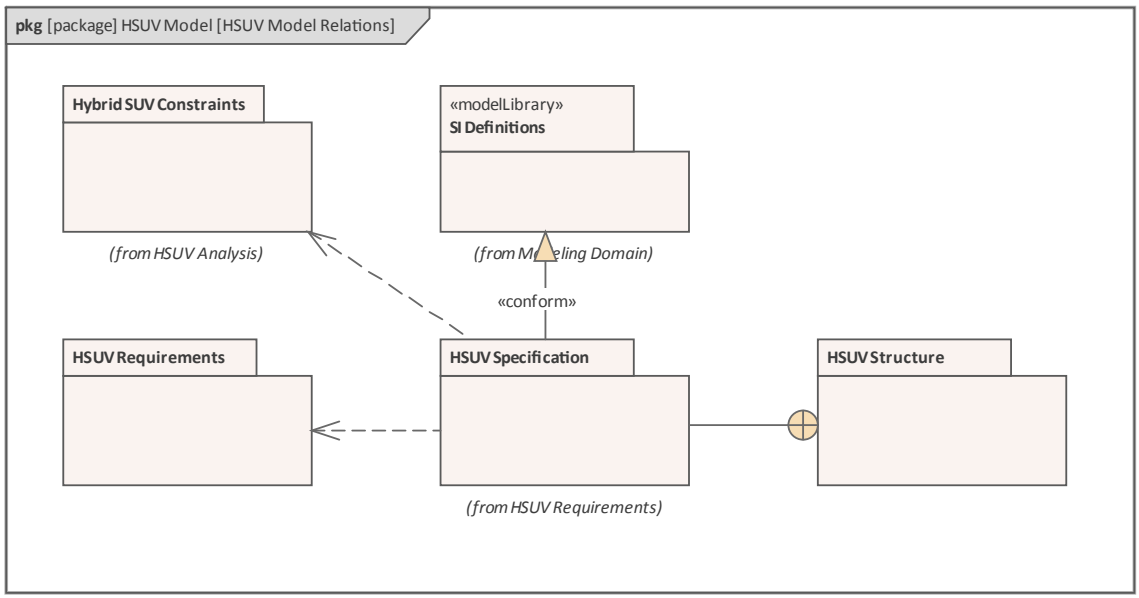
此外，从铁路系统到航空系统、能源系统等广泛学科中，系统和软件工程问题和解决方案之间的交互作用不断增加。 Enterprise Architect具有独特的定位，因为它支持这些学科的强大特征以及它作为架构工具的优势。



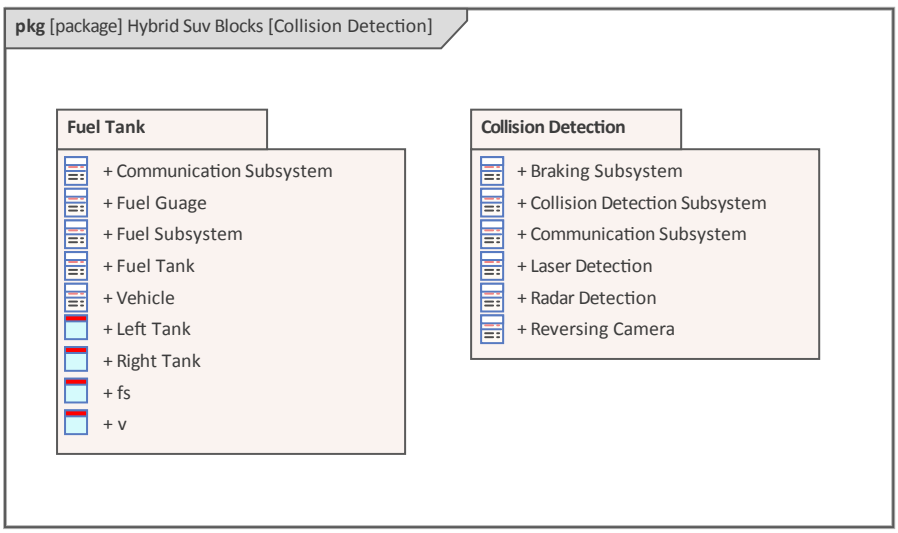
带包分区

包是语言中划分的基本单元，旨在防止循环依赖。语言被正式划分为模型元素集，这些模型元素在逻辑上对元素进行分组并允许语言用户将元素理解为语言单元的集合。

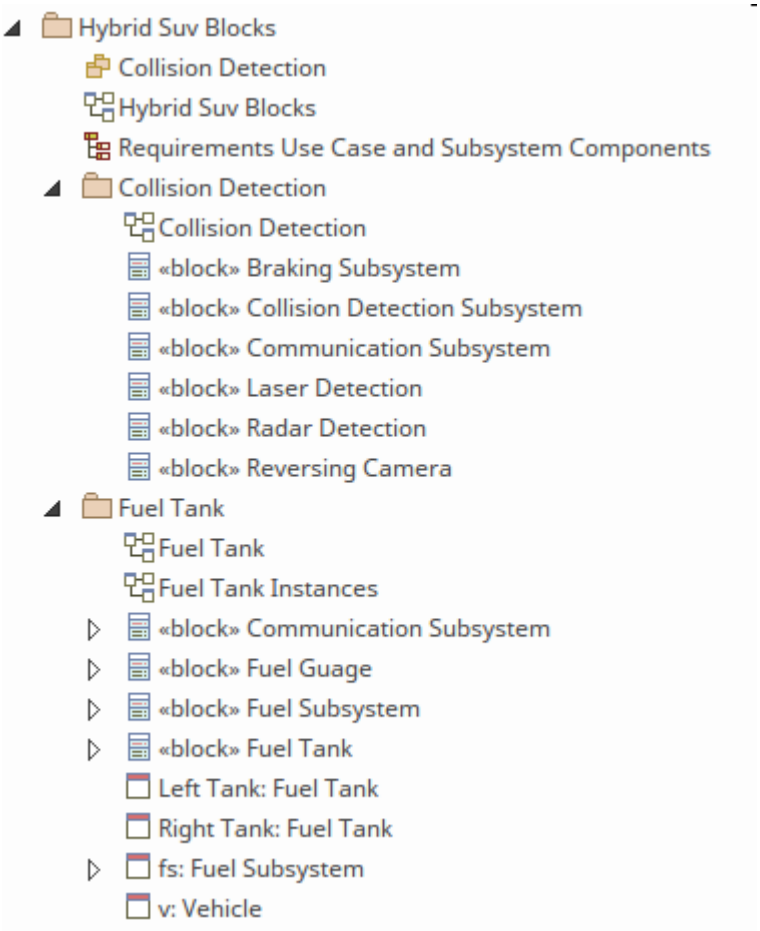
它们也是用户定义模型中的基本结构单元，并作为一种一般机制，用于根据用户定义的因素对元素进行分组。形式上，它们可用于指定名称空间，这在某些建模构造中很重要，例如 XML 模式的定义或代码生成。包可以在浏览器窗口或图表中创建和查看，两个位置提供不同的方式与包一起工作。图表可用于直观地显示包的内容或描述包之间存在的关系。



建筑师包提供了多种图表展示方式，帮助用户理解包与包含的元素和图表之间的结构关系。当包中包含包裹时，该工具允许用户从多个显示选项中进行选择，并且可以更改隔间可见性以显示包内容。在此图中，作者希望显示两个包的内容，这些内容在不太可能发生的碰撞事件中具有重要意义。在元素隔间可见性中选择了 显示包内容”选项，从而清楚地显示每个包中包含哪些元素。

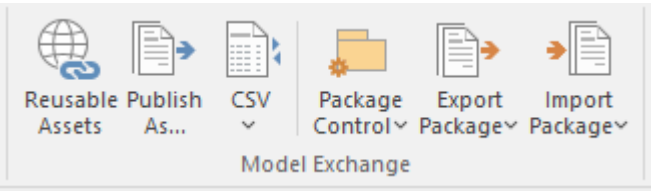


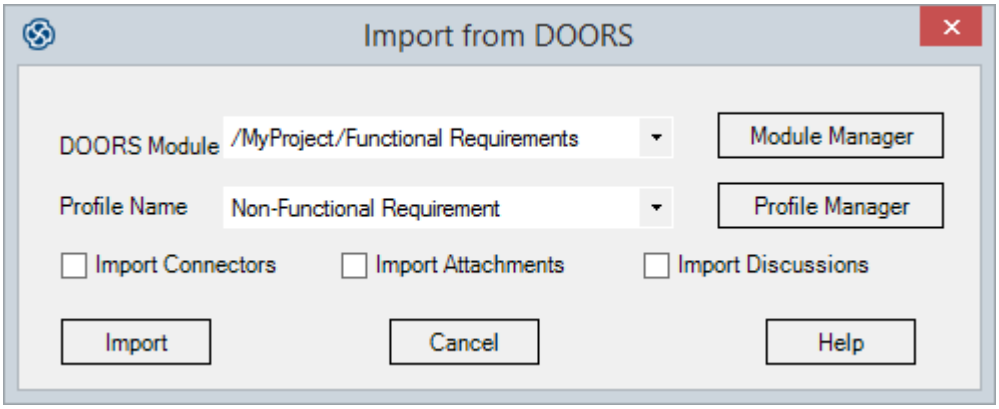
相同的包及其内容可以在浏览器窗口中查看，重要的是要记住，虽然可以在报告等出版物中包含图表，但浏览器窗口的内容在这些文档中是不可见的。



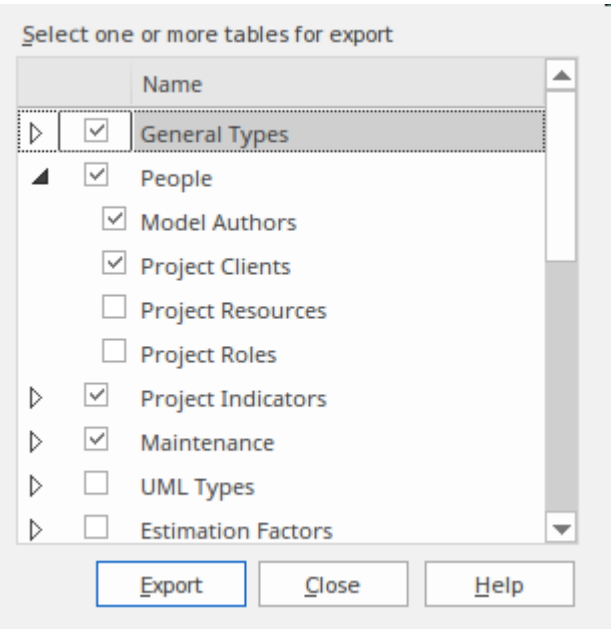
互 操作 性与模型交换

Enterprise Architect是领先的 SysML 工具之一，具有一组必要的特征，但设计人员意识到，组织将需要使用各种工具来完成复杂的业务和工程任务，这些任务在 21 世纪中的每个组织都面临着世纪。为确保重要的工程和业务信息可以与其他工具和平台进行交换，丰富的模型交换支持符合ISO 10303-233数据交换标准，支持其他工程工具之间的互操作。这是基于UML XMI 交换功能实现的，该功能在包级别等级的工具中得到支持，允许任何包及其包含的层次结构与其他兼容工具进行交换。

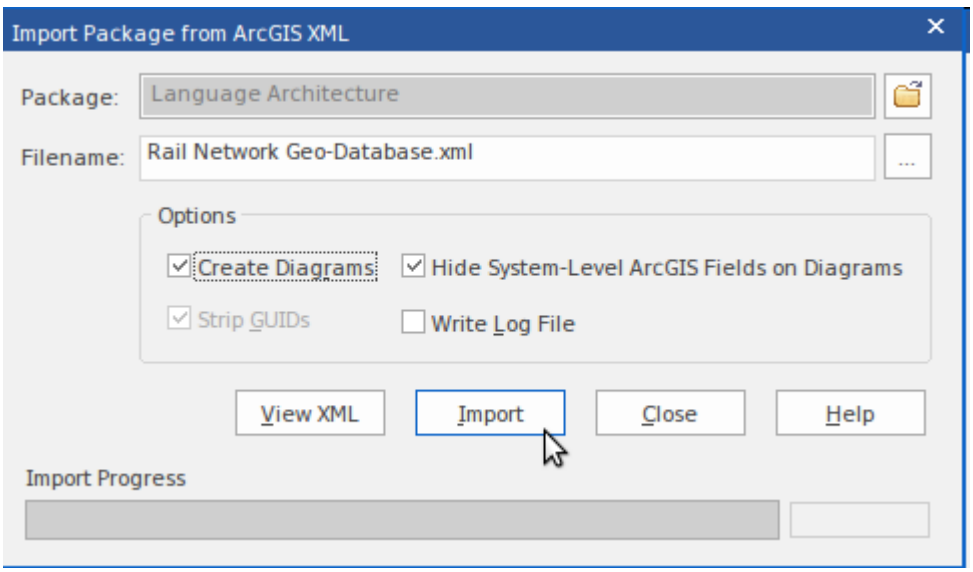




Enterprise Architect比这更进一步，它提供了具有各种业务、项目管理、分析和项目交付工具的交换机制。这是在建模工具级别实现的，提供使用 CSV 文件格式交换电子表格中包含的数据，以及文字处理器中的文本。参考，如优先级列表、状态、复杂性、约束和其他数据，如词汇表角色和作者、日历等都可以从存储库中导入和导出。

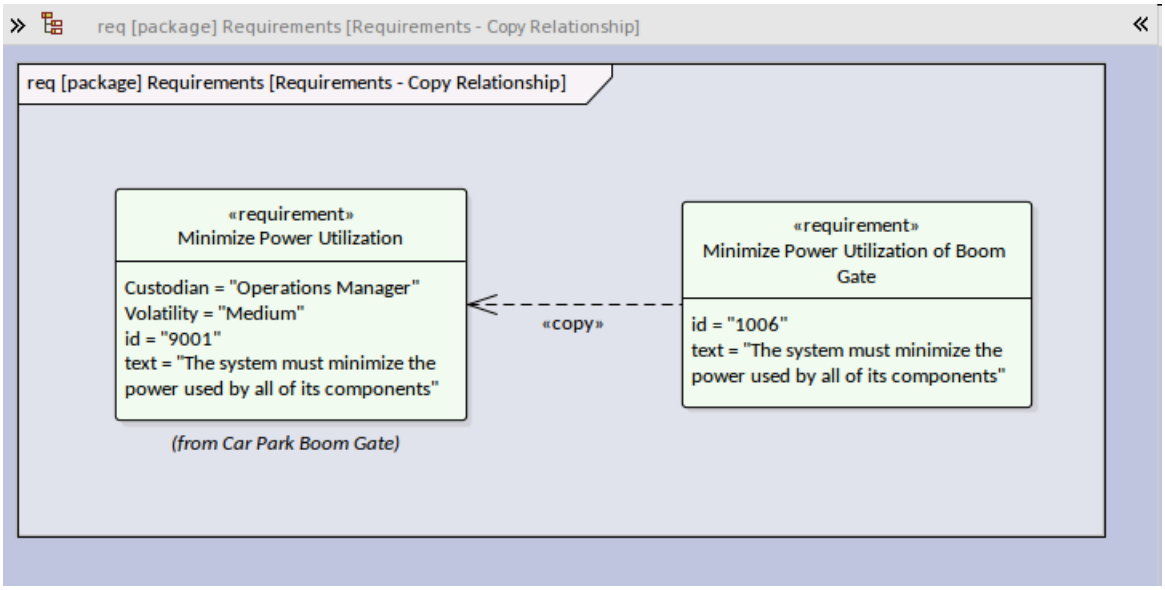


地理空间信息形成了一个关键数据集，在这个世界中，地理位置几乎是每个项目和计划的重要方面。 Enterprise Architect提供与领先的地理空间建模工具的数据交换，允许一起查看和管理两个以前完全不同的异构数据集。

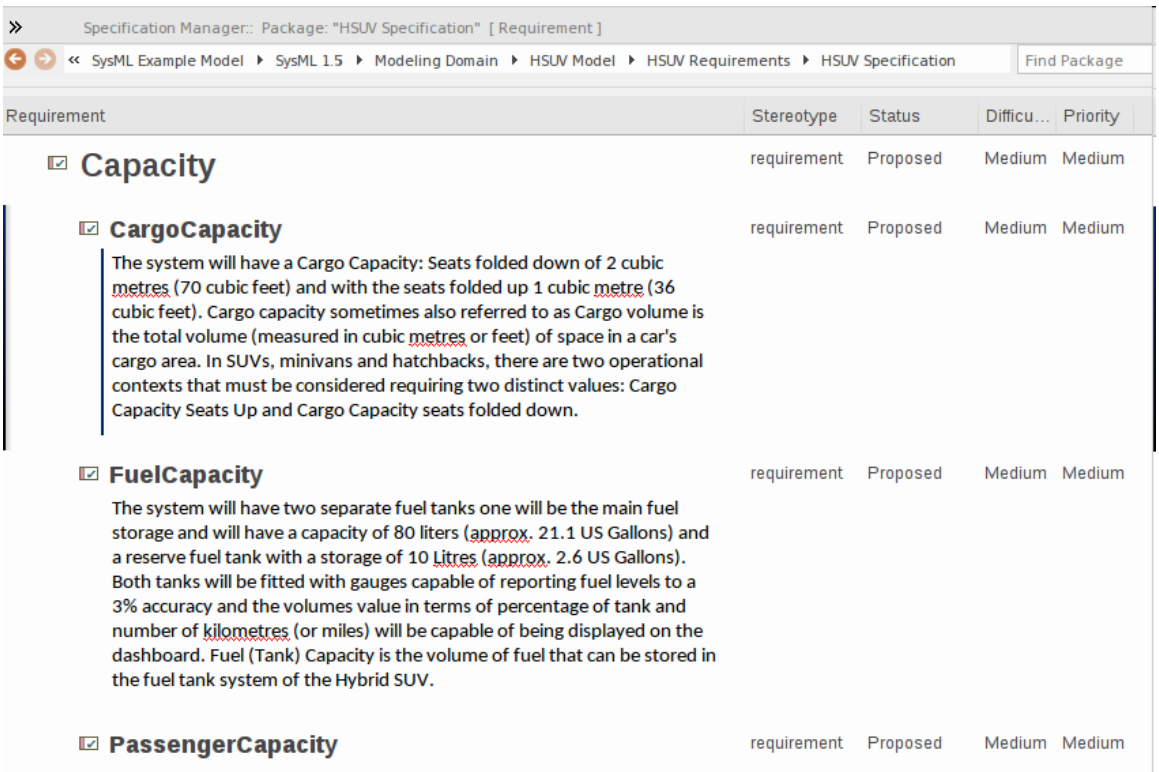


关键语法概念

SysML 与其父级UML一样，是一种可视化语言，其中图表是该语言沟通策略的核心。尽管语言的重点在于这种视觉编码和思想的传递，但语言还具有功能表达思想的功能，这是对视觉机制的重要补充。许多元素，**例如需求元素**， A视觉形式，但是需求的详细信息写在称为 需求文本”的属性中，如图所示。



Enterprise Architect也经过精心设计，尊重不同用户处理信息的方式。设计团队与其用户社区密切合作，并意识到一些用户使用图表可视化效果更好，而其他用户使用文本效果更好。 Enterprise Architect中提供的许多工具在设计时都考虑到了这些不同类型的用户。例如，在需求管理学科中，用户通常更熟悉在文档和电子表格中工作。为了满足这个需求， Enterprise Architect有许多用户可以切换到的视图，允许他们通过这些类型的界面输入、编辑和管理需求。其中一个工具是规范管理器，它从基于文档的视图和电子表格视图中提供了一种灵活且熟悉的需求处理方式，允许轻松查看、创建和管理需求。



模型、图表、元素和视图

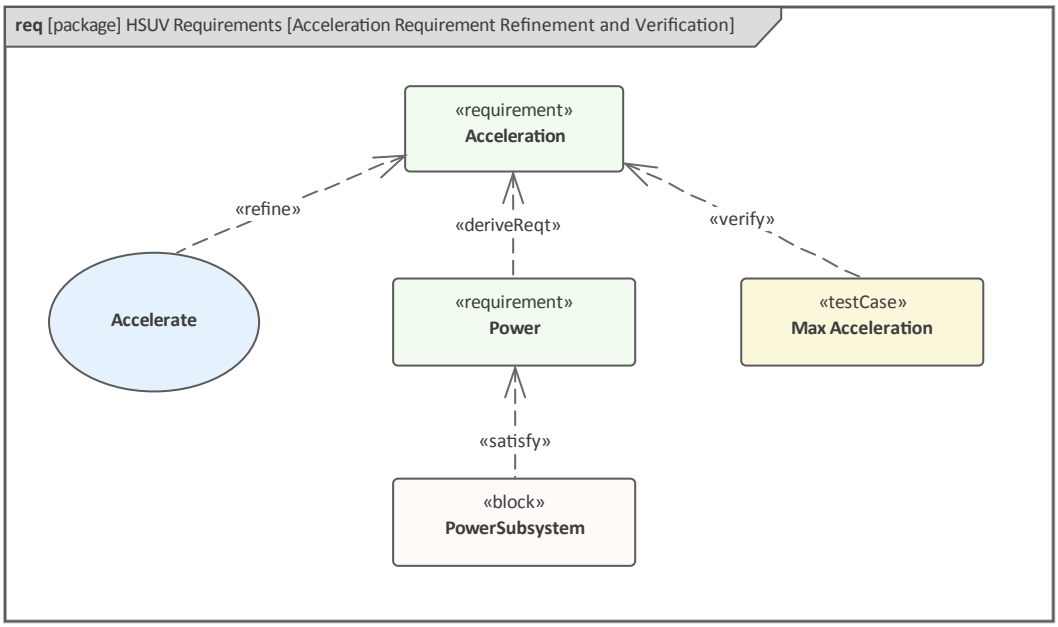
楷模

模型这个词被过度使用并且意义重大。有些人使用模型来表示整个存储库，而其他人则使用它来指代完成存储库的一部分。模型是存储库中的结构划分。

图表

SysML 规范定义了九种图表类型。这是规范列表，并且 SysML 还定义了通常在每个图表上使用的元素。许多新手甚至一些有经验的用户都没有意识到，尽管这些元素列表描述了特定图表类型的常用元素，但这并不妨碍建模者在这些图表上使用其他元素。事实上，在同一个图表上使用多种元素类型会产生一个富有表现力的模型，并使来自不同学科的利益相关者和工程师能够理解模型之间的图表间联系。

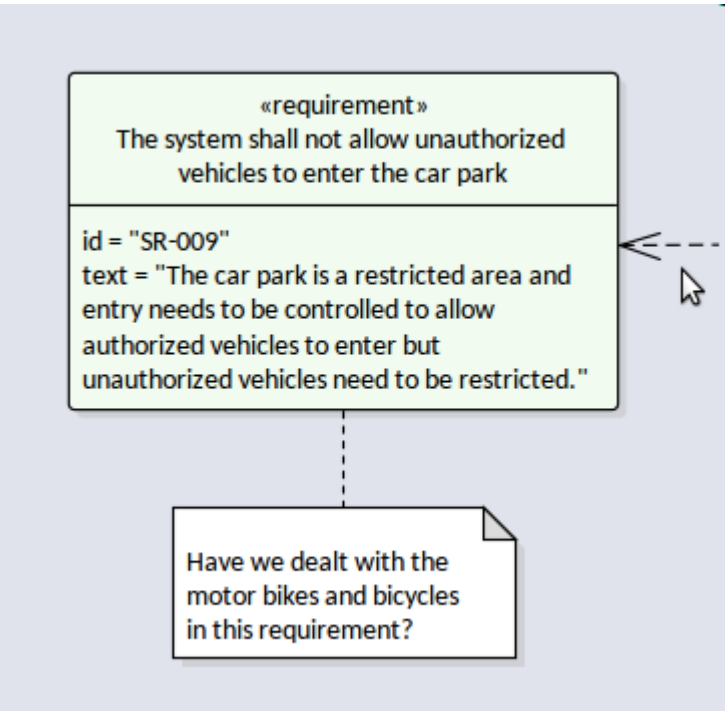
在本节中，我们还将了解到规范建议可以将许多 通用”模型元素合并为任何图表的一部分，包括注释、约束和基本原理。此图显示了一系列元素类型，包括块、用例、需求和测试用例，都在块定义图表(BDD) 上表示。



如前所述，系统建模语言指定了九种不同类型的图表。

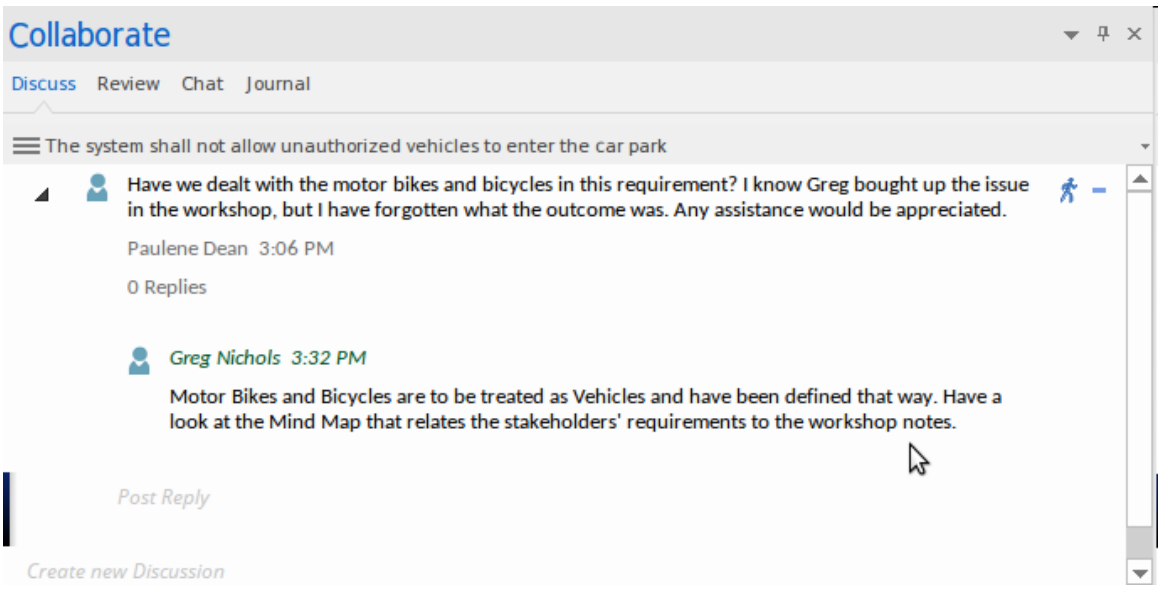
元素

前面描述的九个图表是单独指定的，以传达工程机会或解决方案的特定方面；例如，参数图旨在显示方程是如何构造的。然而，有许多类型的元素对模型工作来说是通用的，并且可以出现在任何类型的图表上。这些元素中的许多元素用于图表以向模型传达重要的注释或帮助解释模型的特定方面。它们包括诸如注记、约束、基本原理和视图。在此图中，一位一直在查看模型的利益相关者添加了评论以质疑模型的一部分。



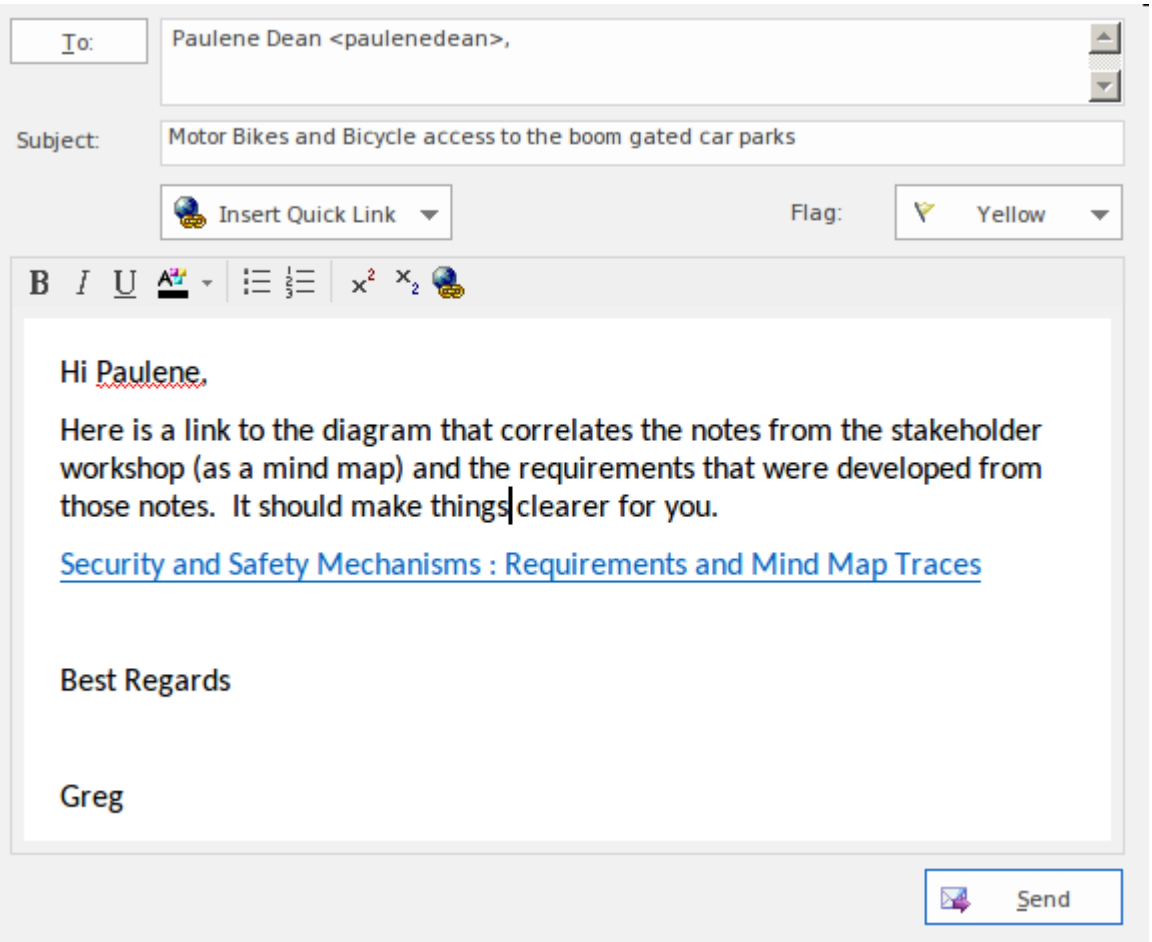
有关详细信息，请参阅Common Pages帮助。

虽然Enterprise Architect高度符合 SysML 规范，但它具有许多协作特征，允许管理此类评论，例如通过其讨论特征。这允许将讨论与构成模型的元素分开。此屏幕图像显示了使用讨论特征添加的相同评论，该评论允许回复和一系列其他协作设备。



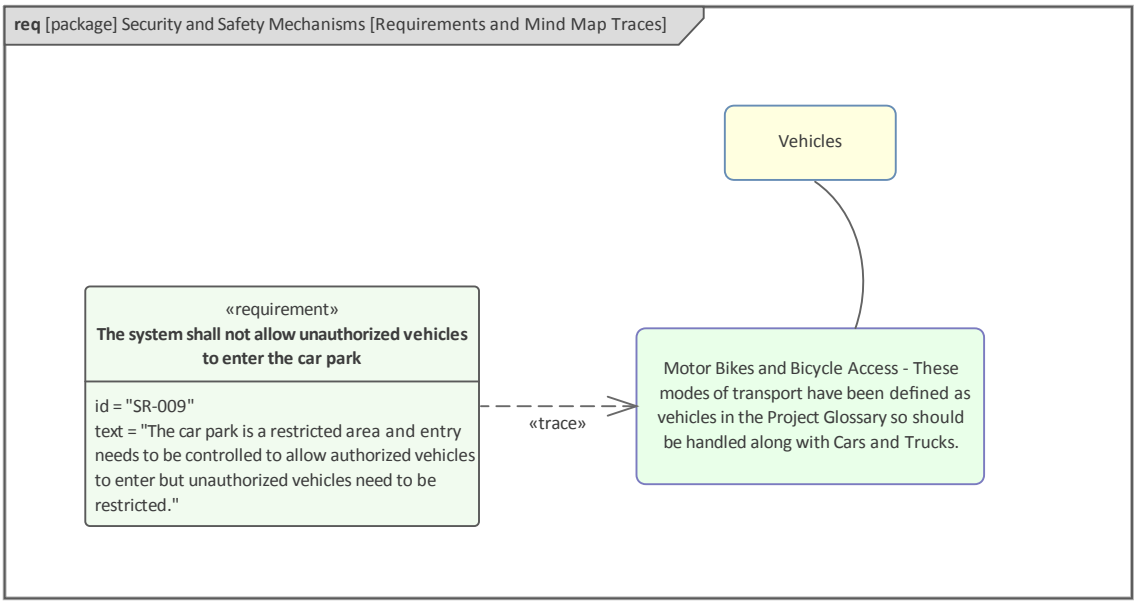
有关详细信息，请参阅Discussions帮助主题。

响应者还可以发送模型邮件消息，其中可能包括指向Enterprise Architect元素或图表的链接，如图所示。



有关详细信息，请参阅Model Mail帮助主题。

当用户打开邮件消息时，他们能够跟随链接并打开链接中引用的图表。这是一种有用的机制，允许引用和访问建模信息的动态和实时视图，而不是在静态文档中发送图像。此图像显示了打开链接时工具中将显示的图表。



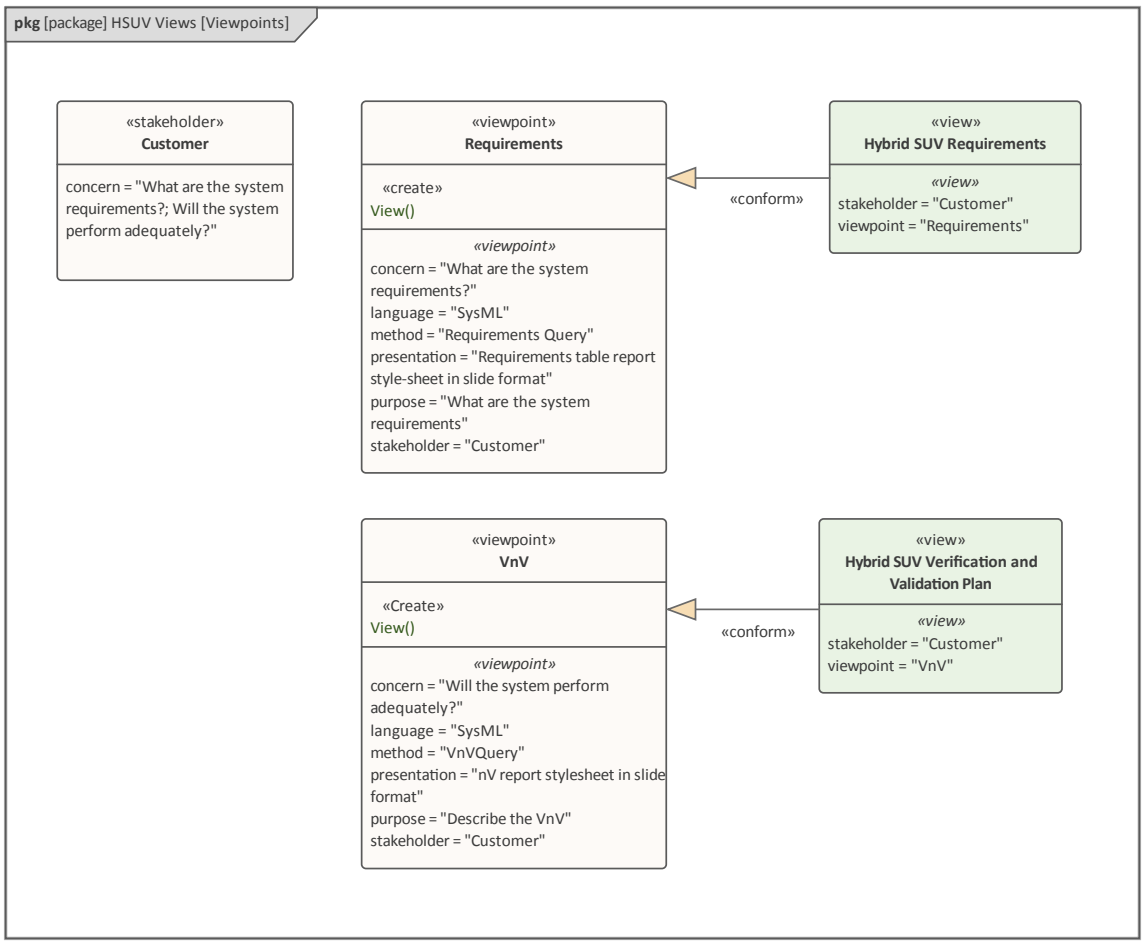
链接来自不同建模领域的元素的能力是Enterprise Architect作为团队和学科的统一平台的一大优势，这在业务战略和工程之间以及软件开发和工程之间的关系中表现得尤为明显。结果是一致且协调的模型，其中不同团队之间的接缝导致故障的可能性显着降低。有关详细信息，请参阅Traceability Window帮助。

视图

从根本上说，系统是为其利益相关者构思、分析、设计和构建的。系统工程师从利益相关者那里收集关注点和兴趣，并应用分析来创建需求和约束。这些用作分析和设计的输入，在系统交付验证和验证之前使用。利益相关者需要能够可视化他们的利益如何在工程过程的各个阶段得到解决，并且这种可视化可以通过视图和观点来提供。视点和视图的概念在 ISO-42010（以前的 IEEE-1471）中进行了阐述，并且 SysML 规范的编写与 ISO-42010 标准保持一致。有许多常用的观点，包括：

- 操作
- 表现
- 制造业
- 安全

观点是构建A观点的处方，该观点将解决给定利益利益相关者的需求、利益和关注。视图是利益相关者从给定A角度所看到的，应该使他们能够看到系统中他们关心的部分，同时忽略或模糊系统中不感兴趣的方面。



除了 SysML 规范以视图和视点元素的形式描述的正式机制之外， Enterprise Architect还具有广泛的工具来帮助创建和管理视点、视图和表示。有几种工具可用于创建存储库中元素的不同视图；这些包括工作集和模型视图。工作集允许将图表、矩阵、团队图书馆和其他项目的集合作为一个集合保存和重新打开，这在与不同的利益相关者组合作时很有用。有关更多信息，请参阅Working Sets帮助主题

模型视图可用于创建组合在一起的元素的视图，无论它们在浏览器窗口中的位置如何。还有一些工具可以隐藏或隐藏图表的某些部分，以使其对特定受众更具吸引力。图表的外观可以通过改变元素的外观来改变，包括使用图像，并且图表Filters 可以隐藏或隐藏元素的视图。有关详细信息，请参阅Visual Filters帮助主题。文档引擎可以直接将模型创建为 pdf 或 docx 格式的高质量出版物。有关更多信息，请参阅Documentation帮助主题。

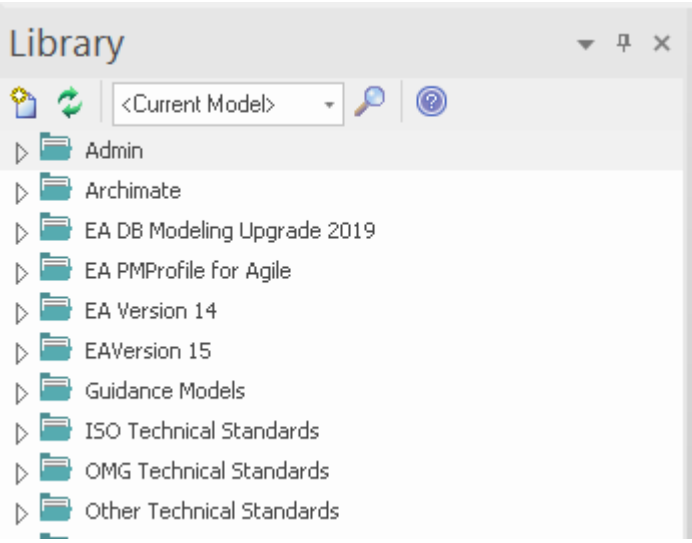
作为工程团队进行协作

工程团队是多学科的，由战略家、经理、系统工程师、软件工程师、测试人员等组成。发布产品或提供解决方案的商业压力意味着团队必须更聪明、更有凝聚力地工作，以确保工程成果。 Enterprise Architect从头开始构建为一个协作平台，不仅适用于工程师，也适用于所有学科。它促进个人一起工作，并通过全方位的工具、评论、聊天到版本控制和团队的模型共享信息、设计和解决方案以及基线。

中央共享库

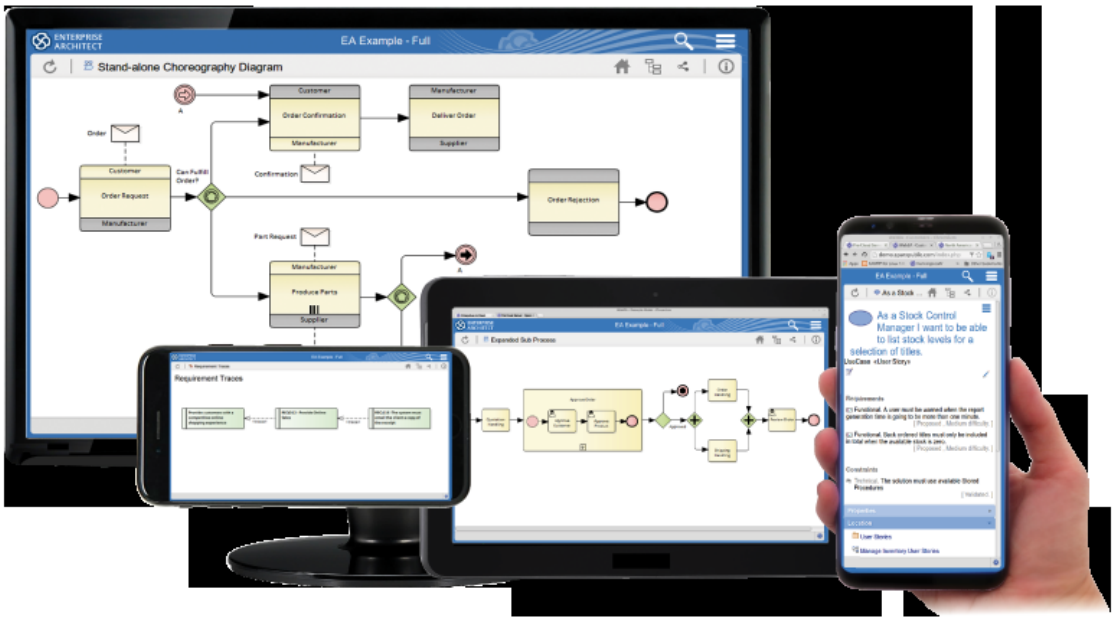
Enterprise Architect不仅是一个复杂的绘图工具，而且是一个建模平台，允许以编程方式验证、模拟、转换和操作模型。这是可以实现的，因为该工具不以光栅或矢量格式存储图表，而是将图像与所有参考数据和其他存储库元数据一起编码在关系数据库中。该平台使用此关系数据库来存储所有建模信息和元数据，并提供可供客户端和基于 Web 的工具访问的后端存储。有关更多信息，请参阅 The Model Repository帮助主题。

该产品架构允许用户共享模型、图表和其他存储库信息，例如参考数据、图像库中的图像和团队图书馆中的文档以及许多其他工具特征，这些特征有助于在同一地点或分布式团队中进行协作和协同工作.有关更多信息，请参阅帮助主题The Modeling Team。



云计算

可以使用适当的数据库连接层 (ODBC) 和随附的驱动程序直接从Enterprise Architect桌面客户端访问上一主题中描述的中央存储库。但是，还有另一种更简单的方法可以访问托管在云中的存储库，即通过使用 http 或https协议的云服务。云可以在内部或外部，并且可以通过多种方式配置云。还有一个可以服务器的包含多个模块的服务器，称为专业云服务器，它提供了多种工具、功能和方法，可以通过计算机或手持设备（如平板电脑）上的浏览器查看模型或智能手机。有关详细信息，请参阅Pro Cloud Server Repositories主题。

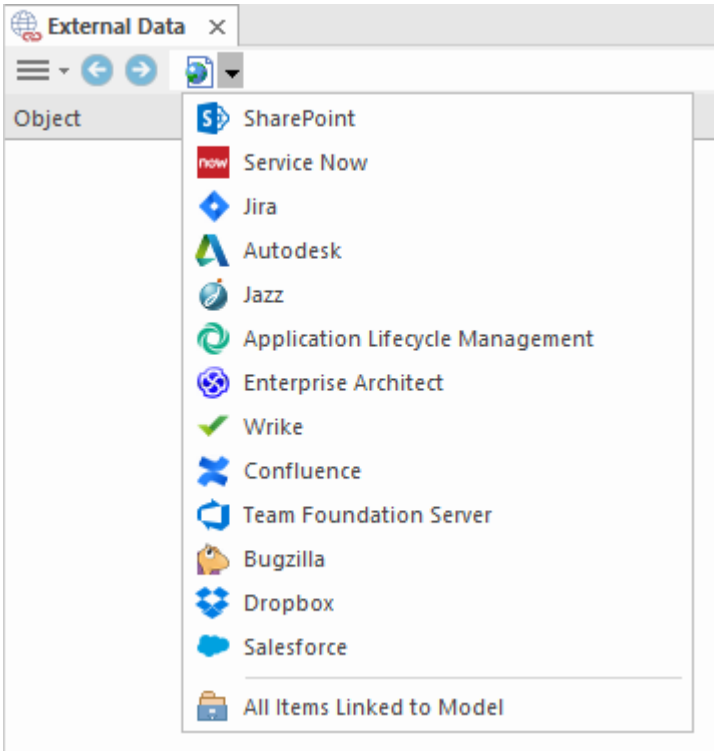


这为与本地或分布在全球的各种利益相关者（从工程经理到咨询工程师）合作提供了一个平台。这种工作方式的力量可以通过最少的设置来实现，并且让所有团队成员和利益相关者查看并为相同模型做出贡献的好处不可低估。

能够获得有价值和及时的反馈，或者在他们乘火车前往贸易展览时与工程顾问讨论图表，或者在研讨会休息时与工程负责人讨论图表，可以提高生产力 - 所有这些实时并在智能手机或平板电脑的模型中。

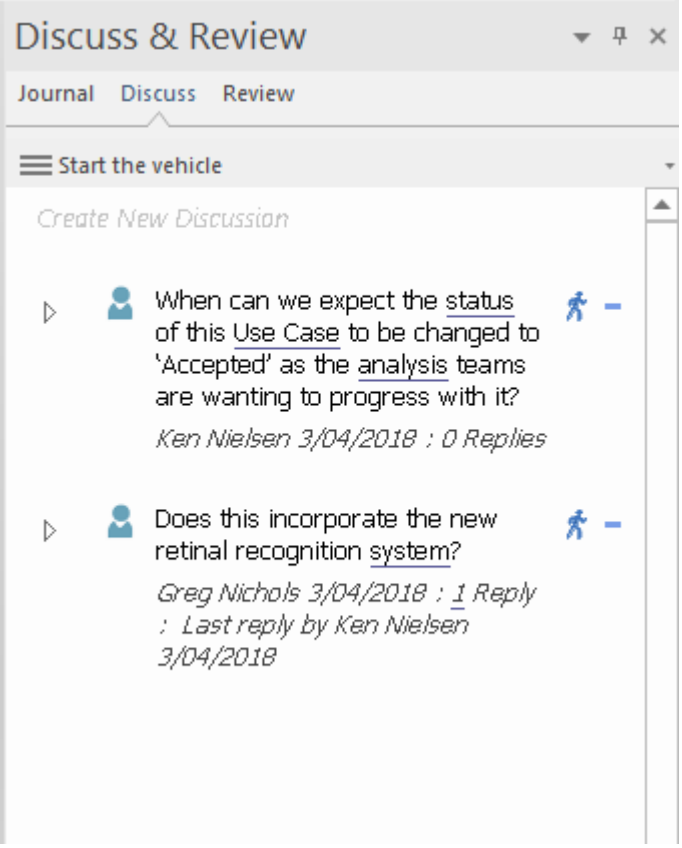
专业云服务器还提供与各种其他工具和平台的连接，包括需求管理系统、配置管理工具、问题跟踪系统、项目管理系统等。

此屏幕截图显示了一些可用集成的列表，尽管此列表会定期添加，因此值得检查是否包含其他集成。有关详细信息，请参阅Integrate Data from External Providers的数据主题。


External Data
Object
SharePoint
Service Now
Jira
Autodesk
Jazz
Application Lifecycle Management
Enterprise Architect
Wrike
Confluence
Team Foundation Server
Bugzilla
Dropbox
Salesforce
All Items Linked to Model

讨论和聊天

协作概念的核心是建模者与同事或行业和标准专家讨论问题或解决方案的能力。 Enterprise Architect允许工程师、经理和其他人参与有关元素、图表和连接器的讨论。可以创建A帖子来启动一个线程或对话，然后其他建模者可以通过回复进入。讨论与元素和图表元信息分开保存，允许在不影响从模型生成的文档或报告的情况下进行丰富和建设性的评论。讨论和聊天是可用的两个选项，讨论和审阅窗口中的讨论和聊天和邮件窗口中的聊天。

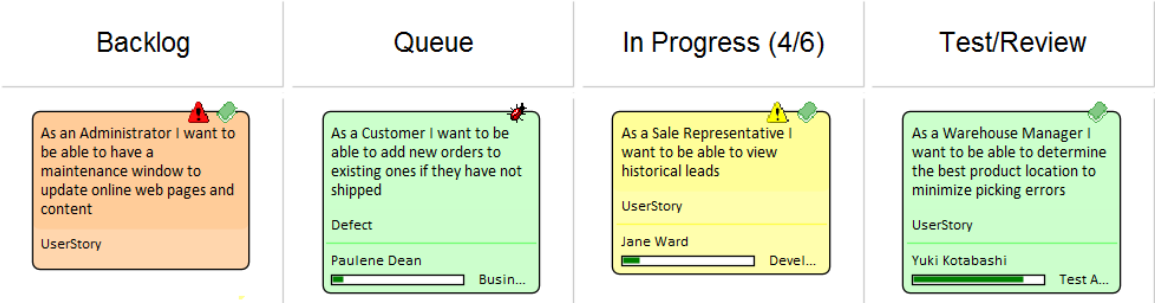


聊天对于与已定义为安全定义用户组的一部分的同事或专家进行快速和响应式通信很有用。聊天不像讨论那样与模型元素相关，而是全局性的，当打开聊天和邮件窗口并选择一个组时，项目按日期时间顺序列出。有关详细信息，请参阅Teams & Collaboration帮助。

看板资源和日历

看板技术已在Enterprise Architect中实施，可极大地提高团队的生产力以及软件和系统工程项目或任务集的项目管理。在Enterprise Architect中，它是一个简单易用的特征，使您能够管理积压中的项目并将它们移动到任意数量的通道，甚至移动到其他板，代表流程中的阶段。该功能可以整合到现有的或新的工程或软件开发过程中，从而带来前所未有的效率。

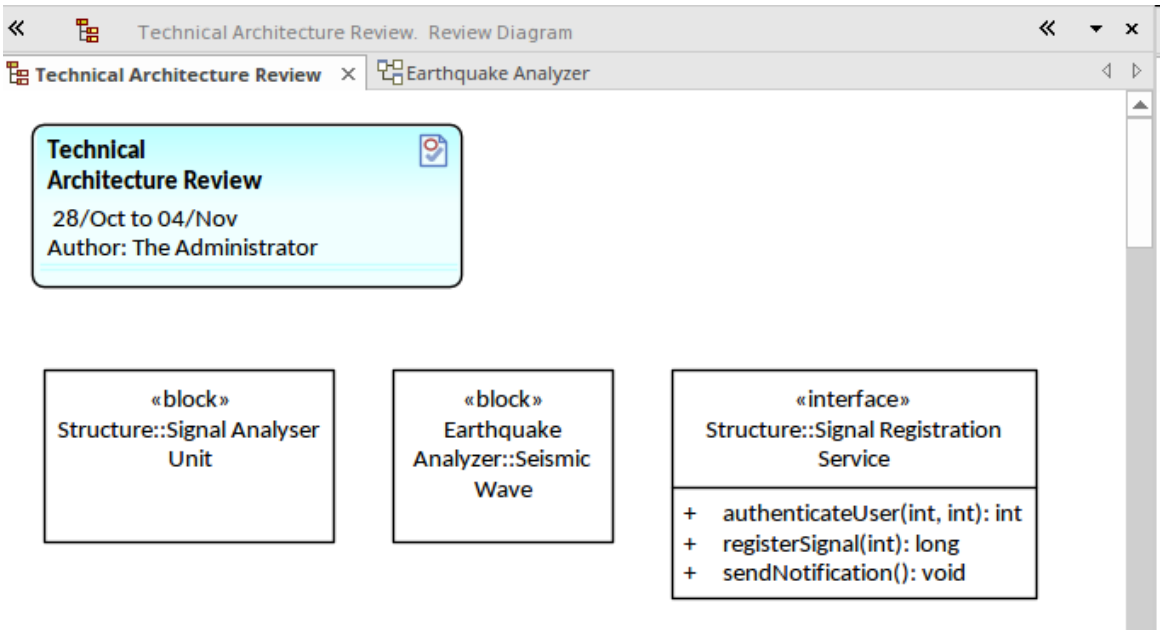
使用此特征的一大优势在于，参与看板图的元素可以链接到存储库中的其他元素，从而实现从需求到战略意图或设计组件的完全可追溯性下至已发布产品的一个元素。当项目通过任意数量的链接看板板时，看板可用于可视化项目完成的资源、工作量和状态。这是一个与仓库系统工程项目相关的软件开发过程的示例。



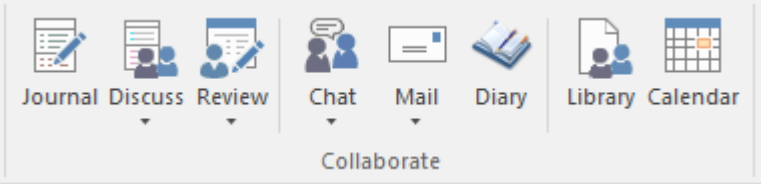
有关更多信息，请参阅Kanban Boards和Resources & Work Items帮助主题。

模型评论

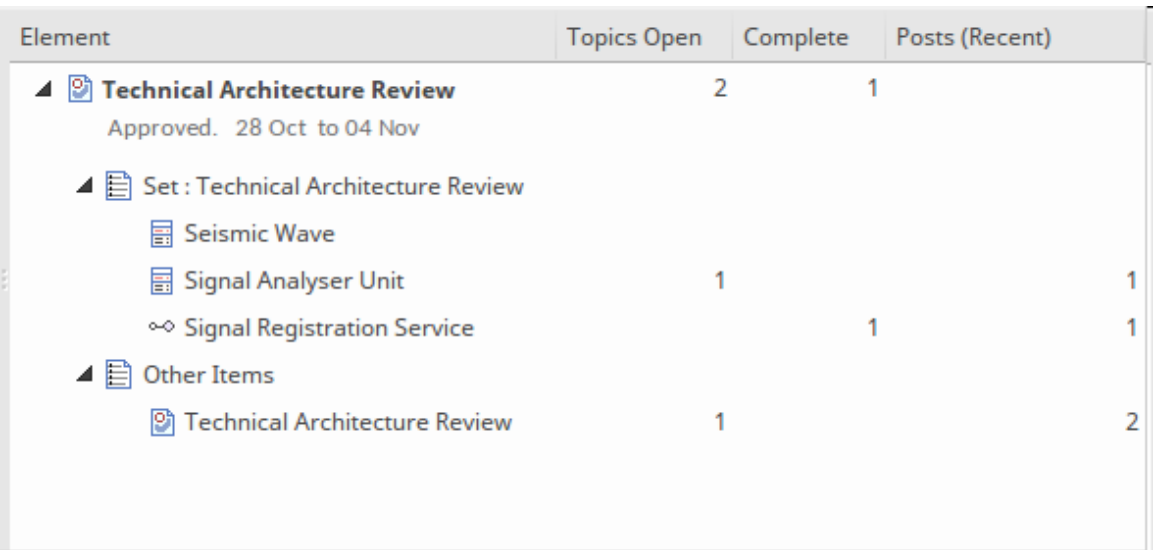
模型审查使项目利益相关者能够在模型内容的评估（审阅）中正式协作，包括元素和图表。这个方便的协作工具利用了许多内置特征- 例如审阅视图 - 来管理审阅过程并可视化讨论和对审阅的贡献。审阅图为参与者提供了A将元素和图表添加到审阅的机制。此图包含许多与审阅主题相关的元素。



可以创建任意数量的评论，建模者可以加入和参与评论。审阅功能的启动板是开始功能区的 协作”面板。

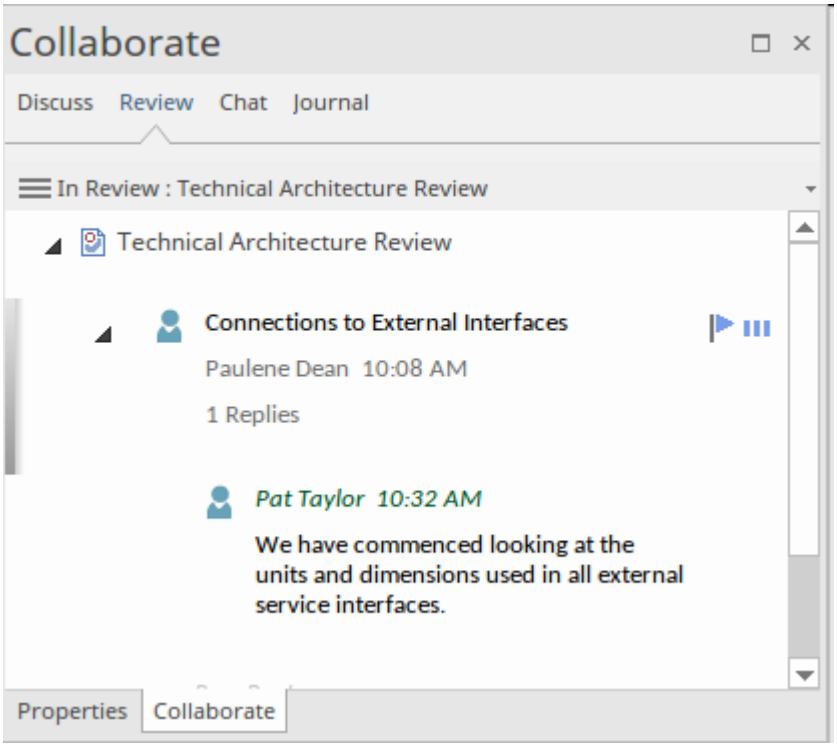


选择 审阅> 管理评论”选项将打开评论视图，其中将列出所有现有评论，并且 - 通过鼠标右键单击 - 可以创建新评论。该视图在左侧面板中显示了审阅元数据（包括开始和结束日期），在右侧面板中显示了审阅详细信息，包括构成审阅部分的元素和图表以及每个项目的帖子和回复。

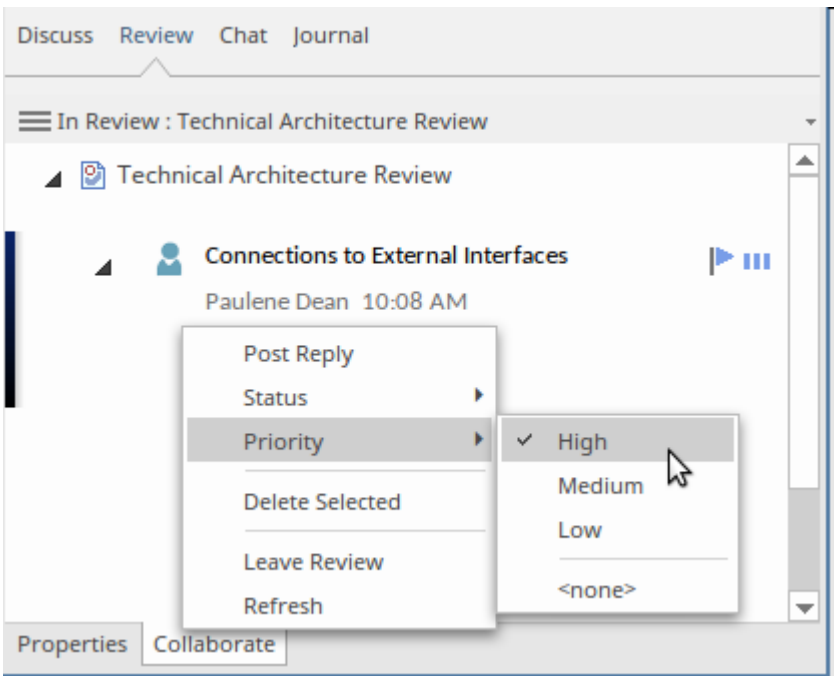


可以在讨论和审阅窗口的 审阅”选项卡中输入和查看审阅评论。这些将记录所有帖子和对帖子进行注释的回复，并回复作者和日期。通过这种方式，可以实现高度协作的结果，并且 - 通过智能手机和平板电脑上可用的

WebEA产品 - 外部顾问和行业专家等利益相关者可以为审阅做出贡献，而无需使用Enterprise Architect桌面应用程序。



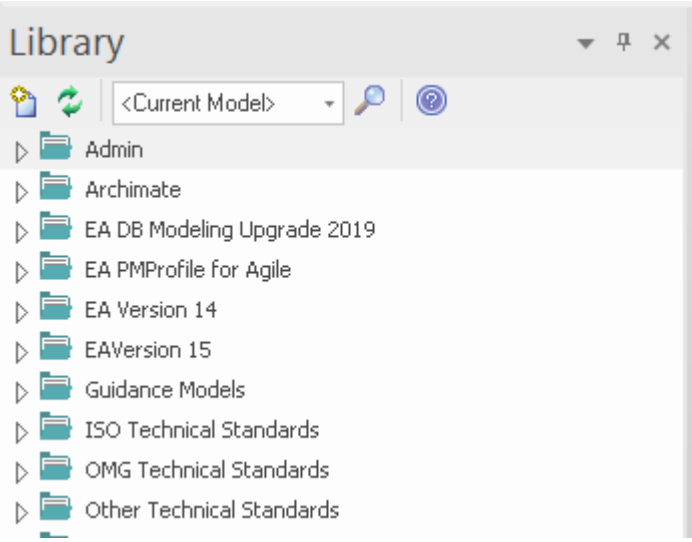
审阅视图中的每一项都可以有多个属性集，包括状态项的属性和优先审阅；这些可以看作是项目右侧的两个小图标（一个标志和一个数量图标）。



在团队图书馆中共享资源

使用以基于模型的系统工程为中心的流程的团队将始终依赖政策、方法、说明、流程描述、指导文档、标准和其他类型的工程或项目文档形式的大量文档。其中一些是基于文档的，而另一些是基于资源的，可在内部网络、共享系统、Intranet 或 Extranet 或更广泛的公共 Internet 上使用。

无论文档或页面位于何处，它们都可以导入Enterprise Architect或通过 URL 作为外部资源引用。它们可以作为内部图表或外部工件包含在图表中，但也可以更方便地从团队图书馆中引用。



还可以考虑将这些文档中的部分或全部项目作为第一类模型元素导入。例如，可以将策略创建为元模型元素，并且可以导入策略列表，从而允许将单个策略跟踪到特定的系统组件。有关更多信息，请参阅The Model Library帮助话题。

在移动设备上查看模型

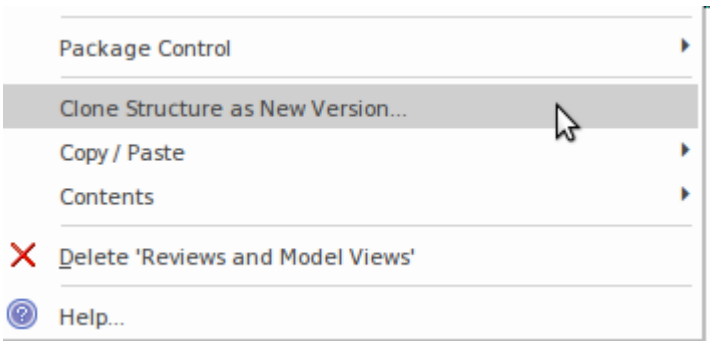
近年来，现代工作场所发生了显着变化，组织鼓励灵活使用办公桌和家办公的形式，导致越来越多的人使用便携式设备工作。此外，在创新和颠覆的推动下，现代世界的变化步伐每年都在加快——例如，一场流行病影响了供应商、客户或同事所在的每个国家，要求人们在各地的办公室缺席，并且无法出差直接咨询或提供专业知识。 .战略家、技术人员和工程师需要协作以实现工程成果，而在分散的劳动力中，这通常意味着他们需要在移动和受限隔离下通过移动设备为模型做出贡献。

Enterprise Architect存储库内容可以通过平板电脑或智能手机等移动设备上的浏览器实时查看。这允许工程师、经理和其他人在会议间隙、场外检查、公共交通或任何他们碰巧在的任何地方进行协作。这是以前从未有过的，现在不需要在等待审查或人们返回办公室时放慢项目的速度；模型可以从任何地方以互联网速度更新。有关详细信息，请参阅Pro Cloud Server Repositories帮助。

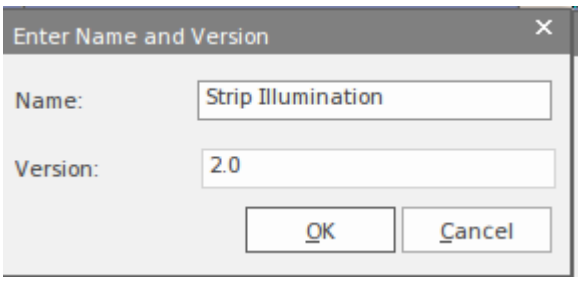
建模未来

世界对变革和创新有着永不满足的需求，这导致系统工程师需要更快地工作，并想出聪明的方法来解决问题。时间感知建模是一种独特的功能，允许工程师和其他利益相关者对任意数量的未来状态进行模型。将状态引入模型提供了一种机制来可视化提议的解决方案可能是什么样子，并允许工程师比较许多提议的解决方案。可以分析和推理A定的解决方案，并且可以在模型中识别潜在的缺陷和问题。任何数量的利益相关者都可以进行审查和演练，这些可以用来确定许多解决方案中的哪一个是合适的。

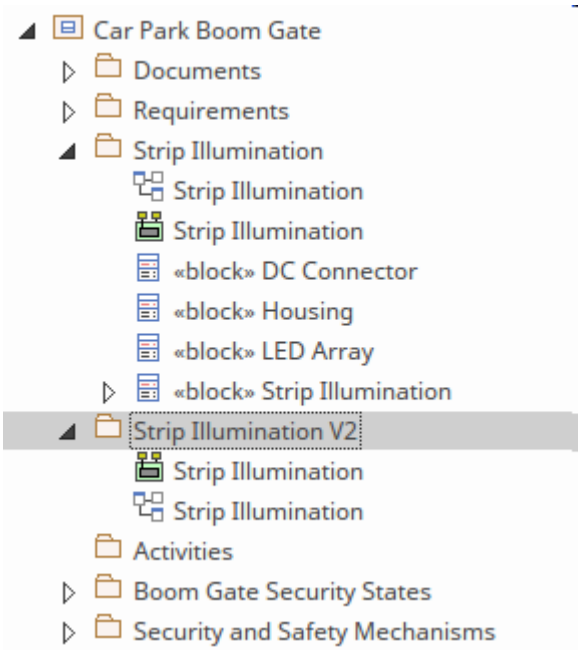
该过程可以通过克隆您需要未来状态模型的包来启动。这可以使用功能区选项或从浏览器窗口上下文菜单完成。



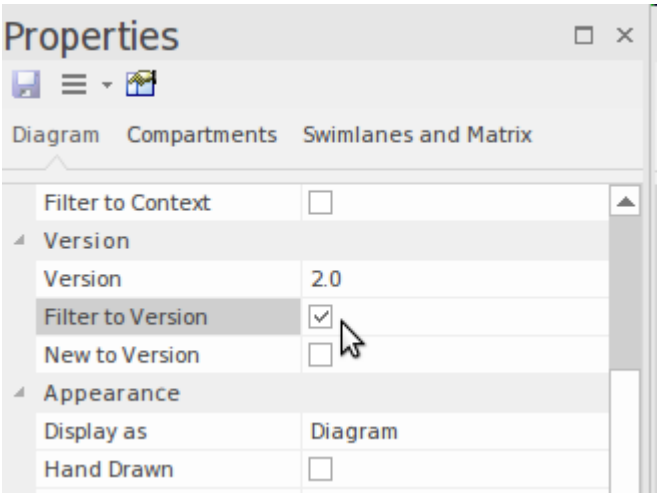
该工具将显示一个提示，允许工程师为克隆的结构指定名称和版本号。版本号对特征的操作至关重要，Enterprise Architect使用它作为跟踪元素和图表的方式，这些元素和图表构成具有指定版本号的未来版本的一部分。该工具还将提示克隆包结构在存储库中的位置。通常可以给它一个名称，包括它的版本和被克隆包的包含的父项；但是，它也可能保存在为未来状态版本指定的存储库的单独部分中。



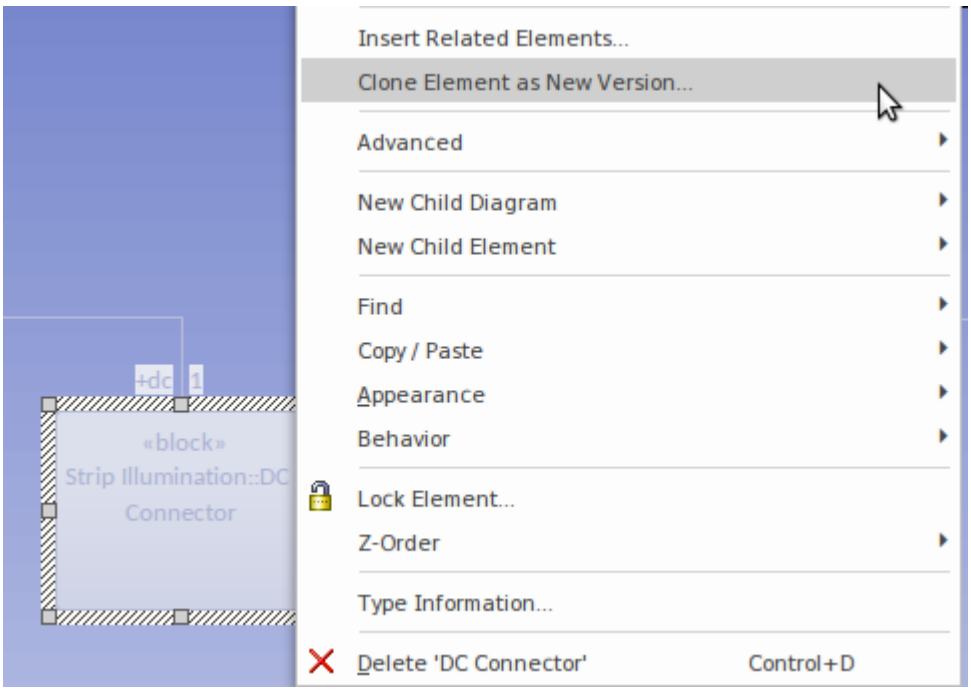
刚刚执行的步骤只是设置了克隆的结构，本身并没有创建元素的未来状态；但是，它确实会复制包中包含的任何图表。此插图显示了一个已克隆的包，其中包含两个图表，它们是原始包中图表的副本。



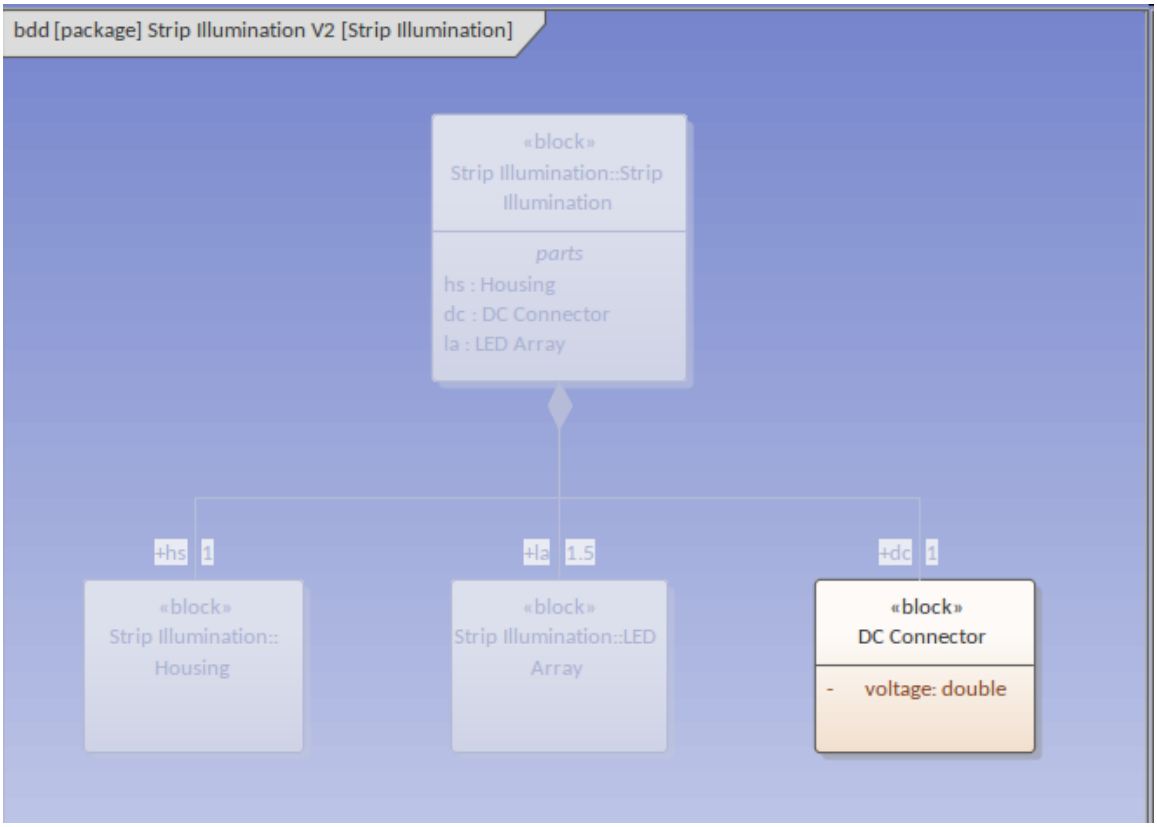
该图被分配了版本对话框中指定的版本号，由用户在上一步中设置。单个元素的克隆是通过在图表上选择一个元素来完成的，**但在此之前**，图表上的所有元素都将是以前的版本。 Enterprise Architect具有可从图表属性表获**得的功能**，它允许建模者将图表上的元素过滤到图表的版本。



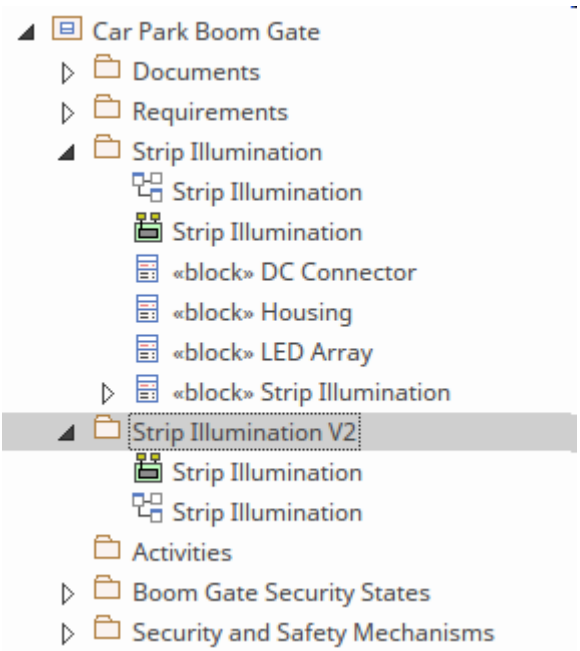
在新克隆的图表上设置此属性将以灰度显示所有元素，因为它们都来自以前的版本。从这一点开始，**可以克隆**单个元素， Enterprise Architect将复制所选元素，创建其所有连接器的副本。这很重要，因为一旦实施更改，它将允许将元素提升为更新的当前版本。可以通过选择图表中的元素并显示上下文菜单来克隆单个图表元素，**如**下图所示。



将再次提示您输入版本号，方便该工具将默认为克隆包选择的版本。一旦这被接受， Enterprise Architect将创建所选元素的副本，并且由于图表仍被过滤为版本，此元素将正常显示在图表中，其他元素仍以灰度显示。



浏览器将显示新创建的（版本 2）元素，该元素将与图表进行核对。这样，克隆的包将只包含具有新版本号的元素和图表。欲了解更多信息，请参阅Time Aware Modeling 帮助 话题。



版本控制和基线

我们生活和工作在一个以互联网速度发展的世界中，因此工程问题和机会也以同样的速度变化。在这个时间范围内记录、分析、构思和实施工程解决方案是工程师面临的挑战。这意味着几乎在工程师完全描述问题之前，它就会发生变化，或者问题的业务或工程上下文会发生变化。大多数情况下，新方向或更改将被描述和采用，但其他时候工程师将需要回溯并返回到问题、机会或解决方案的先前版本。 Enterprise Architect拥有复杂的工具来执行这种回溯。

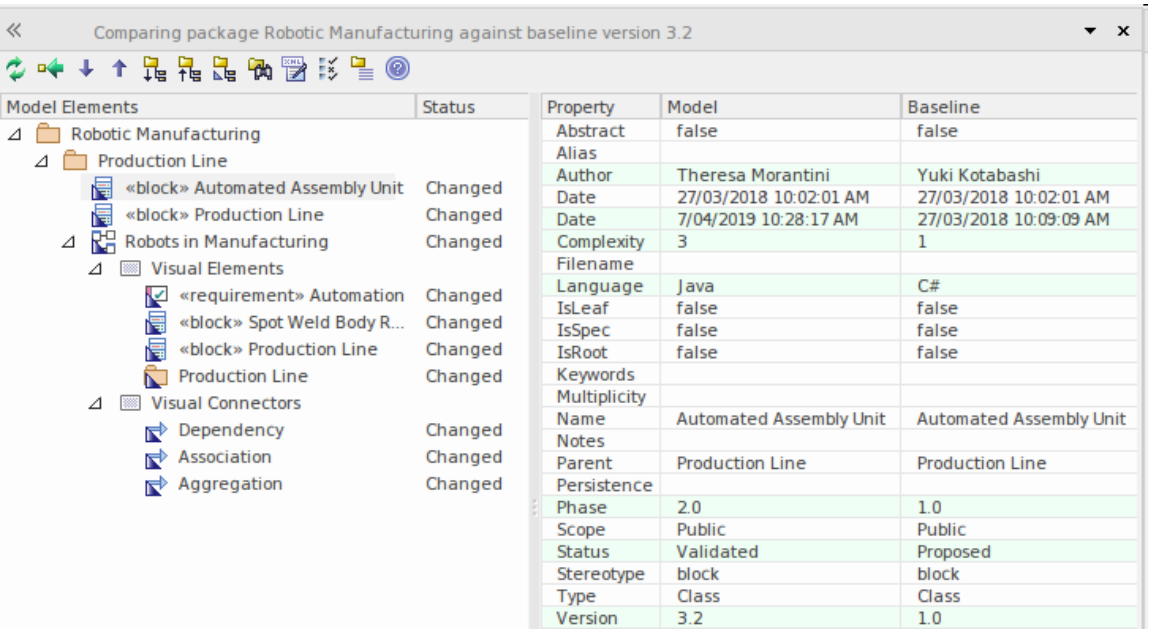
Enterprise Architect有两个基本工具用于处理先前版本的建模内容。

- **版本控制-一旦配置，任何更改点都可以返回**；用户签出模型、进行更改并签入片段，从而创建版本 - 有关**更多信息**，请参阅Version Control帮助主题
- **在模型中的**基线或重要的发展点创建；用户可以创建一个基线的内容，并且可以将其与基线的任何级别的模型进行比较 - 并且先前恢复的更多信息，请Baselines 帮助话题。

有一个重要的区别，那就是两种方法基线必须有意地创造。例如，如果一个工程师团队创建了一个模型并由产品所有者签署，然后工作继续进行到相。如果出于某种原因，团队想要相并返回一个可能的模型，而没有基线**的地点**，这将是不可能的。相比之下，**一旦配置了版本控制**，同一个团队只要知道它的完成日期，就可以轻松地返回到第一个相。

基线

基线是确保团队能够回溯到模型中的里程碑的重要点或模型的有效方法。它们提供了一种用户驱动的变更管理**方式**，并让建模团队感到舒适，如果他们因某种原因或问题的某些方面偏离轨道，机会或解决方案发生变化，他们需要回到之前的点可以很容易地完成。重要的是要记住，基线只是存储库中一个包（**可能包括子包**）**的快照**，但它必须是有意创建的，并且需要在它所代表的时间点创建。欲了解更多信息，请参阅Baseline Tool帮助。



版本控制

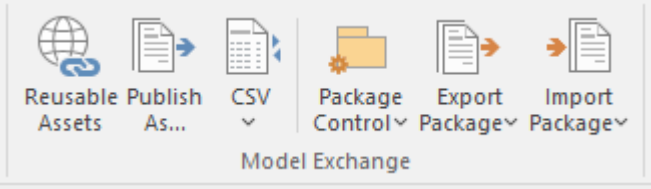
Enterprise Architect允许工程团队通过将单个模型包、视图节点或根置于版本控制下来管理项目的更改和修订。**版本控制在**Enterprise Architect中通过任意数量的管理访问的第三方源代码控制应用程序进行配置到并存储受控包的修订。一旦安装并配置了版本控制软件，团队就可以保存对包的更改历史记录、查看和检索工作的先前修订、签出和签入正在处理的内容等等。此功能允许团队协作工作，同时为工程师提供一种独立的方式来处理模型的特定部分。有关详细信息，请参阅Version Control帮助主题。

可重用资产服务器

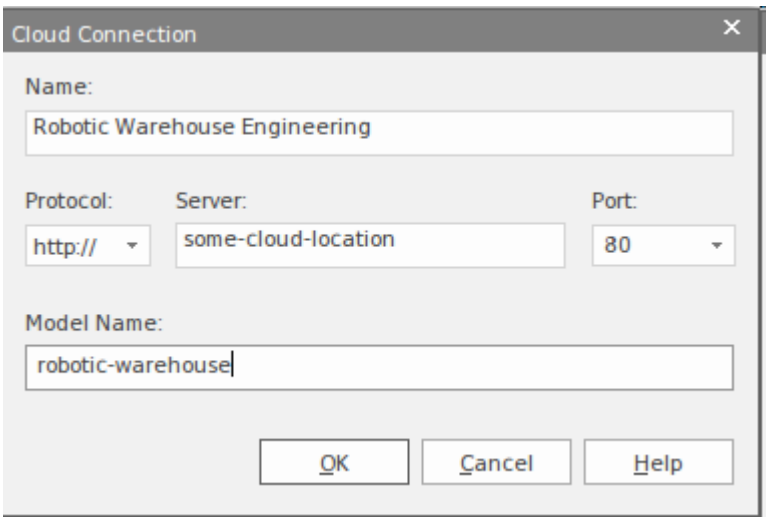
可重用资产服务器(RAS) 是一种团队生产力特征，它允许团队将建模内容存储在分布式建模者组可以访问以进行重用的位置。任何团队或组织都可以设置 RAS、存储内容并 - 通过安全设置 - 使其可用。存储的原子单元是一种资产，既可以是基于建模的信息，**也可以是基于文件的信息**：

- 包包含在存储库中，可在浏览器中查看
- 各种文本、代码和图形格式的文件，**包括** .qea **文件**

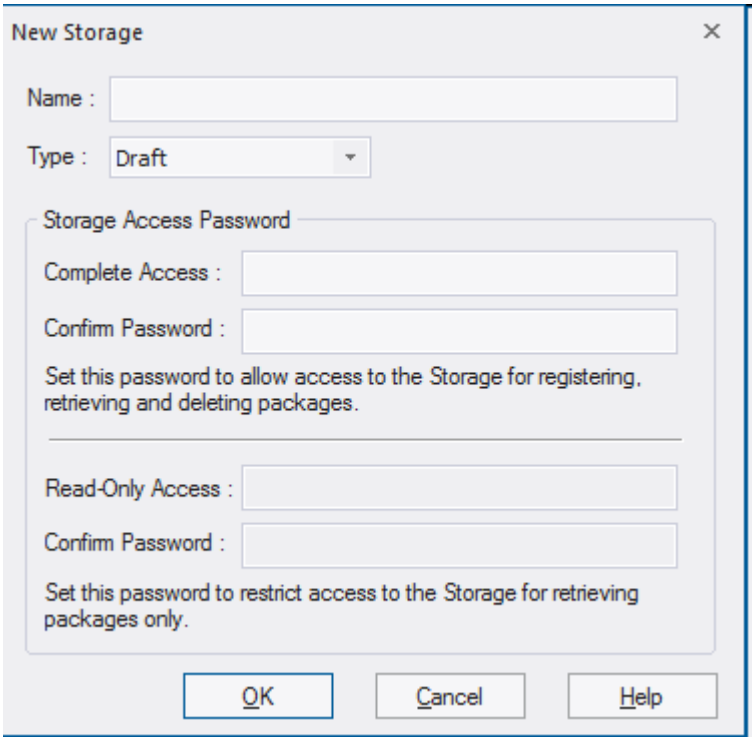
可重用资产服务器可从 "发布"功能区使用 "模型交换"面板获得。这使这项服务触手可及。



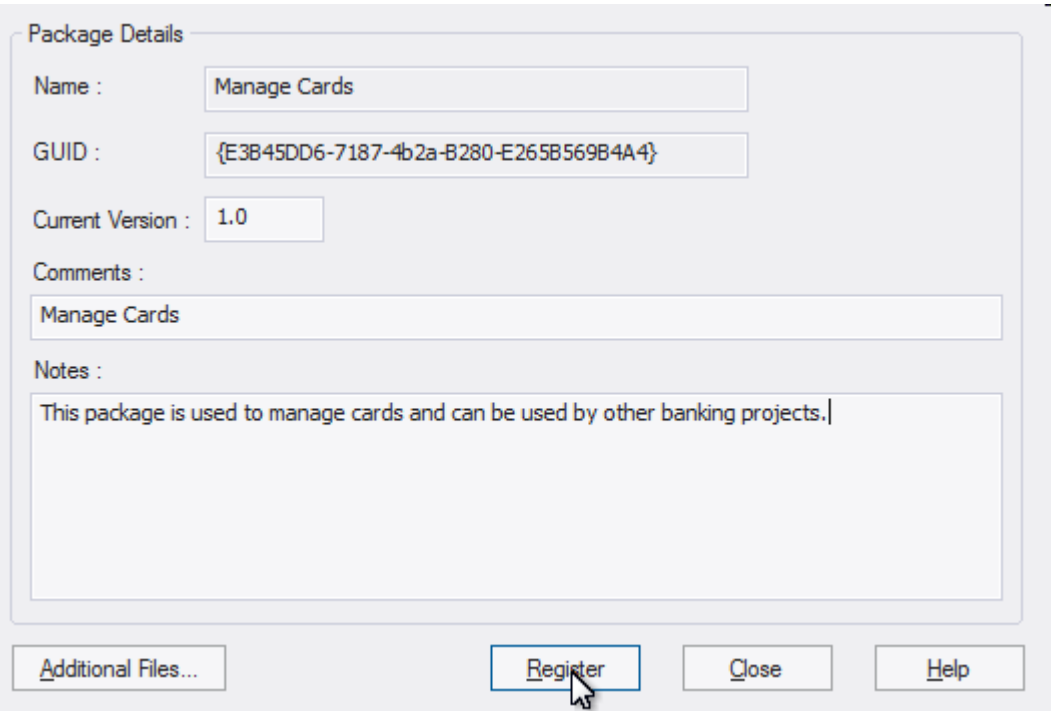
资产存储在云中，需要指定与为此目的设置的Pro云服务模型的连接。通常，此任务由 IT 部门的基础架构部分执行，而如何连接的详细信息将简单地提供给工程团队。此屏幕截图显示了建立基于云的连接所需的详细信息。



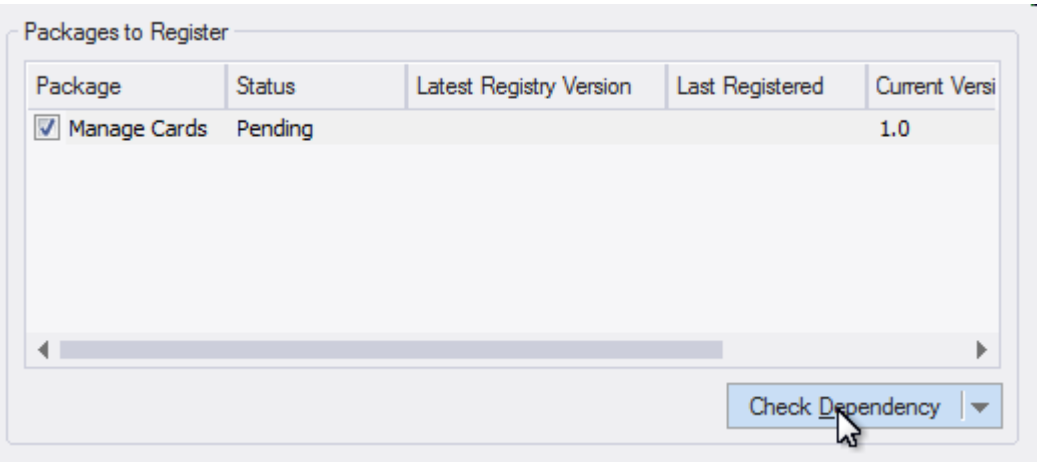
设置服务器后，可以向其中添加任何资产。形式上，服务器是一个注册中心，内容被设置为在服务器上注册。



可重用资产服务器可用于在项目之间存储信息和建模资产，对于存储信息以在项目或程序之间重用特别有用。当一个项目向业务交付了它的价值时，项目工件被归档并有效地使其他团队无法访问是很常见的。可重用资产服务器是存储这些工件的便利位置，以便其他团队可以重用它们。例如，为新医院或升级医院开发模型的项目可以将这些有价值的建模工件存储在 RAS 中，然后在另一家医院工作的任何时候，它们都可以重复使用，从而可能节省数千小时的工作时间。此图显示了在 RAS 中注册的包的详细信息，包括版本号、用户和注释，这将帮助工程师浏览可重用内容找到他们正在寻找的资产。



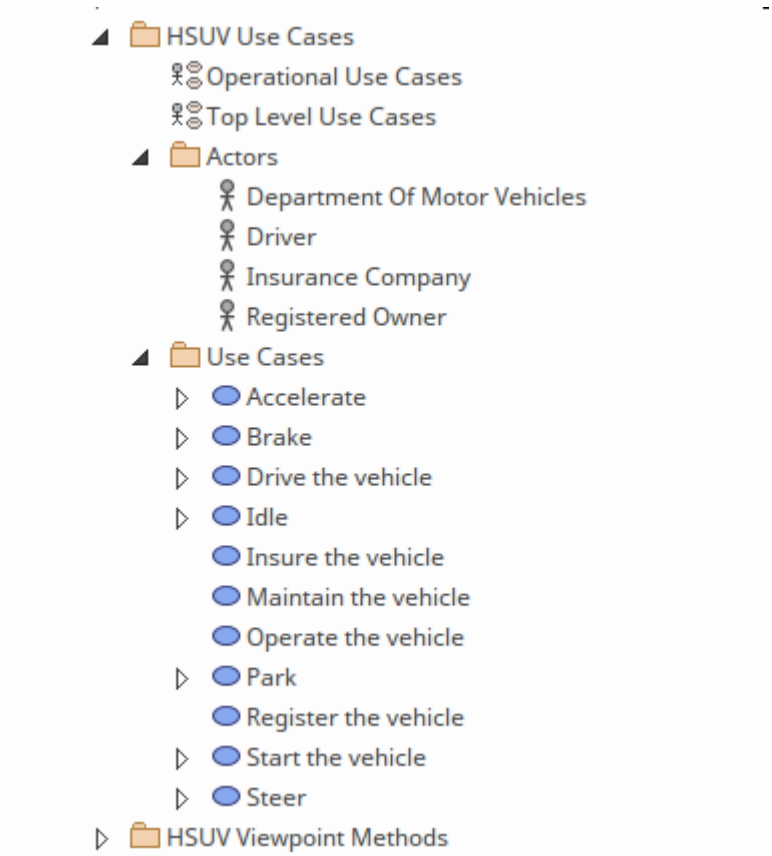
使用此特征的主要优点之一是 RAS 将资产保存在依赖树中，允许资产的潜在用户了解它所依赖的包。这是软件安装程序使用的类似机制，用于确定选择安装的软件项目是否依赖于目标机器上不存在的其他项目，以及这些项目是否具有其他依赖关系。可重用资产服务为工程师完成这项工作并执行依赖图的遍历，使用户能够了解所需资产依赖于什么。此屏幕截图显示了如何在 RAS 中管理依赖关系。



有关详细信息，请参阅Reuseable Asset Service (RAS)帮助主题。

使用包构建存储库

包是 SysML 中最基本和最重要的元素之一。它用作容器，简单地看，它类似于您最喜欢的计算机文件资源管理器软件中的文件夹。因此，通过这种方式，它首先是一个将其他元素（包括其他包）组合在一起的容器，但我们将在本主题中了解到它在Enterprise Architect中还有其他重要功能。

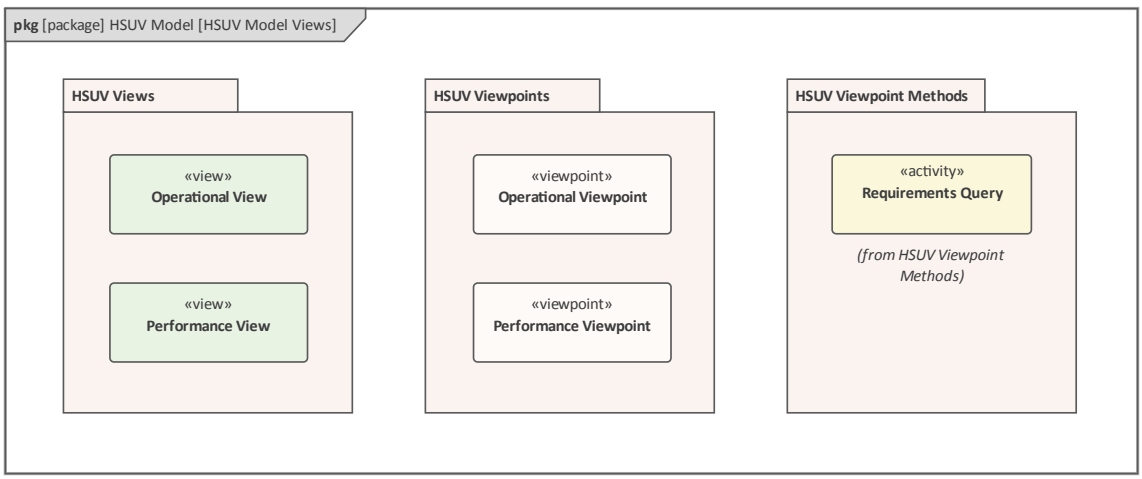


在更深的意义上，包是一个命名空间，它提供了一种唯一标识存储库中任何元素的方法，类似于 URL 的工作方式。此处显示的路径已自动从Enterprise Architect中提取；它允许您可视化命名空间。

SysML示例使用模型.HSUV使用案例.使用案例.驾驶车辆

设置包结构是一项重要且经常令人恐惧的任务，但幸运的是， Enterprise Architect消除了新手（和经验丰富的建模者）在处理此任务时所感受到的许多痛苦。

接下来的几个主题将向您介绍设置包结构的最佳实践，以及一些额外的工具和功能，这些工具和功能将使使用包变得更加容易。



函数的包

除了前面的讨论之外，**包是使用**Enterprise Architect的一个重要元素，因为它们被用作工具中许多功能的基础，**包括**：

- **元素的容器**，
- 命名空间定义，
- **安全**，
- **版本控制**，
- 基线
- 进口和出口，
- 文档，
- 审计，
- 时间感知建模等等。

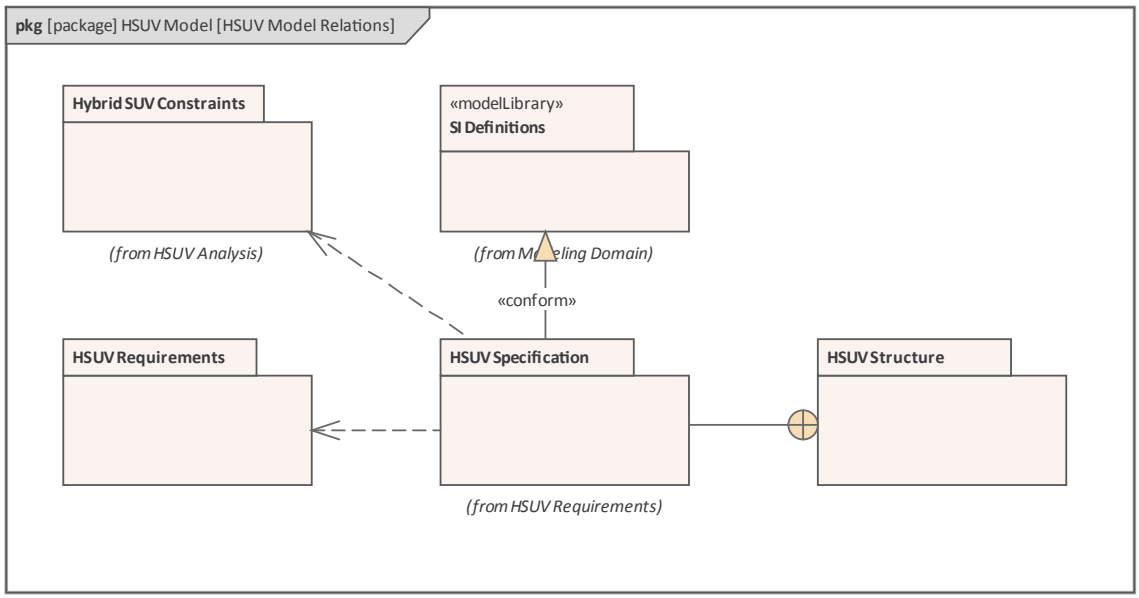
在决定包的结构时，这些事情都必须考虑。集装箱船和命名空间是最关键的，但在决定初始模型结构或重构模型结构时，必须牢记所有其他功能。通常情况下，某些功能最初并未使用，仅在存储库获得一定程度的成熟时才发挥作用。这通常是重新组织存储库的触发因素，但幸运的是 - **由于易于拖放** - 这可以轻松有效地完成，并且不是耗时的练习。

介绍包图表

包图是一个简单的图表，它直观地描述了存储库的结构，包括包与其他包和元素之间的关系。包图看起来非常简单，元素数量很少：

- 模型
- 团队图书馆
- 包
- 视图
- 视图
- 利益相关者

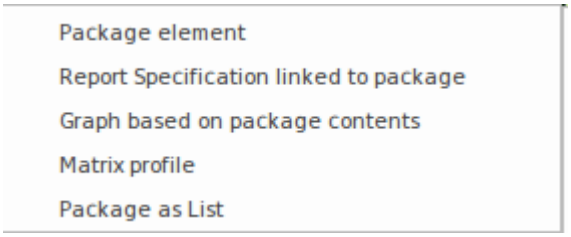
这些是由一系列关系连接起来的。



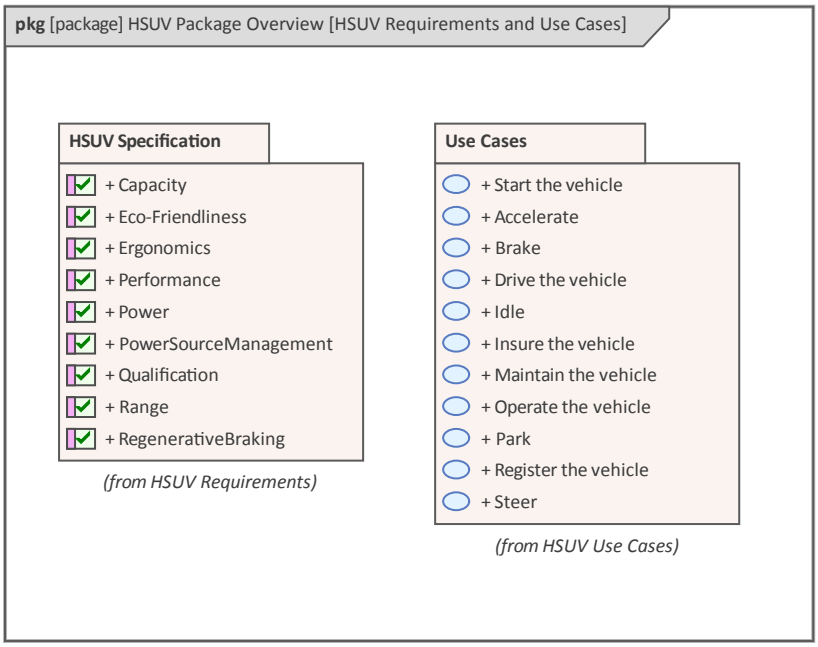
同样，关系的数量非常有限，但每个关系在图中都有特定的含义。

- 符合
- 依赖
- 导入
- 遏制
- 实现
- 细化
- 暴露

与所有 SysML 元素一样，元素既有图形方面也有文本方面，可以在每个包和关系中添加注记，以阐明元素或连接器的用途。包图可以包含任何类型的模型元素，但通常它包含包。 Enterprise Architect扩展了 SysML 规范，提供了许多不同的和创新的方法来可视化包及其在包图中的内容。这些选项可以在将包从浏览器窗口拖到图表上时显示的菜单中看到。

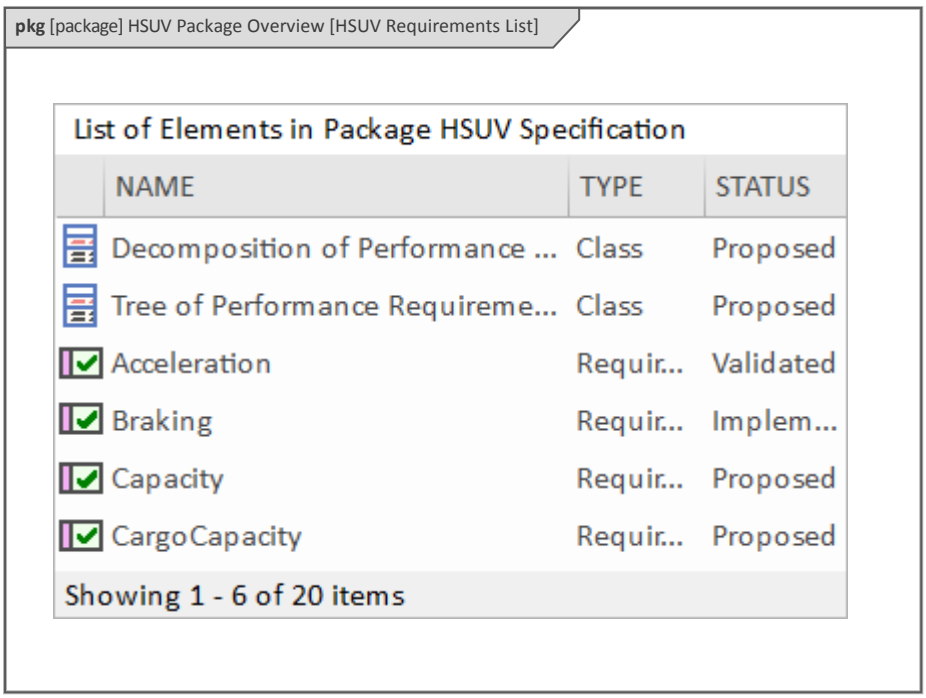


在此图中，我们可以看到建模者选择了 包元素"选项，并设置了包图object的 Compartment Visibility 以显示包内容。 Compartment Visibility 选项可从图表object的任何元素的上下文菜单中获得， Enterprise Architect根据元素类型和可用的隔间动态更改选项。



在下图中，工程师选择了 包作为列表"选项，该选项依赖于自动生成但用户可配置#

基于任何包元素的元数据创建元素的动态列表的代码。在这里，我们看到了相同的需求列表，但这次还显示了许多属性。



可以创建任意数量的包图来定义或帮助可视化存储库的结构。更多信息请参见Package Diagram帮助主题。

创建图表包

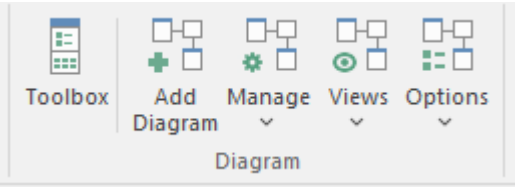
可以从用户接口中的多个位置创建A包图，方法是选择：

- 设计ribbon - 在图表面板上添加图标
- 浏览器窗口工具栏 -新的图表图标
- 浏览器window c *ontext menu* - 添加图表

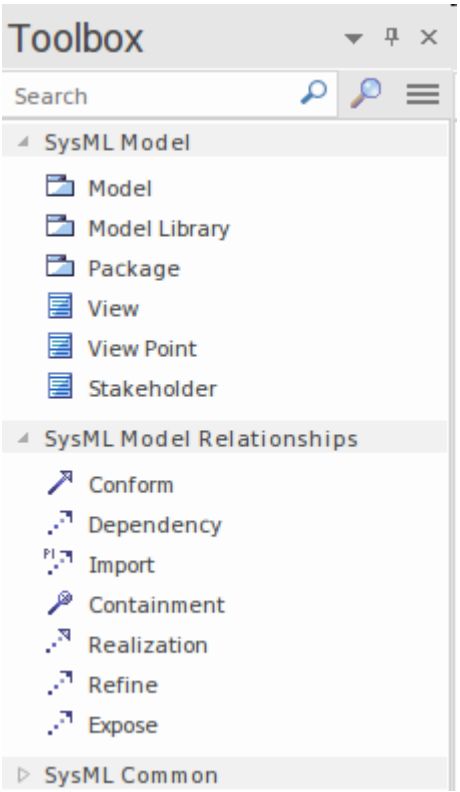
访问选项都将显示相同的对话框，它们只是相同工具特征的不同入口点。我们将使用设计功能区创建一个包图。

首先在浏览器窗口中选择您希望包图所在的位置。这可以是包或元素，但通常将包图插入包中。在浏览器窗口中选择包位置后，选择功能区选项：

设计>图表> 添加图表



选择此选项将打开*New*图表对话框，您可以在其中命名图表；该名称最初默认为包含图表的包或元素的名称。选择 SysML 透视图并选择 SysML 的版本后，将显示一个图表列表，您可以从中选择包图表。单击确定按钮在浏览器窗口中选择的位置创建一个新的包图。将打开图表视图，允许您开始添加描述系统结构的元素和连接器，并将其划分为这些结构组。 Enterprise Architect还将显示包含由 SysML 规范定义的元素和关系的 包"工具箱页面，这些元素和关系适用于构建包图。如果需要，除了始终可用的 公共"元素和 公共关系"页面之外，还可以打开任意数量的其他工具箱页面。



与包图一起使用的最重要的元素和连接器是：

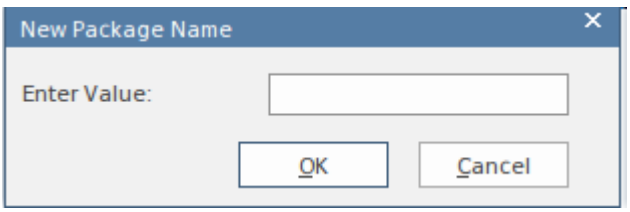
元素

- **模型**- 用于定义系统的高级部分
- 团队图书馆- 用于定义一组可重用的元素
- **包**- 用于创建基本结构单元
- 视图- 用于定义利益利益相关者在查看时会看到什么
- 视点 - 用于定义视图的参考点
- 利益利益相关者——用于描述有物质利益的人或当事人

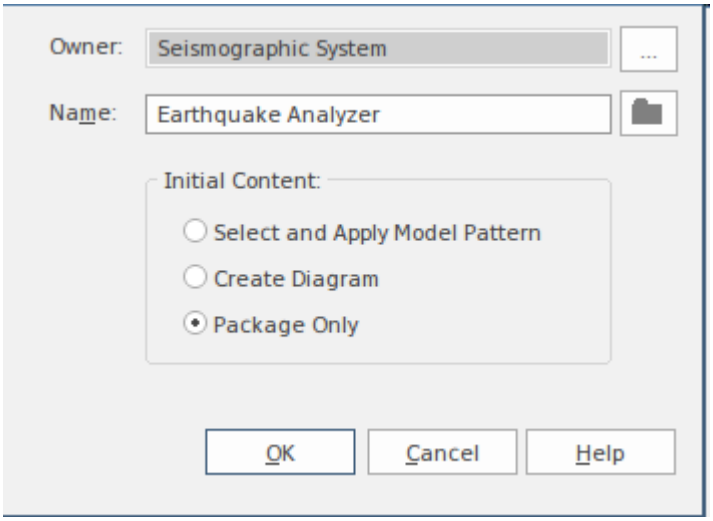
连接器

- **符合**
- 依赖
- 导入
- **遏制**
- 实现
- 细化
- **暴露**

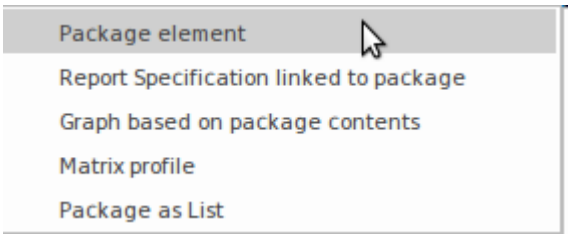
可以通过将元素从工具箱到图表视图视图中来将元素添加到图表中。从定义模型和团队图书馆开始被认为是一种很好的做法。当模型或团队图书馆图标从工具箱拖到图表时，将提示建模者使用此对话框输入新的包名称：



输入包名称后，将在图表上创建一个新的包元素。当从工具箱创建一个包或将一个包插入浏览器窗口时，会为建模者提供许多选项，如此屏幕截图中的对话框所示。



如前所述，当从浏览器窗口拖动现有包（或从现有图表复制）时， Enterprise Architect允许您以多种不同的方式创建它，如下面的屏幕截图所示：



包组织制度

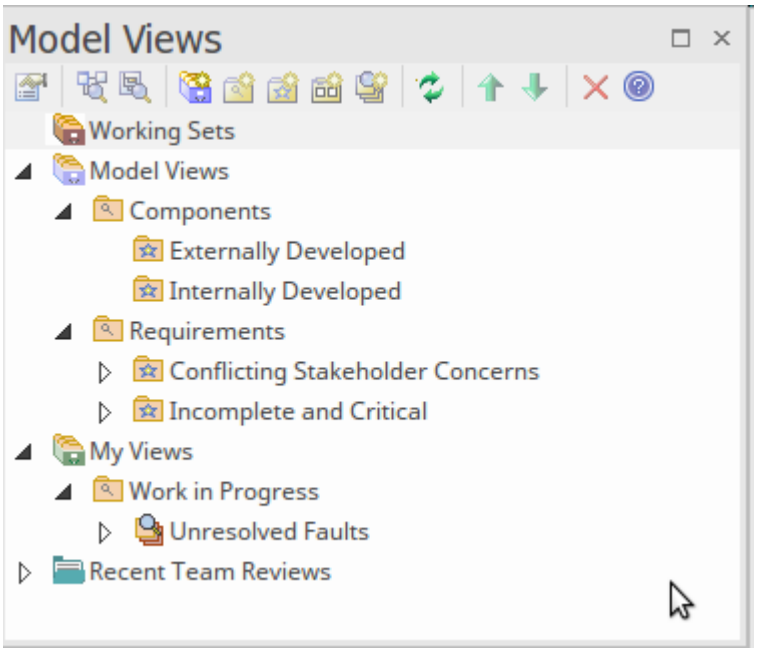
如前所述，参与建立存储库的图书馆员、管理员或工程师可能会发现自己在选择哪个方向上存在冲突，因为有广泛的组织原则可用于构建存储库的内容。其中一些是：

- 分解结构A系统 |子系统 |组件 |部分
- 工程团队，致力于系统的不同方面 - 团队一 |二队
- 工作计划和项目 - 计划一 |项目一、项目二
- 方法内的划分-架构|需求|设计
- 安全和访问控件
- 易于导航

这些原则中的任何一个或任何组合都可以用于构建存储库，并且可以随着时间的推移进行更改，以适应工程实践的演变以及用户的模型使用和体验。这些原则中最困难的可能是需要使存储库对其居民友好，以确保易于导航，以便他们可以轻松找到他们正在寻找的东西。 Enterprise Architect有一些有用的功能可以减少这种紧张，允许使用其他机制进行导航，并释放存储库设计以基于许多其他更重要的原则进行开发。此处列出了其中一些工具特征。

视图模型

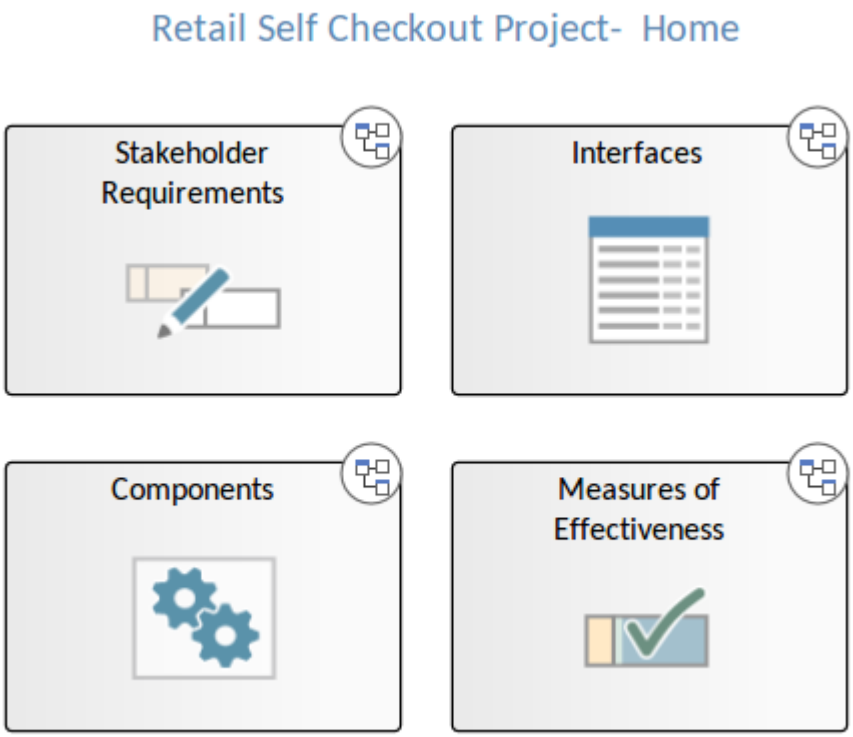
提供灵活有效的机制，允许工程师或团队创建他们认为有用的模型的任何视图。使用此功能，建模人员和工程师无需访问浏览器窗口，因为他们可以通过模型视图窗口定位感兴趣的元素。



例如，可以基于从存储库的任何部分返回元素的搜索来创建视图；工程师可以定义一个视图，该视图返回所有具有高优先级的需求，状态为 已批准”并标记为 困难”，无论它们属于哪个项目或它们位于包层次结构中的什么位置。或者，建模者可能只是挑选对它们重要的特定元素和图表，并将它们包含在收藏夹视图中，或者基于新创建的组件创建视图。该功能为访问存储库的重要部分提供了一种高度灵活的机制，并且可以在建模者或团队级别创建视图。我们将在后面的主题中回到模型视图功能，因为它是该工具非常有用的部分。

图表导航单元格

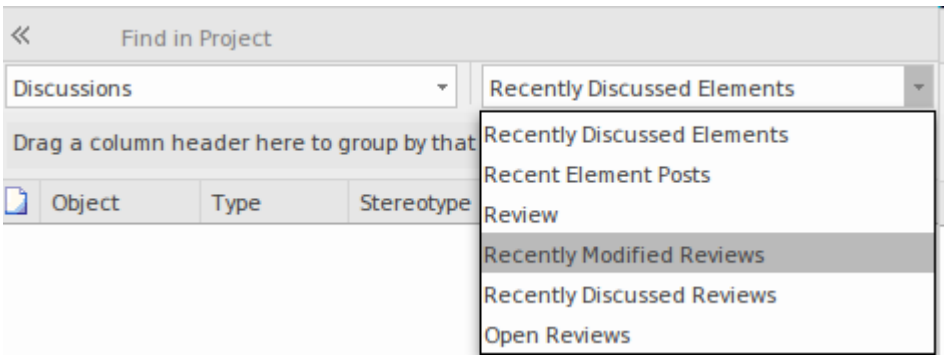
Enterprise Architect通过提供图表机制来超链接到存储库中的任何图表，使用户可以轻松地在存储库中导航。



这允许图书馆员甚至建模师自己创建任意数量的图表作为启动板，将查看者带到感兴趣的图表，有效地使他们无需了解存储库的结构。这些图表可通过 Internet 浏览器和云产品查看，并为临时用户和非建模者提供引人入胜的体验。

搜索功能

这是一个强大的特征，它提供内置和用户定义的搜索来检索满足一组指定标准的元素或图表列表。随着越来越多的人为模型做出贡献，并且信息从外部来源（如风险、政策、规则、原则等）导入，存储库中包含的信息量会呈指数级增长。有一组丰富而有用的搜索被定义为产品的一部分，在许多情况下，这些内置搜索之一足以让建模者或工程师找到他们正在寻找的元素或图表。这些搜索可以是参数驱动的，提供一种重用搜索来查找各种元素的机制。例如，可以编写具有允许用户输入状态的用户输入参数的状态，例如在搜索运行时 建议”。

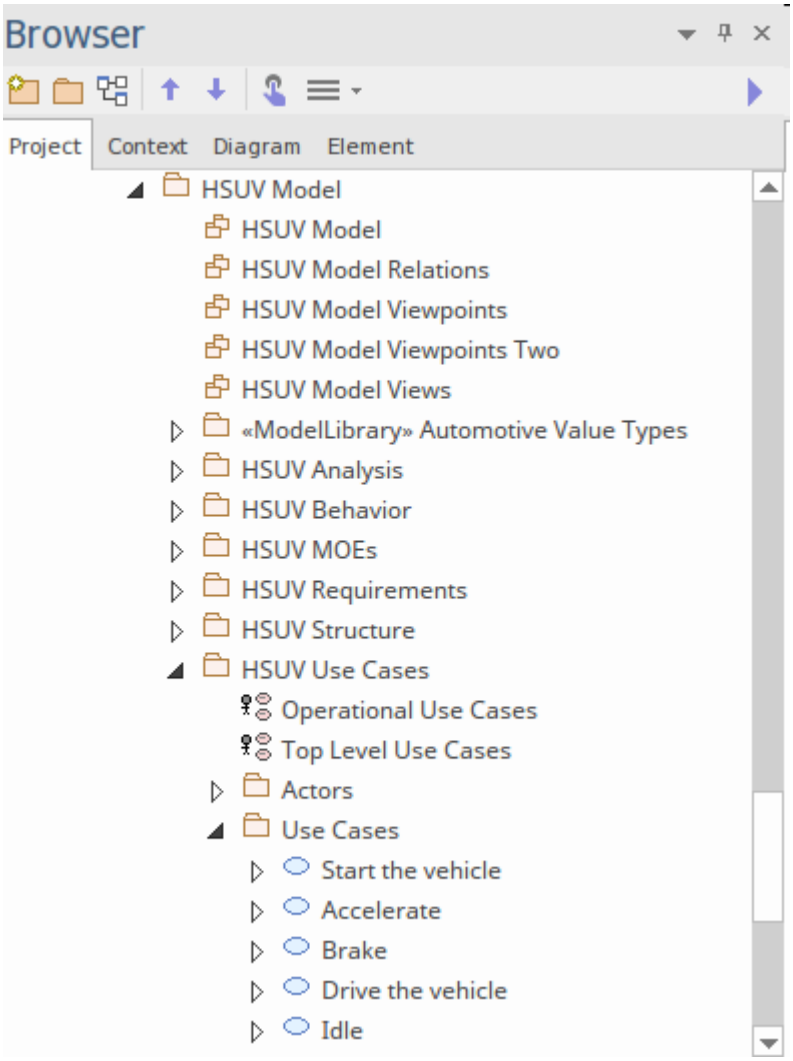


非技术人员可以使用直观的查询生成器创建搜索，但也有许多其他方法可以创建搜索，包括#

需要数据库库表知识的基于查询的查询和需要技术人员创建定义搜索的程序的加载项查询。这些搜索可以由许多其他功能使用，包括如前所述的模型视图。

浏览器窗口

浏览器窗口是使用展开和折叠树节点构建和导航存储库的主要工具。关键的结构元素是包，它是一个类似文件夹的元素，可以包含其他元素和图表，包括其他包。这些元素又可以包含其他元素、特征和图表，但不能包含包。

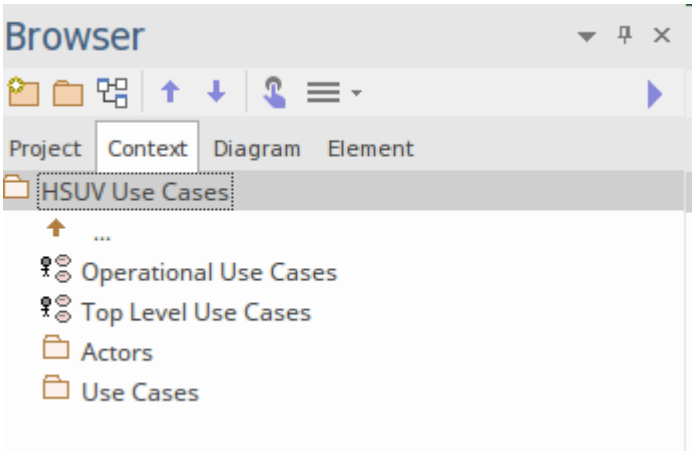


根是树中最高的节点；这些根包可以包含视图，而视图又可以包含任何级别的包和元素。包括包、元素、特征和图表的树节点可以在位置之间复制和粘贴，或拖放到新位置。包层级应用了很多重要的模型导入导出和包控件，包括文档、窗口、文档等基线。有关详细信息，请参阅The Browser Window主题。

上下文图表和元素浏览器

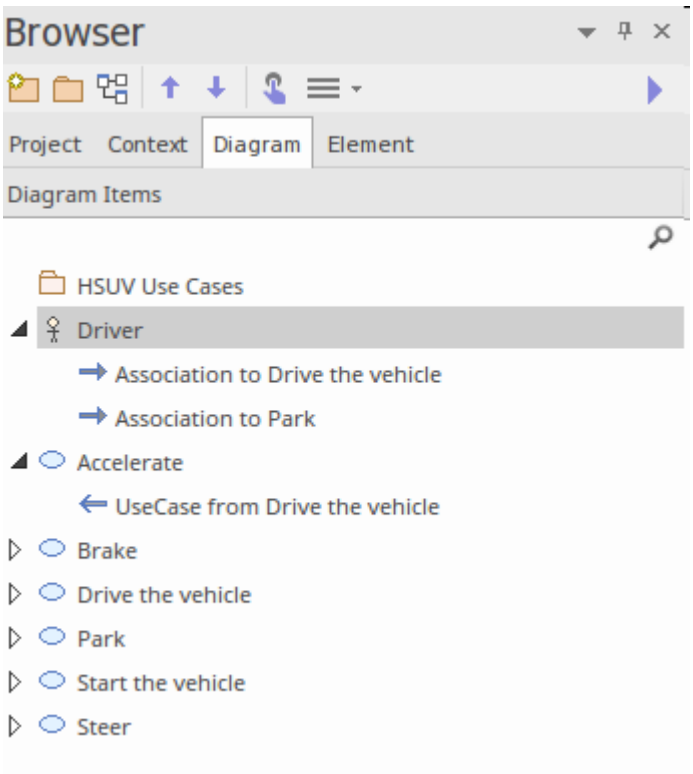
Enterprise Architect提供了许多额外的浏览器，可帮助工程师或建模师聚焦存储库内容的子集。可以从主浏览器窗口中选择这些浏览器作为选项卡。

上下文浏览器仅提供模型特定分支的过滤视图，以仅在特定时间相关的存储库部分上工作。这个聚光灯视图消除了项目视图的噪音和复杂性，只显示了感兴趣的模型的一部分，允许工程师单独查看它。

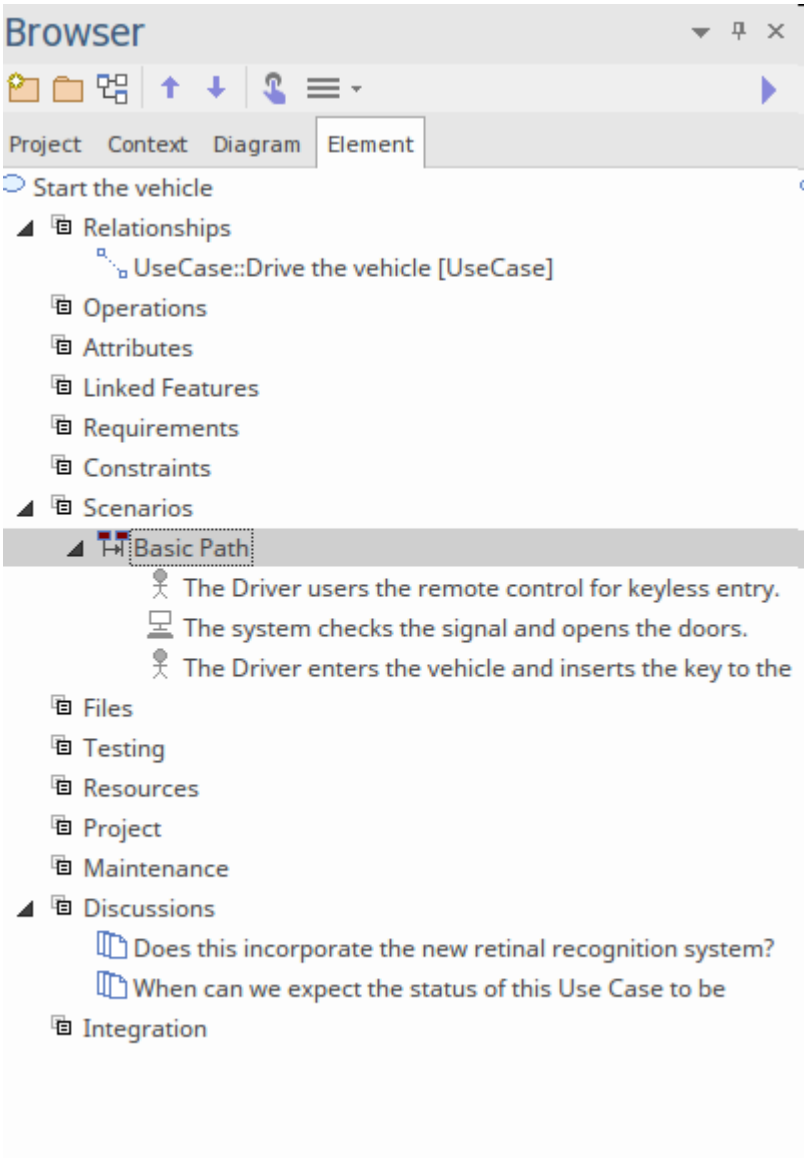


在我们的示例中，建模者希望将聚焦集中在用例包上。使用 ... 按钮，您可以向后移动树，或者通过单击包含其他元素的包或元素，这些元素可以显示，但视图始终保持在单一层次结构级别。

图形图表浏览器列出了活动图表上的对象。还可以显示每个object的连接器，使其成为图表的宝贵视图。

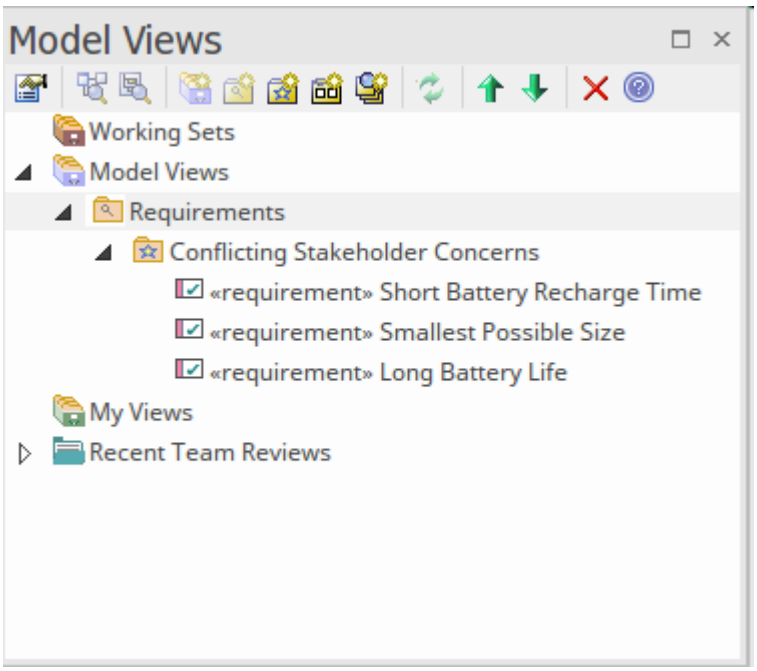


元素浏览器显示有关当前选定元素的信息。这提供了一种可视化所有元素关系、场景、需求、特征等的方法。


Browser
Project
Context
Diagram
Element
Start the vehicle
Relationships
UseCase::Drive the vehicle [UseCase]
Operations
Attributes
Linked Features
Requirements
Constraints
Scenarios
Basic Path
The Driver users the remote control for keyless entry.
The system checks the signal and opens the doors.
The Driver enters the vehicle and inserts the key to the
Files
Testing
Resources
Project
Maintenance
Discussions
Does this incorporate the new retinal recognition system?
When can we expect the status of this Use Case to be
Integration

使用模型视图访问存储库

模型视图提供存储库中元素的替代视图。浏览器窗口旨在以结构方式组织包和元素，而模型视图功能允许建模者创建多个视图，这些视图可以对元素和图表进行不同的分组。这是一个方便的功能，个人和工程团队可以使用它在任意数量的专有视图中查看存储库内容，这些视图旨在仅显示重要或相关的信息。它有效地允许您创建**窗口**，通过这些窗口以独特且引人注目的方式查看存储库，这将提供洞察力和清晰度，允许建模者看到使用浏览器窗口可能无法看到的东西。



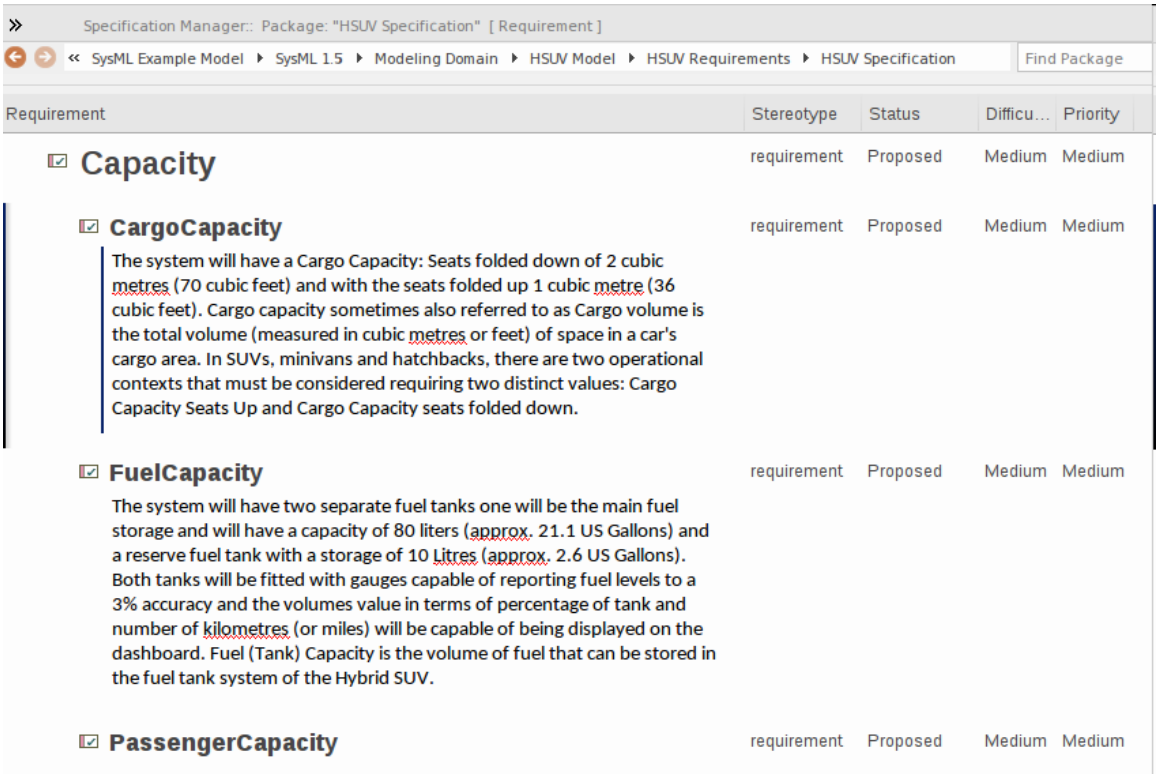
视图可以基于广泛的标准，包括包含手工挑选的感兴趣项目的收藏夹文件夹，以及基于搜索的文件夹，例如"上周创建的所有具有已提议状态的元素"或 所有组件由特定的工程供应商"。有关更多信息，请参阅Model Views帮助话题。

需求定义与管理

需求工程领域是解决方案开发生命周期中最关键的学科之一，它对项目的成功有记录的影响。用 20 世纪著名物理学家阿尔伯特·爱因斯坦的话来说：

如果你足够仔细地定义一个问题，那么解决方案就会突然出现在你身上。”

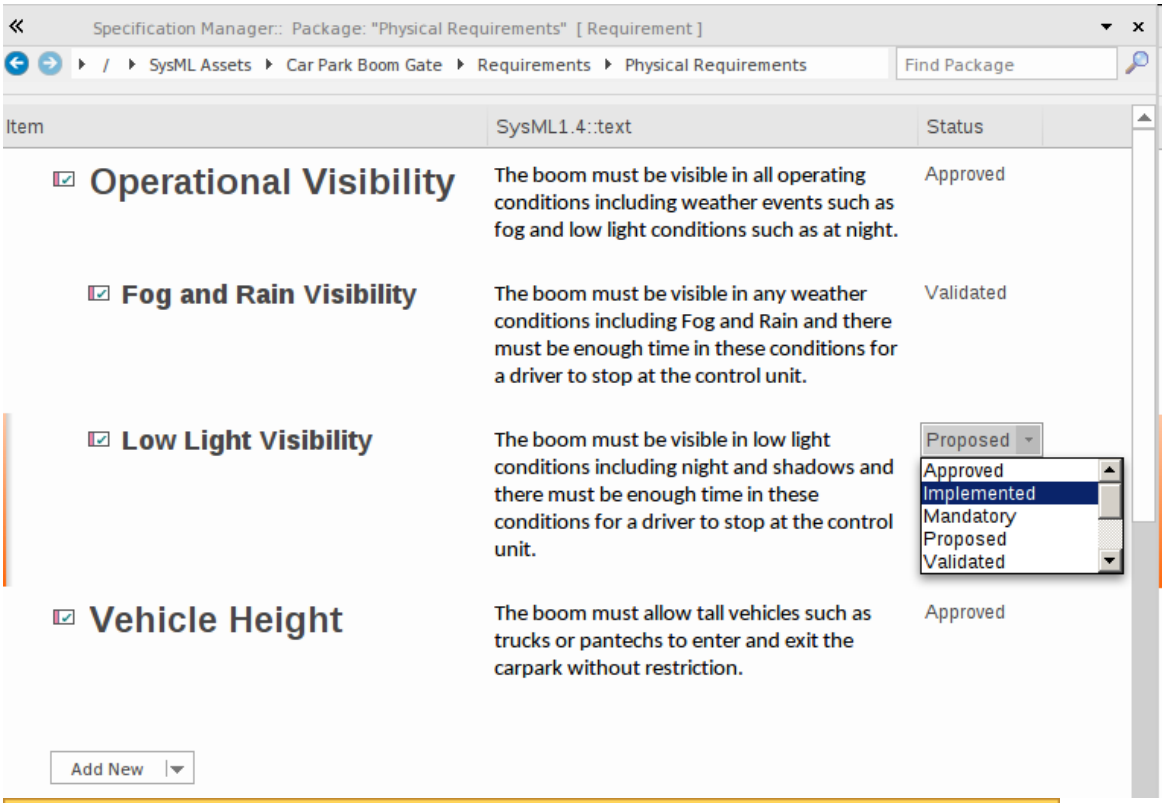
Enterprise Architect拥有一系列无与伦比的工具，用于开发、管理、可视化和记录需求，包括用于导入或集成并与外部需求管理系统同步的工具。

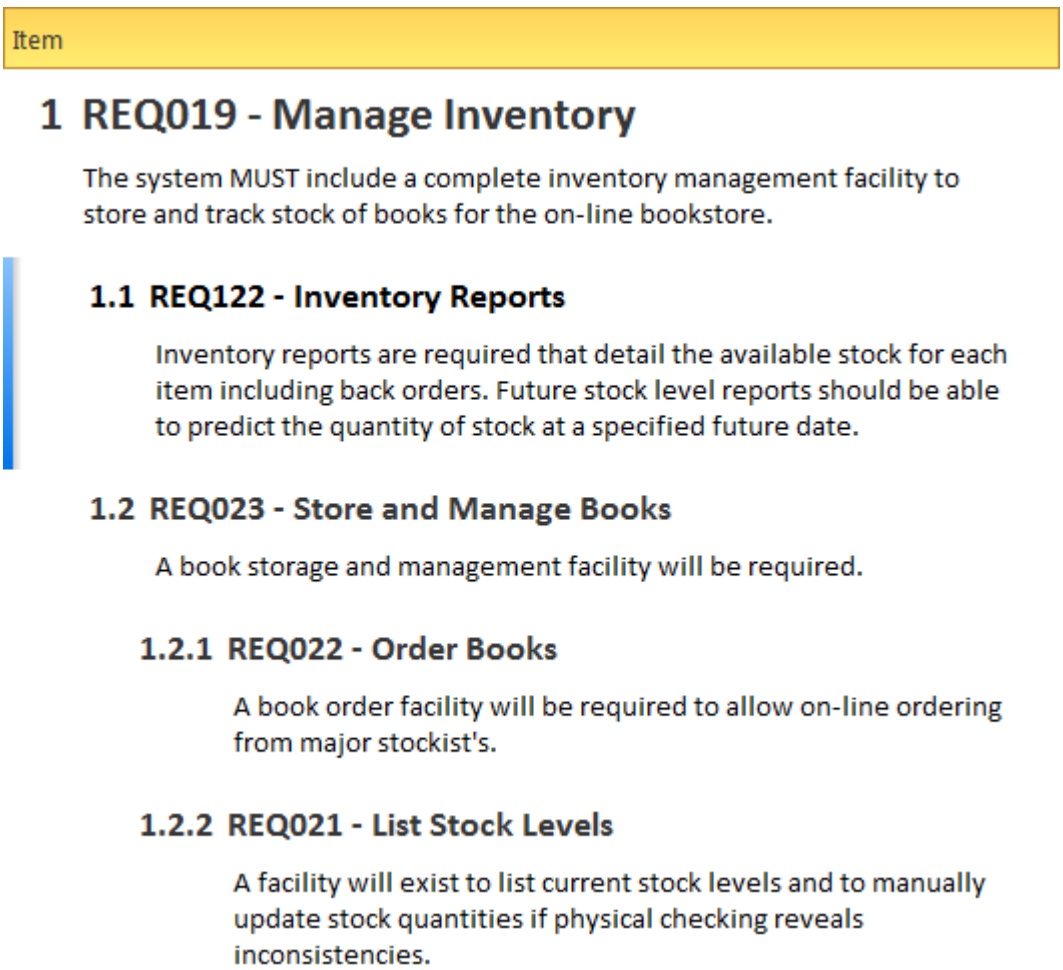


这些工具实现了 SysML 规范中定义的需求的所有方面，但工具特征远远超出了这一点，它创建了一个复杂的需求平台，其中包含与需求管理和定义相关的所有学科的工具。这些工具不仅对那些直接工作的工程师或管理人员有用，而且还有一个功能范围，例如可以帮助任何学科的设计可追溯性窗口，并且可以由架构和团队使用以确保将要求纳入设计，并因此实施到交付的产品或服务中。有关更多信息，请参阅The Requirements Model帮助主题。

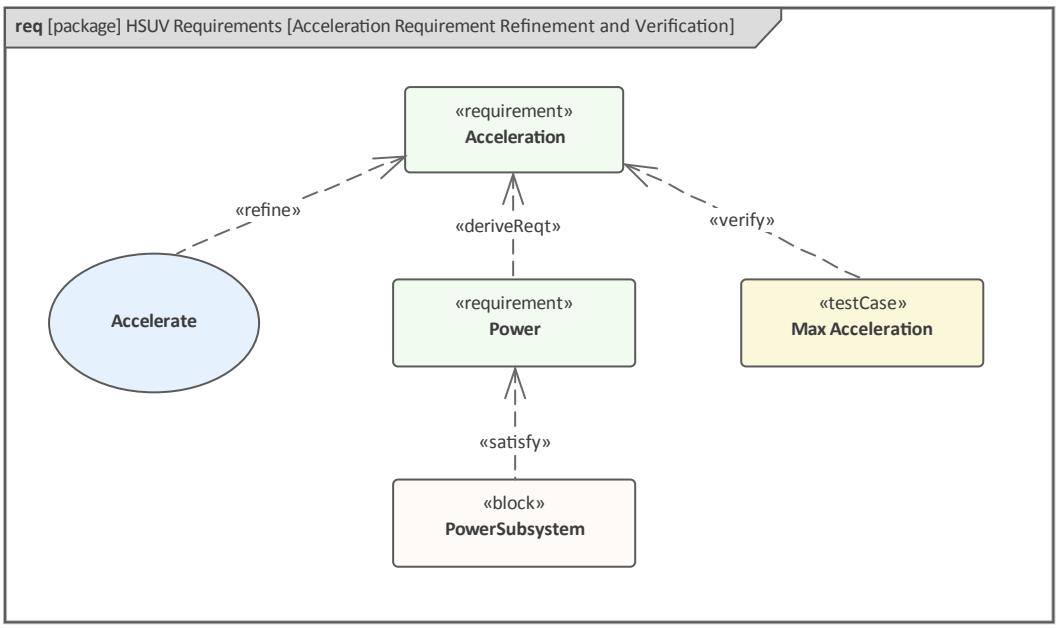
开发需求

需求开发包括与发现、评估、记录、记录和验证特定项目的需求相关的所有活动和任务。需求被发现、分析、指定和验证。 Enterprise Architect拥有广泛的工具和特征来协助系统工程师开发需求。需求开发的核心是规范管理器，工程师可以通过它以文本形式输入、查看和管理需求，就像在电子表格或文档中一样。需求属性，例如状态、优先级和作者，可以在线编辑，并且可以应用过滤器来限制特定要求的显示。



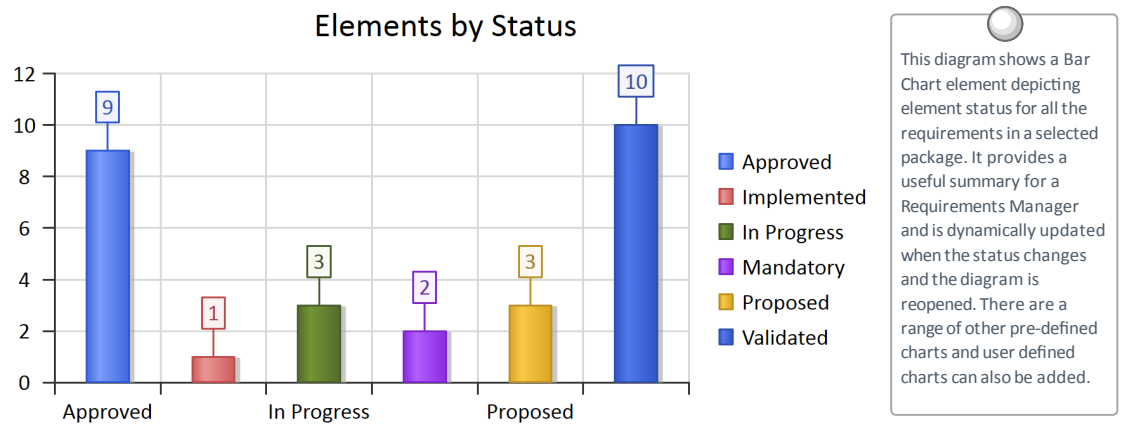


规范管理器可与图表、可追溯性功能窗口、讨论区等其他管理工具平台配合使用。



管理需求

这包括维护一组要求的活动，这些要求代表项目团队和客户之间的协议或协议。它还聚焦确保设计和开发团队可以接受需求，并且它们足够具体，可以实施到工作业务、软件或硬件系统中。 Enterprise Architect是一个复杂的需求管理平台，无论领域、项目规模或所遵循的方法如何， Enterprise Architect都提供了可以轻松管理复杂项目中最大的需求存储库的工具。



需求关系

有一组丰富的需求关系允许需求元素连接到其他建模元素，包括其他需求。关系包括：

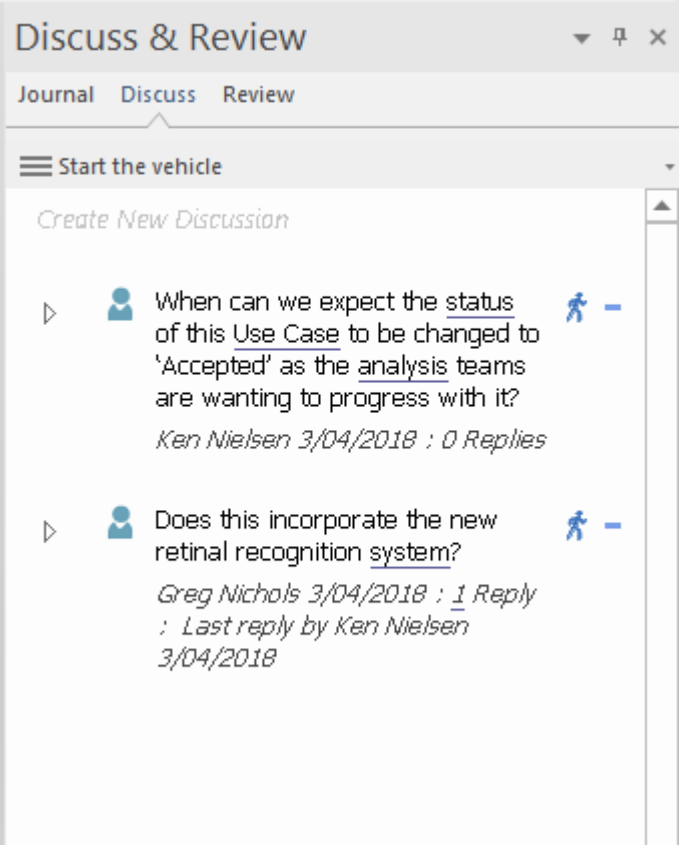
- 遏制
- 跟踪
- 复制
- 派生

- 核实
- 细化
- 满足

我们将在本主题的正文中探讨这些关系。

可视化需求

需求的可视化是需求过程的一个重要方面，因为在需求目录被指定、分析、开发和管理时，所有利益相关者都可以查看需求目录是至关重要的。这些要求代表了工程师对利益相关者就手头的问题或机会所做的讨论、观察和表述的解释。 Enterprise Architect具有广泛的机制，不仅可以向利益相关者社区提出这些要求，而且还允许对这些要求进行讨论、审查和策划。



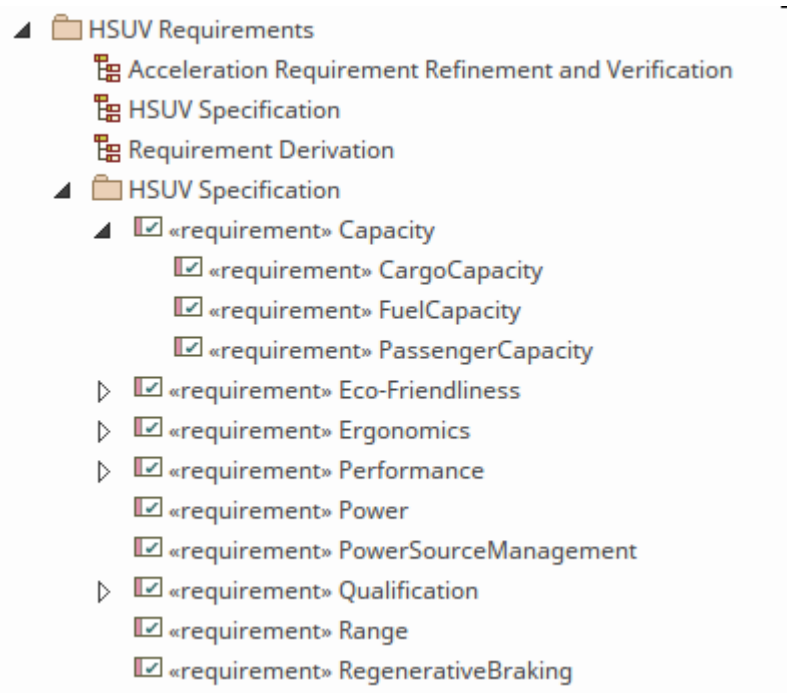
记录需求

作为需求工程学科的一部分，通常会生成许多文档，例如系统需求规范和用例报告，这些文档可以使用内置模板或用户定义的模板从需求模型中自动生成。此外，可以使用内置或自定义模板生成范围广泛的其他文档。

还可以在手机、平板电脑或 PC 等便携式设备上的网络浏览器中查看模型。此功能作为专业云服务器产品的一部分提供，并提供了生成静态文档的替代方案，并允许工程团队与建模环境之外的更多受众进行交流和协作，而无需任何软件安装或配置。

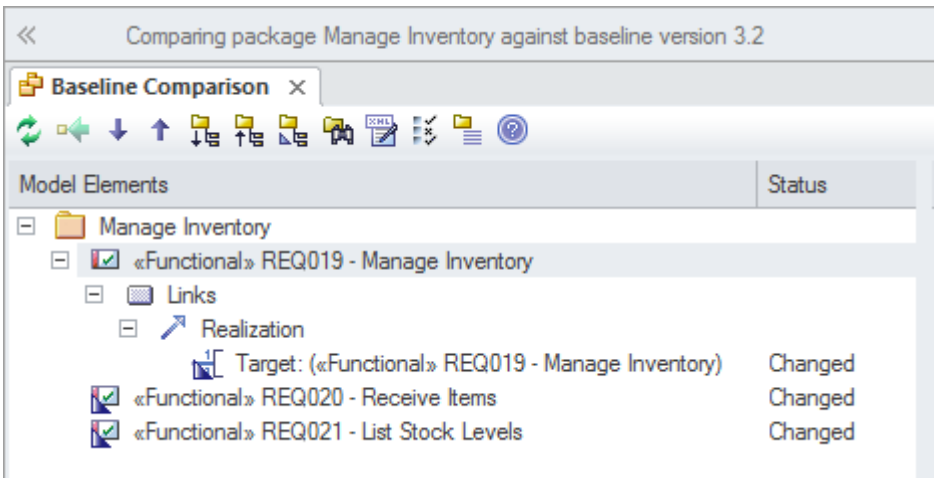
作为第一类公民的需求

Enterprise Architect提供了广泛的功能，可用于需求作为一流元素的开发、可视化、管理和文档化。阅读一般 SysML 教科书的人通常会想到用图表表达需求，但是Enterprise Architect提供了广泛的其他方法来可视化需求，这将有助于工程师将它们作为基于文本的元素使用，包括能够在浏览器窗口中以层次结构将它们可视化。



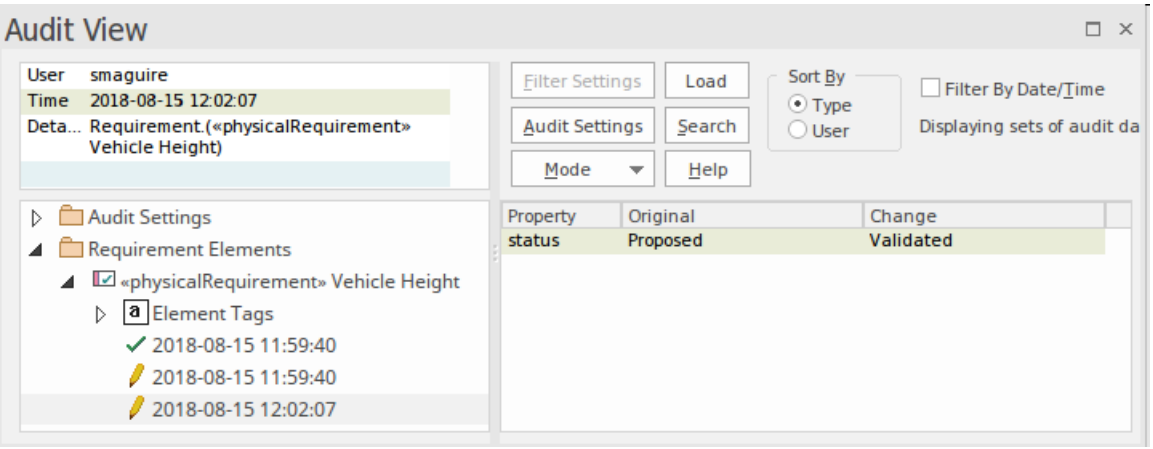
需求可以作为规范或标书的一部分创建，也可以作为合同文件的一部分，在这种情况下，它们可以很容易地导入Enterprise Architect 。然而，更常见的是，它们是作为通常通过研讨会和审查进行的启发工作的一部分来开发的。 Enterprise Architect有许多特征可用于记录这些会议的结果，例如思维导图。研讨会完成后，这些会议中记录的想法可以转换为需求或映射到会议元素，以允许协作开发。

需求通常是组织之间合同关系的一部分，或者是同一组织不同部门之间协议的一部分，因此需要严格维护和管理。 Enterprise Architect提供了多种功能来协助这种严格的审计工具，包括基线的审计工具、版本控制等等。



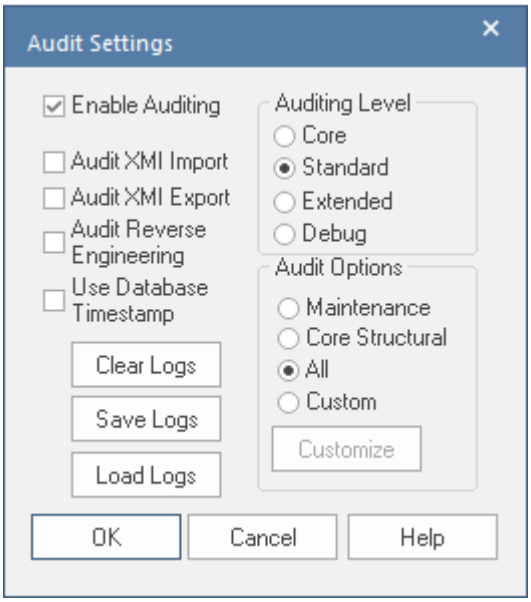
审计需求

审计可以在模型中打开，并可以跟踪需求更改的详细信息，包括更改时间、更改者以及更改前后的审计。可用于跟踪模型中的更改，谁更改了它以及何时更改。有多种模式，存储库管理员可以使用这些设置来指定审计中记录的内容。而一个基线可以用来显示一个模型和一个时间点之间的差异，记录每个单独的审计工具它不能用于以前的状态（但是，如果需要恢复到该工具，基线使用该工具）。



这是一个特别有用的特征系统工程，其中存在对流程的监管或合规性方面，或者当必须将故障追溯到其设计或需求规范时。审计通常由图书馆员或行政函数内的管理部门设置和管理可以使用功能区选项 设置>模型>审计 "启用、设置和查看审计。

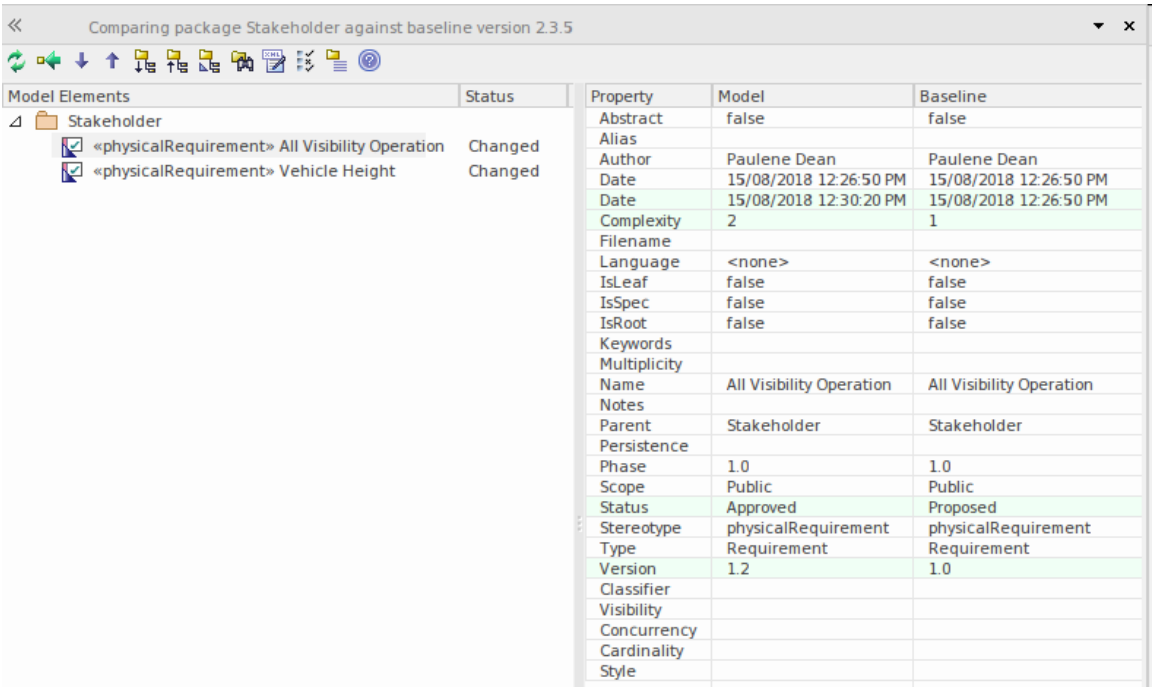
审计默认是禁用的，必须在系统开始保存审计log之前启用（打开）。这以及一系列其他选项可从 "审核设置"窗口中获得。



更多信息请参见Auditing帮助主题。

需求基线

基线用户在模型中启动的包的快照。基线地复制包的层次结构及其内容的分支。在随后的时间点，可以与模型的基线、模型和模型进行比较，如果模型模型了变化，这些变化将通过一个可视化工具，允许用户查看发生变化的每个部分，包括存在于基线和模型中。然后可以将内容从基线级别注入到离散变化的模型中。



为基线提供方便的方式，确保系统工程团队以正确的方式发展，当需要将模型直接恢复到以前的版本时，可以使用它们来恢复基线模型。通过按 Ctrl+Alt+B 或从功能区位置查看：

功能区：设计包>管理>管理基线>

如前所述，基线库存储在存储库中，如果用户启动并复制了存储库，基线将被复制对于大多数用户来说，创建基线权限是很常见的，但可以将管理员权限恢复为基线或管理员角色。欲了解更多信息，请参阅Baseline Tool帮助。

版本控制

版本控制允许对模型中的包进行版本控制。要开始部分模型的工作，用户需要签出一个包（包括其子包），然后在本地副本上工作。当工作完成或在任何时候用户可以签入包允许其他模型用户看到更改。

版本控制提供强大的运行方式，并且与用户启动复杂的工作模型相比，它不需要其他包的基线。当工作完成并进行更改时，系统会在后台自动创建一个版本。版本控制可以通过这些功能区选项进行设置和查看。

Settings >版本控制> Project-VC, Package-VC

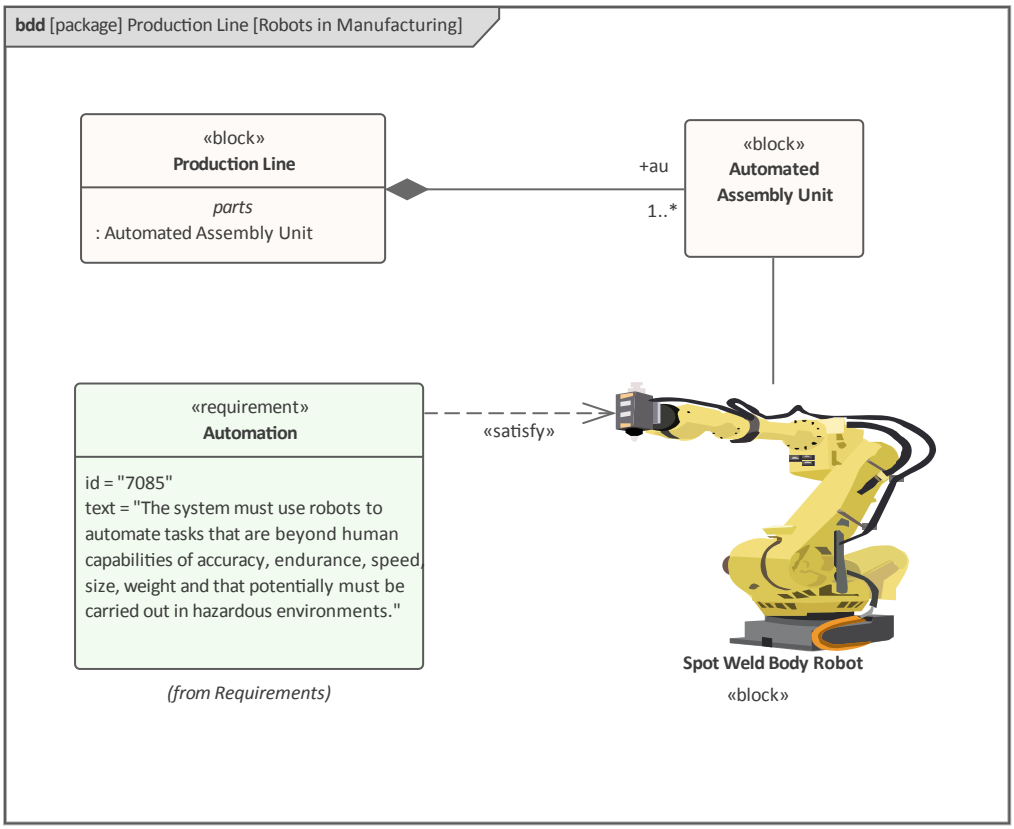
版本控制提供了管理模型内容的有效机制，并允许用户或团队对包及其内容随时间变化的方式保持细粒度的控制。更多信息参见Version Control of Model Data 帮助话题。

需求图表介绍

需求图提供了A可视化需求及其联系的方式。在这些图表上可以看到的不仅仅是任何两个需求之间的关系，还有需求与其他类型元素（例如用例、活动块）之间的关系。工具箱中提供的两个元素是：

- 需求
- 测试用例

这些元素可以相互连接，也可以连接到其他元素，创造出丰富的表达。

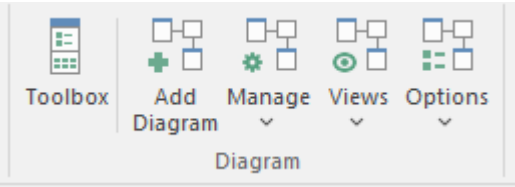


创建需求图表

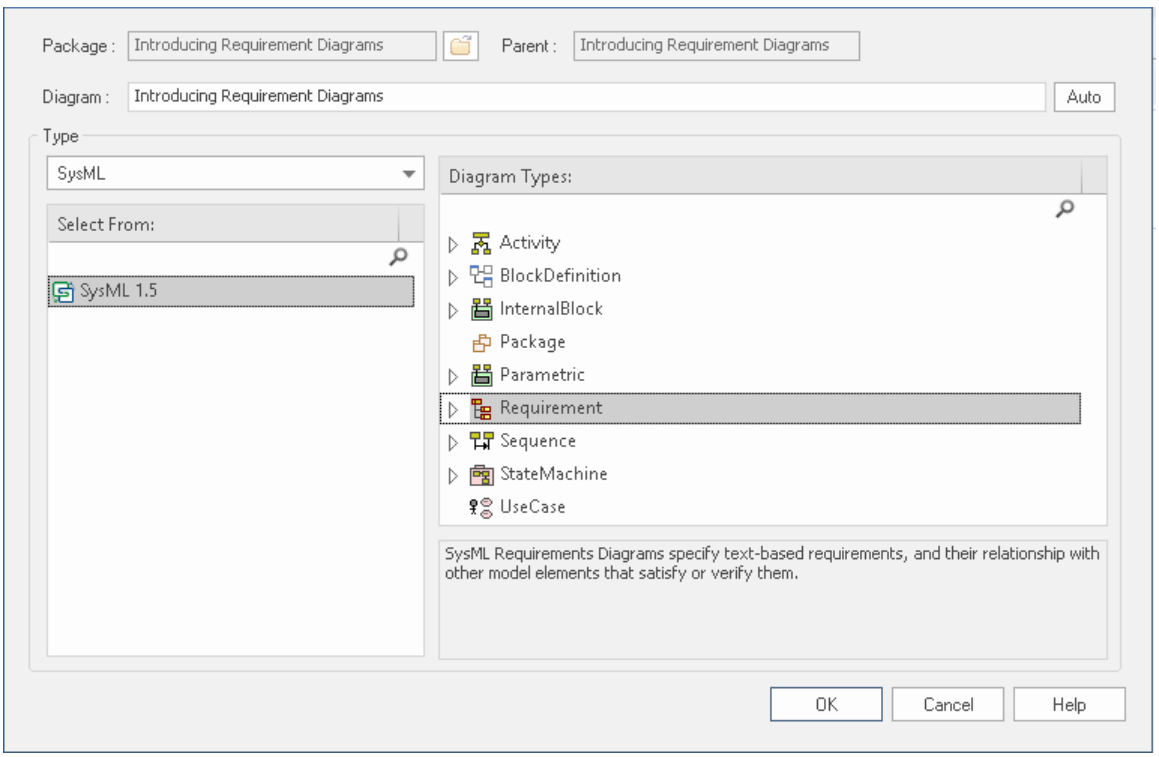
可以从用户接口中的多个位置创建A需求图表，例如：

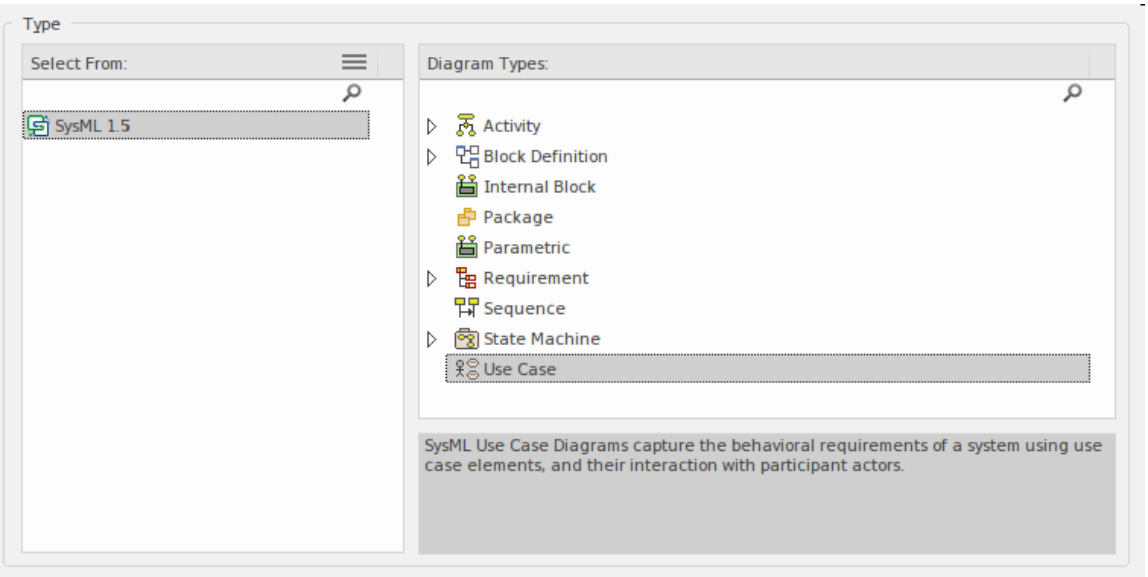
- 设计功能区 - 图形”上的“添加图表”图表 控制板
- 浏览器窗口工具栏 - “新图表”图标
- 浏览器窗口上下文菜单 - 添加图表”选项

我们将使用设计功能区创建需求图。首先，在浏览器窗口中选择您想要创建需求图的位置。与所有图表一样，这可以是包或元素，但通常将需求图表插入包中。在浏览器窗口中选择包位置后，单击功能区选项 设计>图表>添加图表”。



此选项打开 新图表”对话框，允许您重命名图表 - 名称最初默认为包含图表的包或元素的名称。当您选择SysML蓝图和SysML版本时，将显示图表列表，您可以选择需求图表。您点击确定按钮在浏览器窗口的指定位置创建一个新的需求图表。图表视图将打开，允许您开始添加描述需求及其关系的元素和连接器。 Enterprise Architect还将显示工具箱的 需求”页面，其中包含 SysML 规范定义的元素和关系，以适用于构建需求图。除了默认显示的 公共”元素和 公共关系”工具箱页面外，还可以根据需要打开任意数量的其他工具箱页面。





需求图使用的最重要的元素和连接器是：

元素

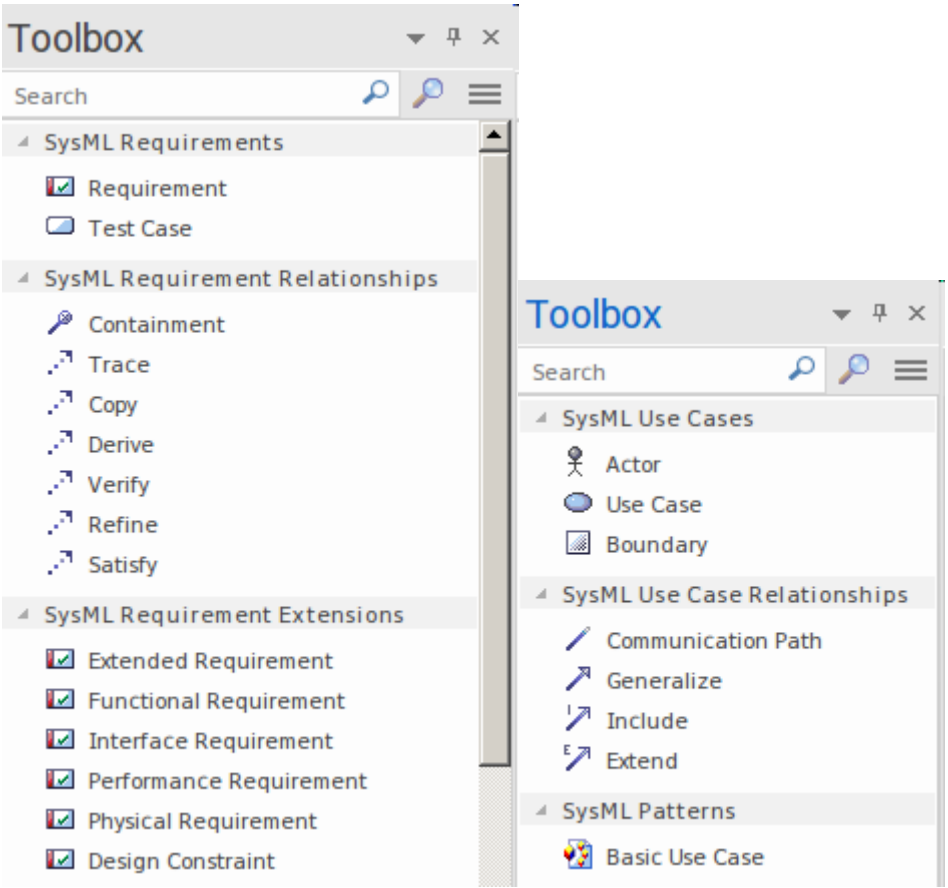
- -需求- 用于定义需求
- 测试用例——用于描述一个测试

连接器

- 遏制 - 用于提供有助于澄清需求的附加信息
- -跟踪- 用于将需求连接到任何其他建模元素
- Copy - 用于表明一个需求是另一个需求的副本
- 派生 - 用于描述一个需求是基于另一个需求的扩展或派生的事实
- 验证 - 用于表示需求已得到满足
- Refine - 用于添加有助于澄清需求的细化或附加信息
- 满足 - 用于表明架构或设计中的一个或多个模型元素满足需求中表达的概念

需求扩展

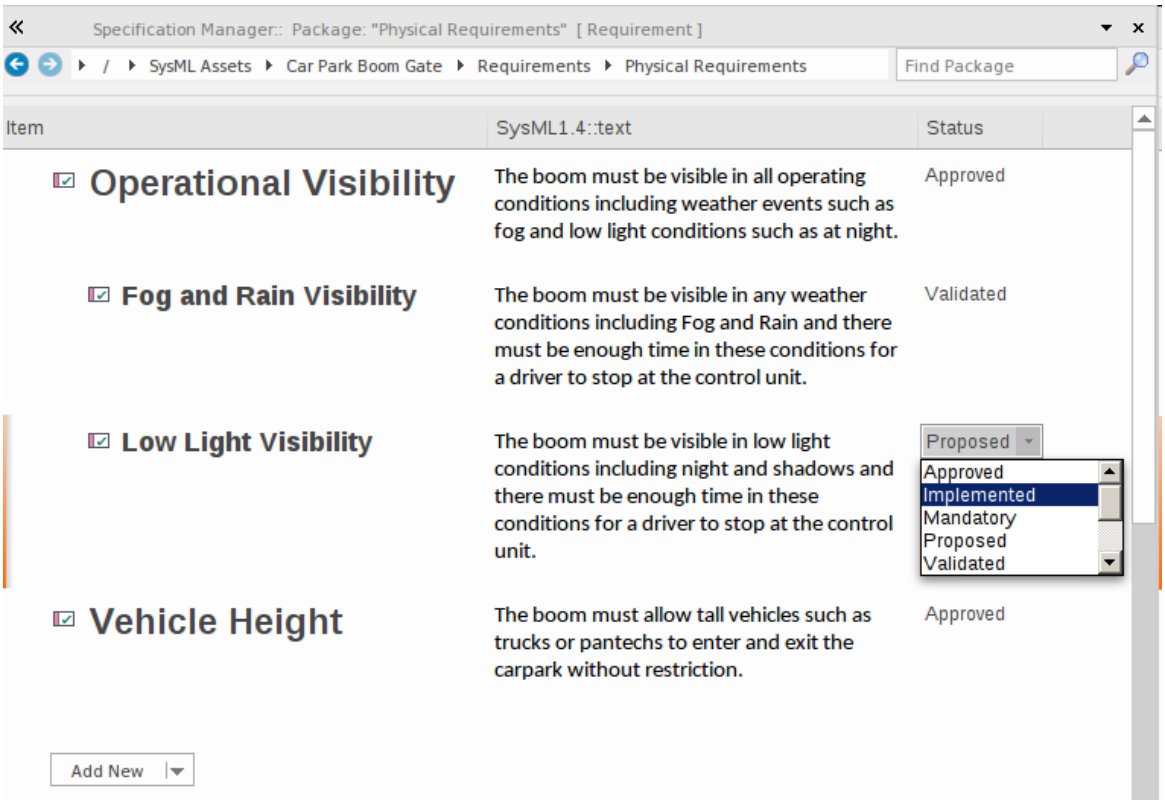
- 扩展需求- 用于扩展需求
- 功能需求- 用于与函数相关的需求
- 接口需求——用于接口相关的需求
- 绩效需求——用于与绩效相关的需求
- 物理需求——用于与系统物理方面相关的需求
- 设计需求——用于设计相关的需求



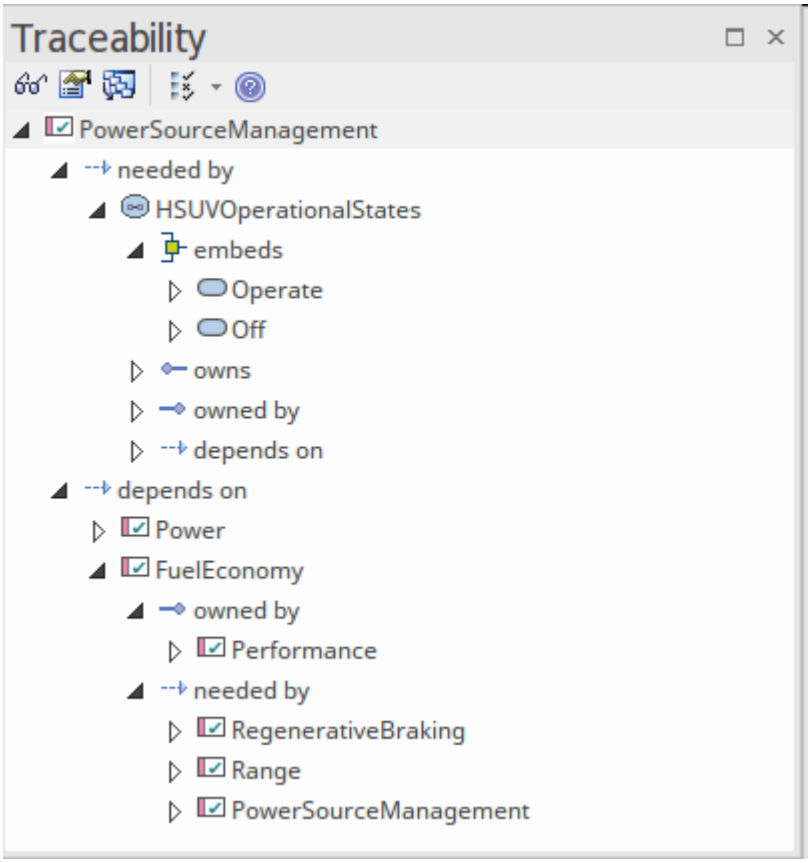
元素可以通过从工具箱拖放到图表视图中添加到图表视图。可以通过首先在工具箱中选择所需的关系，然后在源和目标元素之间拖放来创建关系。通常不创建仅列出需求的需求图，而是创建显示任何两个需求之间的关系或需求与其他元素（例如使用案例、活动和块）之间的关系的图表。

开发需求

需求开发包括与发现、评估、记录、记录和验证特定项目或工作计划的需求相关的所有活动和任务。需求被发现、分析、指定和验证， Enterprise Architect拥有广泛的工具和特征是协助系统工程师开发需求。需求开发的核心是规范管理器，允许需求工程师以电子表格格式以文本形式输入、查看和管理需求。



规范管理器可与图表、可追溯性窗口、讨论功能等其他工具平台配合使用。这些窗口提供了需求的其他视图，让建模者和查看者深入了解需求与存储库的其他部分的关系，并提供通过规范管理器不可见的细节。


Traceability
PowerSourceManagement
needed by
HSUVOperationalStates
embeds
Operate
Off
owns
owned by
depends on
depends on
Power
FuelEconomy
owned by
Performance
needed by
RegenerativeBraking
Range
PowerSourceManagement

引出

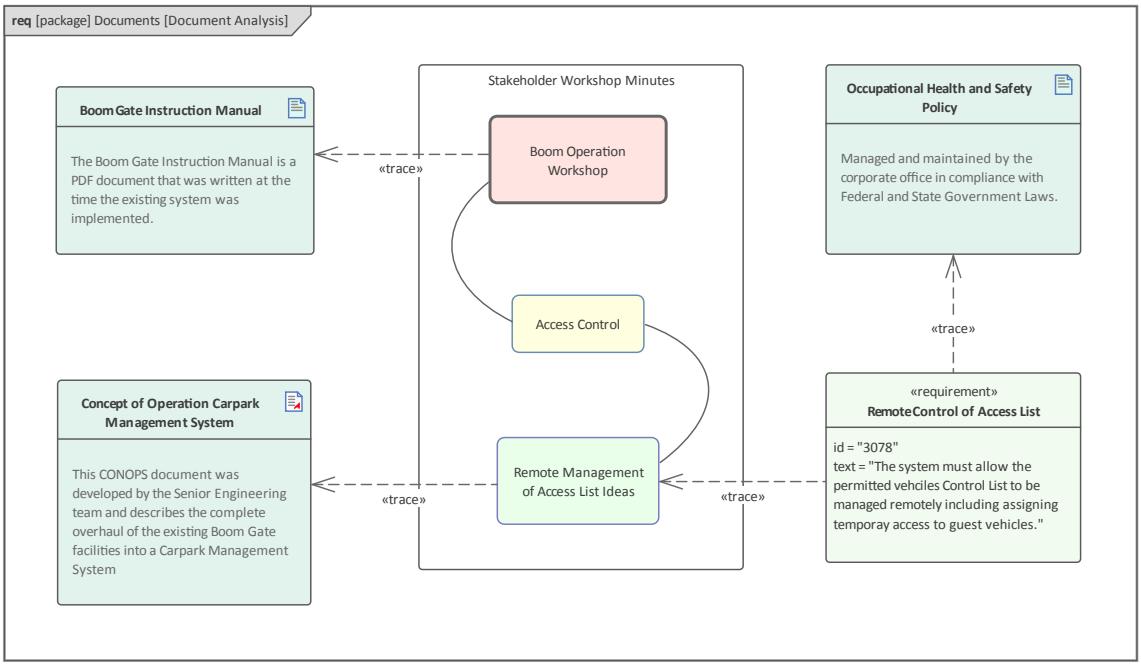
启发是信息发现的过程，从这个过程中收集到的信息将形成需求的前兆。这些信息通常是原始的并且经常是相的，并且直到执行了需求分析才能从中得出真正的需求。启发将采取多种形式，并且需要需求工程师的所有技能来确定要检查哪些文档、机器、工具、人员和流程以发现信息。

文件来源

需求通常可以从广泛的位置获取，包括以下文件：

- 业务案例
- 经营理念
- 需求规格（现有系统的）
- 用户手册
- 标准文件
- 政策文件
- 监管或合规文件

虽然可以使用Enterprise Architect文档工件中的所有文档功能开发文档，但它们通常是在其他工具库中开发的，并且存在于其他存储库中。它们可以被拖到图表上，或者导入到存储库中，或者保存为外部文档的参考或代理。



另一个可能更有用的选项是将它们添加到团队图书馆，这是一个文档和网页库，用于创建可以由需求引用的项目目录。还值得考虑查看文档的内容并将信息合并为模型元素。这有利于工程师能够在诸如业务动机和问题以及解决方案元素（例如需求、使用案例和组件）之间创建可追溯性关系。有关详细信息，请参阅The Model Library帮助主题。

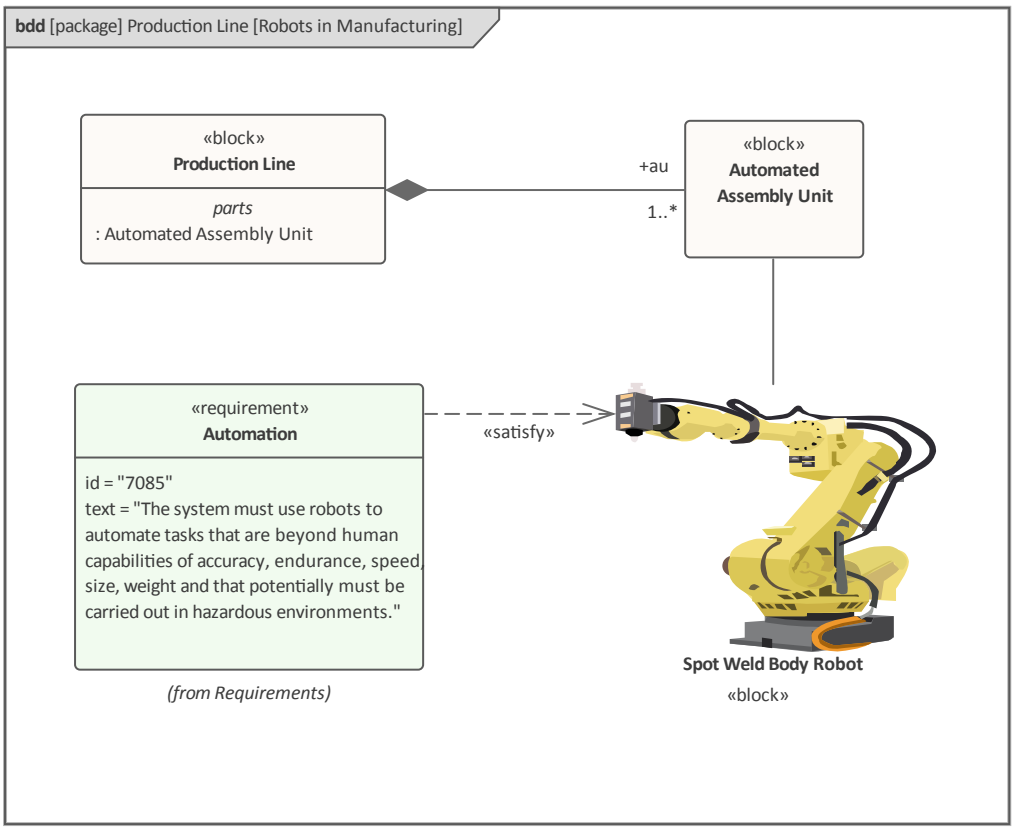
用户观察

观察用户执行他们的工作是了解以下内容的一种有用且不引人注意的方式：

- 他们执行的任务和
- 他们如何使用信息和其他软件和硬件设备来实现与系统交互的结果

即使支持计划系统的流程会有所不同，但对当前流程的观察将为讨论提供有用的上下文。它还将帮助工程师同情用户，从而更深入地了解他们面临的问题，并为发现潜在解决方案提供基础。工程师经常会发现未提及的文件、清单和线索卡，有助于阐明流程。配备手机或相机，工程师还可以为用户工作拍照，这将有助于工程师和其他人在需求分析阶段相和讨论任务。

Enterprise Architect支持建模者直接在模型中表示照片和扫描文档等文件，从而为工作中的用户创建丰富而富有表现力的表示。可以将这些表示为单个键或选项（甚至可以使用工件启动文件）或使用超链接将图像本身包含在图表中。有关更多信息，请参阅Changing Element Appearance主题。

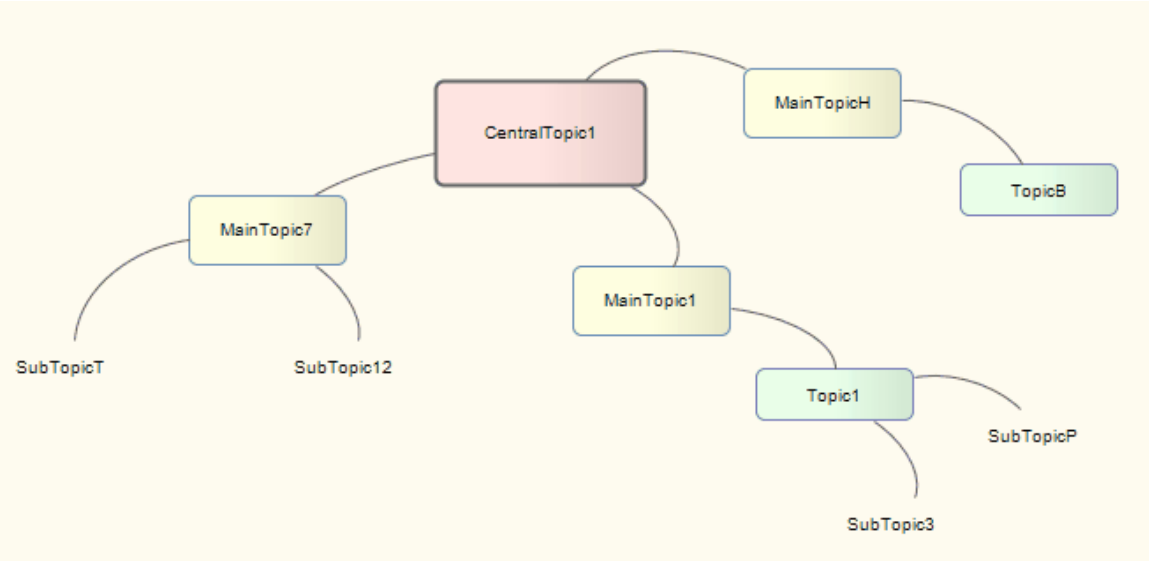


利益相关者研讨会

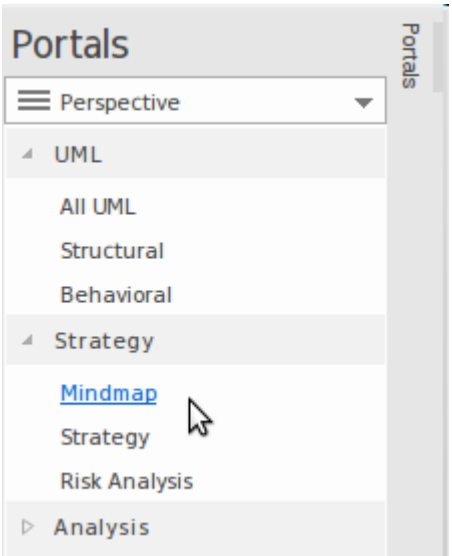
需求工程师负责引出需求的艰巨任务，这需要与包括客户和分析团队在内的利益相关者进行良好的沟通。促进利益相关者需求的一种非常成功的方法是与所有关键利益相关者一起运行研讨会。需求工程师作为沟通者、外交官和调解者的技能对于创造一个有利于探索利益相关者需求和关注点的协作和尊重环境非常重要。工程师必须使用利益相关者理解的术语，并表现出对构成工程领域的元素的理解或学习的意愿。

有时会出现一种误解，即在这些研讨会中阐述的是一组明确定义的要求，这些要求可以作为利益相关者的需求输入到工具中。这与发生的事情相去甚远。利益相关者通常会表达广泛的想法，包括政策、业务规则、数据定义、项目管理约束、功能需求、业务需求、现有系统问题甚至建议的解决方案。即使使用外部顾问来运行这些会议，工程师也没有时间将所有这些陈述归类到会议中。需要一种方法让负责记录语句的抄写员将它们放入工具中，而不用担心正在记录什么类型的信息。将它们记录在工具中而不是在工程师的笔记本中潦草是最佳实践，因为它允许它们在会议期间显示并且利益相关者可以看到彼此的评论。

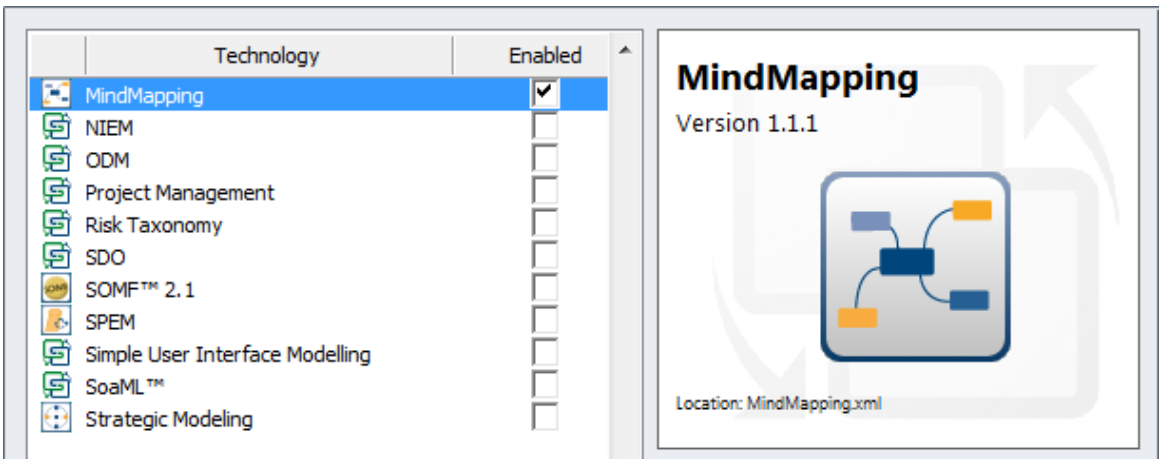
Enterprise Architect有许多功能可以帮助这些研讨会。一种非常实用的方法是使用思维导图来记录涉众的陈述，这种方法非常有效，因为它是一种众所周知的方法，并且没有引入任何建模语言（如 SysML）所附带的形式。此图显示了从可以更改以满足研讨会需要的模式创建的初始思维导图。



思维导图功能可以通过切换到该蓝图来使用，或者，如果经常使用，可以使用我的蓝图功能将其添加到用户定义的蓝图集。

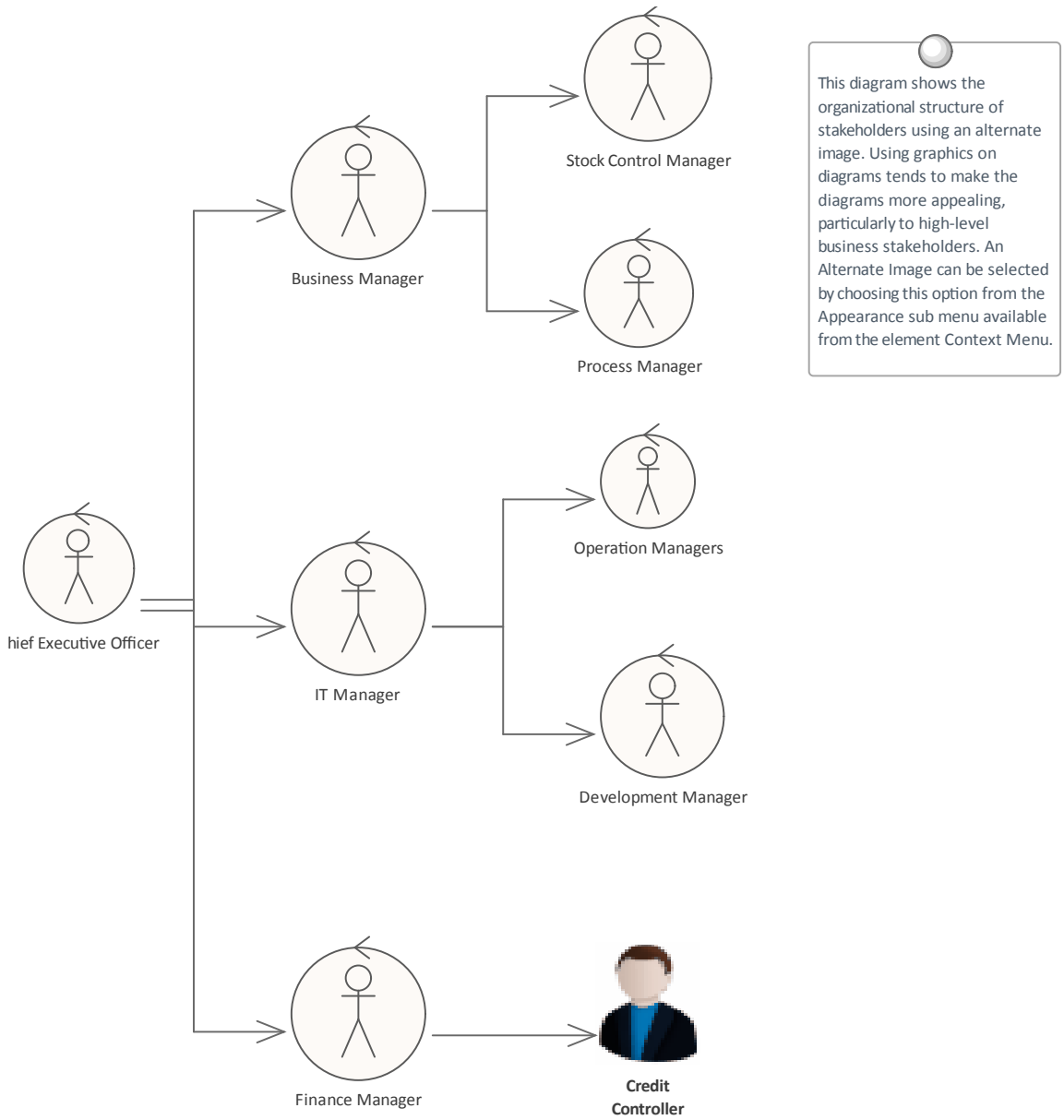


这个蓝图和其他蓝图一样，需要启用适当的技术，在这种情况下是思维导图。



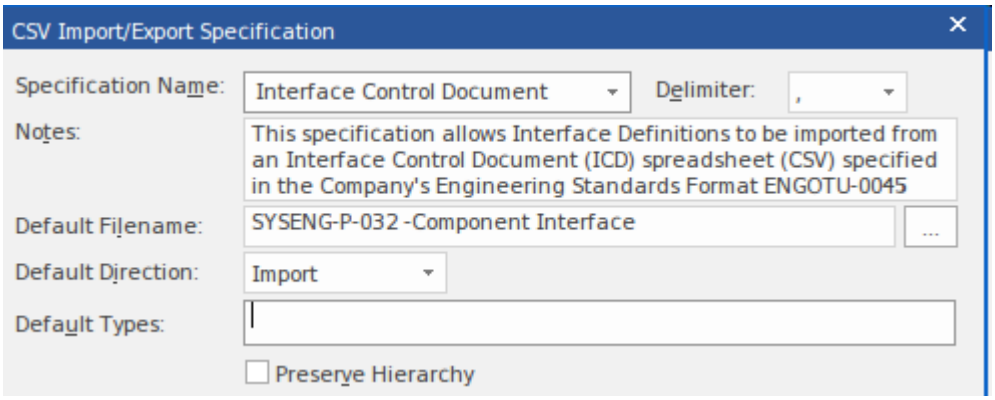
由于发现了重要术语，因此可以将它们输入到项目词汇中，即使没有时间讨论和辩论商定的含义，这些词也将作为该领域中重要实体的初始列表。或者，可以将术语创建为块定义图中的块，并通过描述术语之间重要关系的连接器相互关联。

还可以对利益相关者进行建模，并且可以在图表中描述他们彼此之间的组织关系。这是一种有用的技术，它允许关键利益相关者在模型中识别自己，从而产生认同感。


Stock Control Manager
Business Manager
Process Manager
This diagram shows the organizational structure of stakeholders using an alternate image. Using graphics on diagrams tends to make the diagrams more appealing, particularly to high-level business stakeholders. An Alternate Image can be selected by choosing this option from the Appearance sub menu available from the element Context Menu.
Operation Managers
hief Executive Officer
IT Manager
Development Manager
Finance Manager
Credit Controller

创造需求

Enterprise Architect对开发需求有广泛的支持，并为此提供了许多专门的工具。与所有模型内容一样，鼓励建模者在开始创建新需求之前检查其他else是否已将需求输入到存储库中。需求也有可能已经在另一个工具（例如电子表格）中定义，并且可以导入Enterprise Architect而无需手动创建每个需求。



Enterprise Architect有两个需求点；它们可以在模型中创建为将出现在浏览器窗口中的元素，也可以在另一个元素中创建为内部需求或责任。

外部和内部需求

Enterprise Architect可以支持任何类型的需求过程，并允许将需求定义为模型中的元素。这些称为外部需求，但该工具还允许为特定元素定义需求，这些称为内部需求。想要定义用户需求的工程师，例如：

系统必须允许更新公交时刻表。

将使用外部需求。想要描述元件应该如何表现A部件者会使用内部需求来部件，例如：

编辑器必须支持统一码。

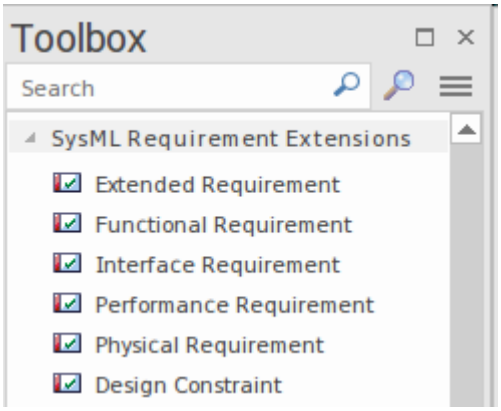
分析师和开发人员之间经常会争论需求应该是内部的还是外部的，而Enterprise Architect提供了一种功能，可以将内部需求转移到元素的外部。当它们被移动时，它们仍然与原始元素相关联。

需求分类

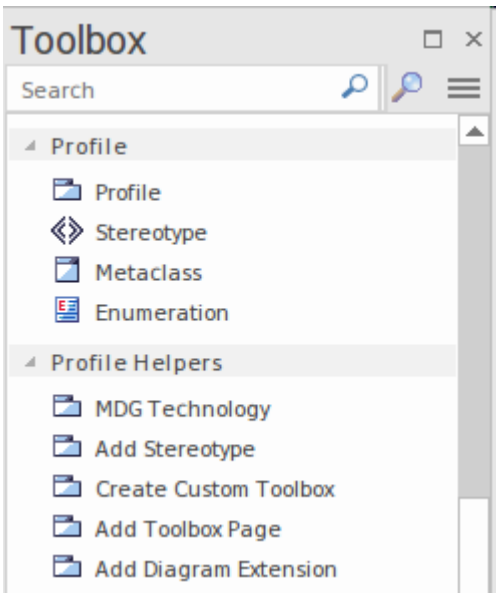
SysML 规范提供了需求类别（类型）的非规范性列表。这些是对基本 SysML需求进行改进或扩展的定型需求，提供了一种机制来创建服务于特定目的或描述系统特定方面的需求。例如，物理需求可用于描述系统的某些物理方面，例如组件的重量或大小。这些和其他用户创建的类别可以定义任意数量的附加属性，例如：

- 风险种类
- 验证方法种类

Enterprise Architect方便地将这些需求类别作为元素提供在 SysML需求的工具箱页面上。



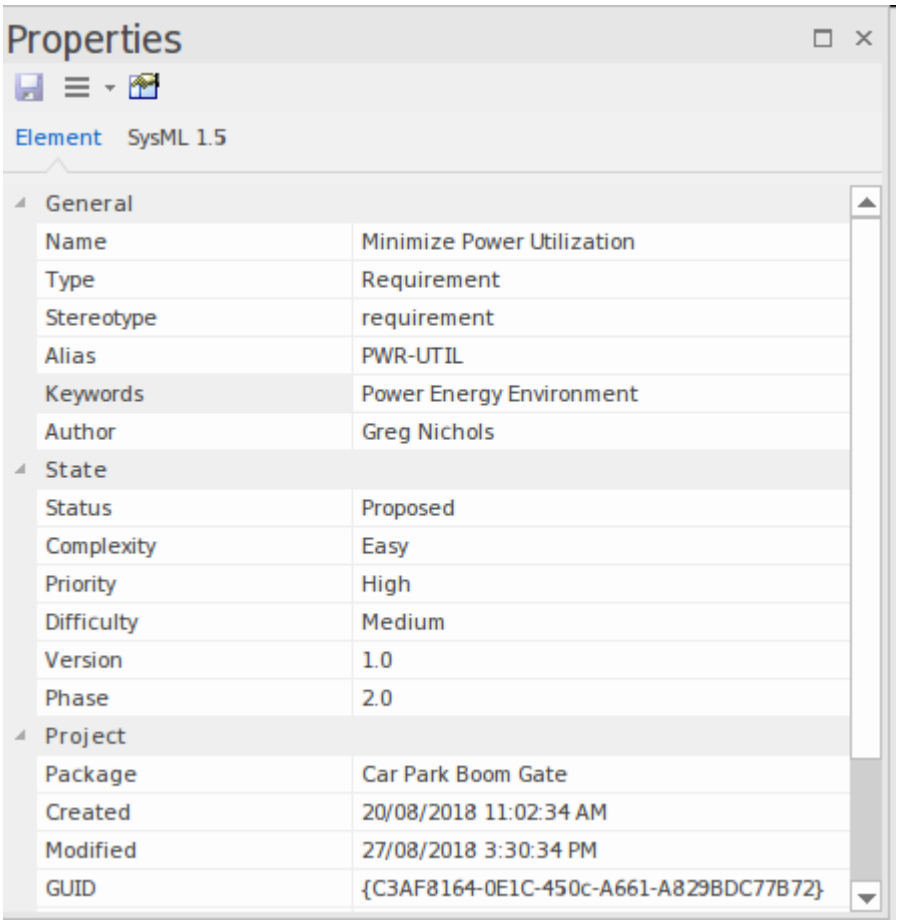
该工具还提供了一个复杂且功能齐全的配置文件系统，允许用户创建基本 SysML需求的扩展以及适用于建模域或问题空间的任意数量的用户定义需求类别。这些定型需求可以具有添加到模型特定需求元素（或其他元素模型）所需的用户定义属性。



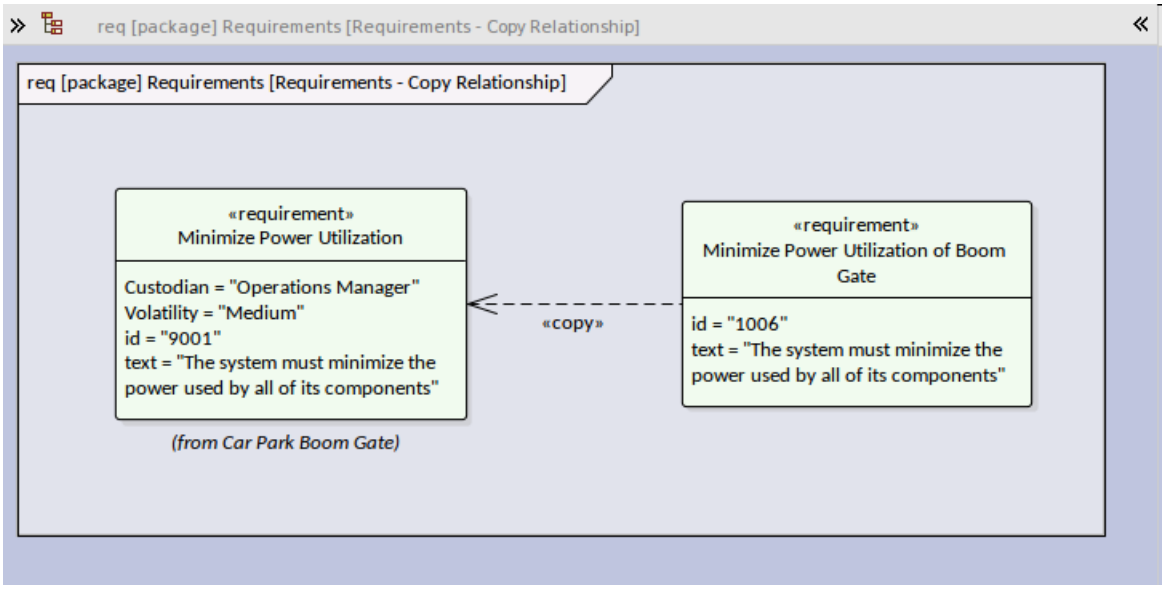
例如，团队可能决定在需求中包含波动性属性，以确保在需求稳定（即不波动）之前不会开始工作。作为另一个例子，一个团队可能正在创建医疗设备，并且需要遵守各种法定标准。用作解决方案一部分的每个组件可能都需要符合要求。 A创建合规级别的属性，并且可以为组件分配一个级别，该级别指示其合规性，该级别来自旋转控件或下拉列表中定义的一系列值。有关详细信息，请参阅Developing Profiles帮助主题。

需求属性

需求开发和管理对于任何项目的成功都至关重要，需求属性对于优先级以及它们在实施或开发团队中的详细说明和使用方式很重要。所有Enterprise Architect元素都有标准属性，例如状态、作者和相，但需求元素具有附加属性，例如难度和优先级。



一些需求过程将规定特定的属性，例如保管人和波动性（稳定性），这些可以使用可以应用于每个需求的标记值进行配置。需求的 注记”字段具有特殊意义，因为它通常包含系统必须如何运行或执行的正式和合同描述。有关详细信息，请参阅Element Property Displays帮助主题。


req [package] Requirements [Requirements - Copy Relationship]
req [package] Requirements [Requirements - Copy Relationship]
«requirement»
Minimize Power Utilization
Custodian = "Operations Manager"
Volatility = "Medium"
id = "9001"
text = "The system must minimize the power used by all of its components"
(from Car Park Boom Gate)
«copy»
«requirement»
Minimize Power Utilization of Boom Gate
id = "1006"
text = "The system must minimize the power used by all of its components"

规格

需求规范是需求演变的一个重要方面。它提供了关于系统在正常和异常情况下的行为的重要陈述目录。需求将引起广泛的利益相关者的兴趣，包括：

- 工程经理
- 架构师
- 设计师
- 客户或其代理人
- 系统工程师
- 软件工程师
- 测试人员
- 合规经理
- 质量工程师
- 安全工程师

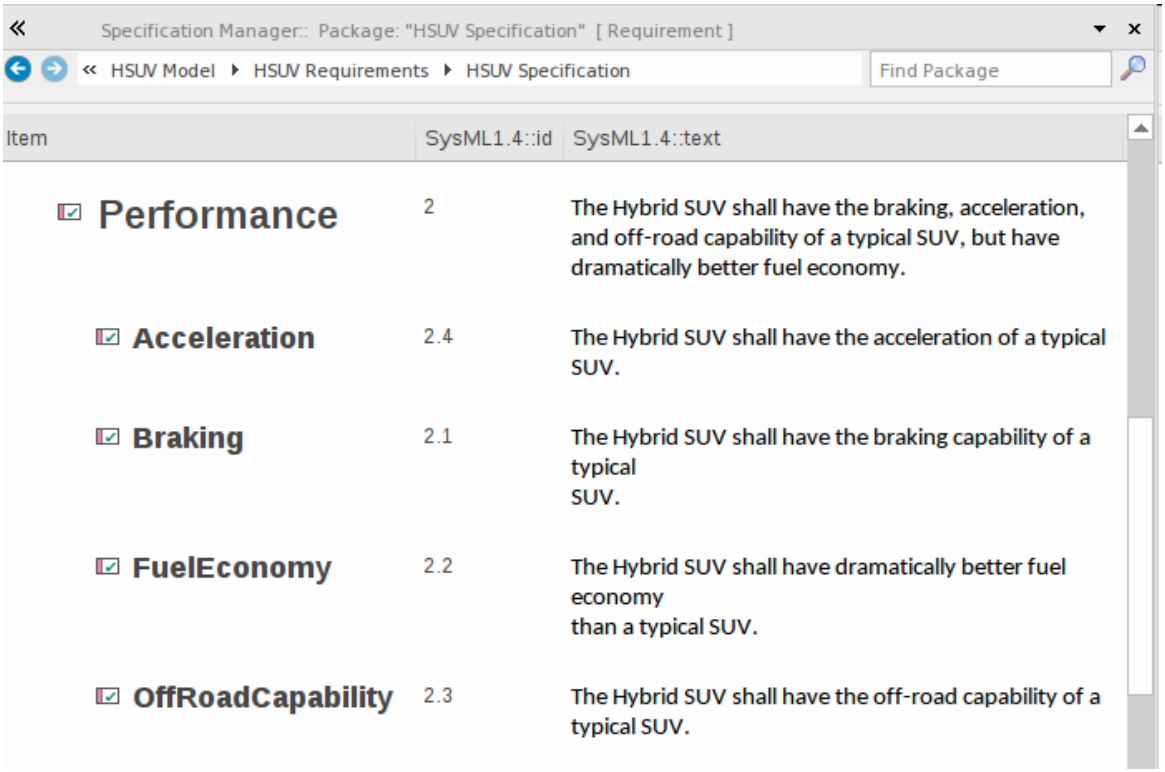
所有这些小组都会有需求的输入，并且需要在他们的工作中使用需求目录。在Enterprise Architect中可以通过多种方式指定需求，包括：

- 直接在浏览器窗口中
- 在图表上
- 使用规范管理器

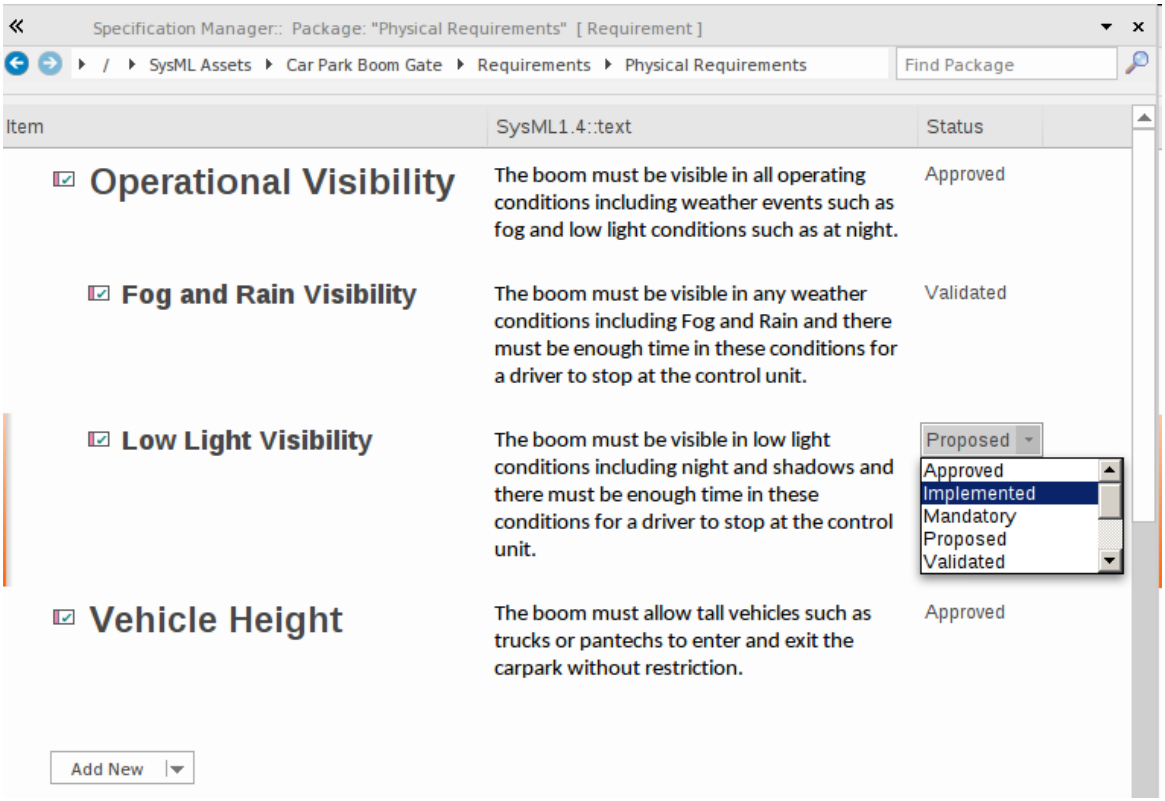
我们将在下一节中查看规范管理器，您会看到它在处理需求和其他带有文本内容的元素时提供了极大的灵活性。

满足规范管理器

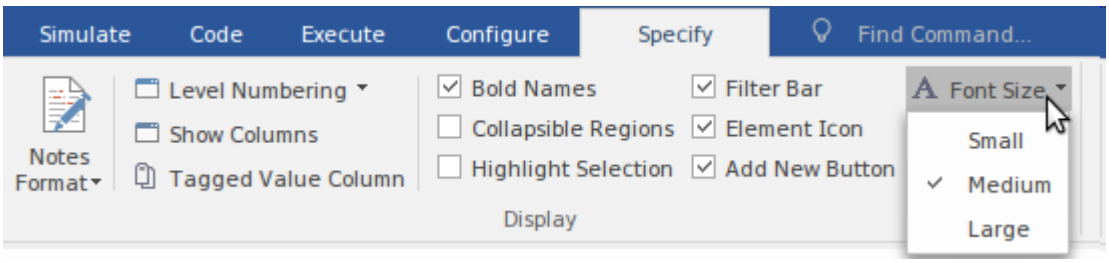
规范管理器是一个独特而有效的工具，它提供了可用于管理任何元素的电子表格或文字处理器视图，尽管它在使用总是有描述性文本来描述需求细节的需求时特别有用。可以使用名称和详细描述创建新的需求，并且可以从下拉列表中添加或更改状态和优先级等属性。现有的需求可以在方便的视图中查看和管理 - 例如图表和窗口 - 在规范管理器中更改它们将在存储库中的所有其他位置更改它们。



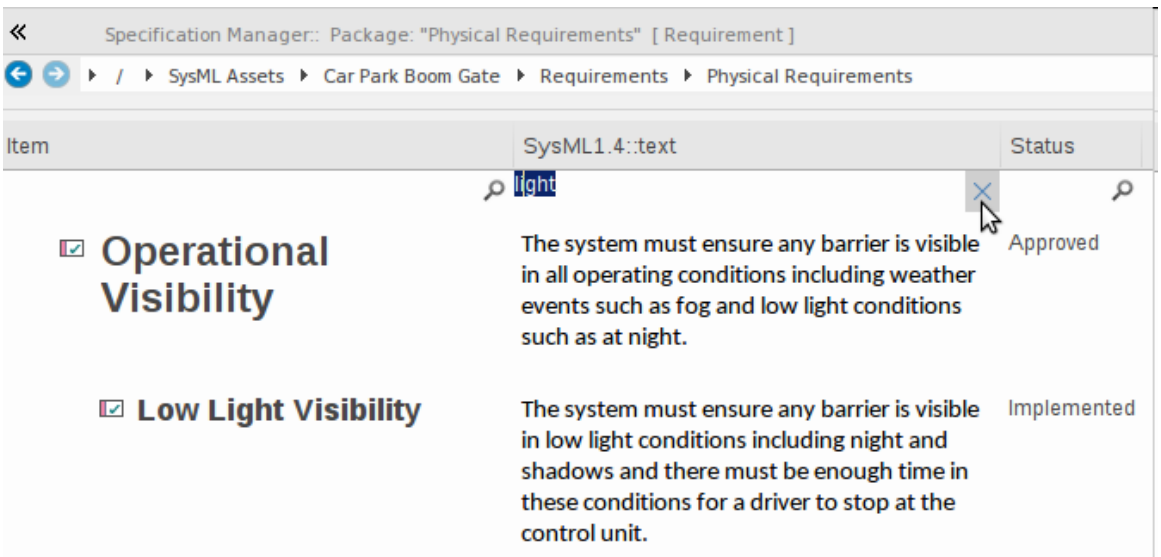
对于那些更习惯使用文本而不是图表以及习惯于使用文字处理器或电子表格工作的分析师来说，规范管理器是完美的工具。它的另一个优点是需求是模型的一部分，并且可以追溯到其他元素，包括驱动业务驱动因素、利益相关者和解决方案组件。在该图中可以看出，需求状态和其他元素属性可以从下拉列表中进行编辑。



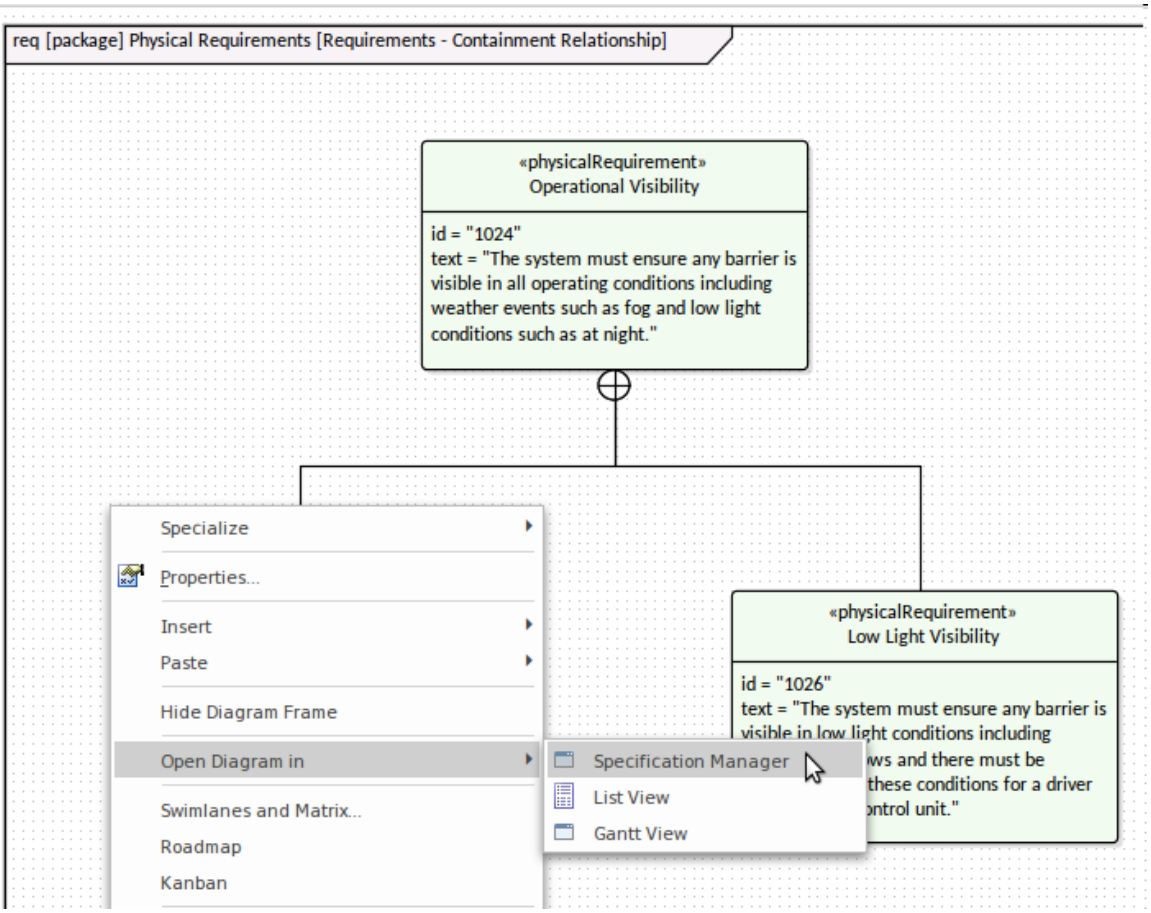
在使用规范管理器时，有多种选项可提供极大的灵活性，包括在电子表格中的列中显示注记或在文档中的内联显示，以及调整文本的大小。这些选项可从 规范 - 指定”功能区获得，该功能区在启动规范管理器时有条件地显示。



过滤器提供了一种有用的方法来将显示限制为在选定列中包含单词或文本片段的元素。在此插图中，建模者决定将显示限制为在需求需求在处理大量需求时，这是一个很好的生产力工具，它可用于定位具有特定状态、优先级、复杂性的所有需求，甚至是指定利益相关者或团队拥有的所有需求（如果已定义）**在模型中**。



图表也可以从规范管理器中打开，允许您A图表上的元素作为一个组进行编辑。对于包括经理和客户在内的一些非技术人员来说，这是一个令人信服且受欢迎的观点。有关详细信息，请参阅规范管理器帮助主题。



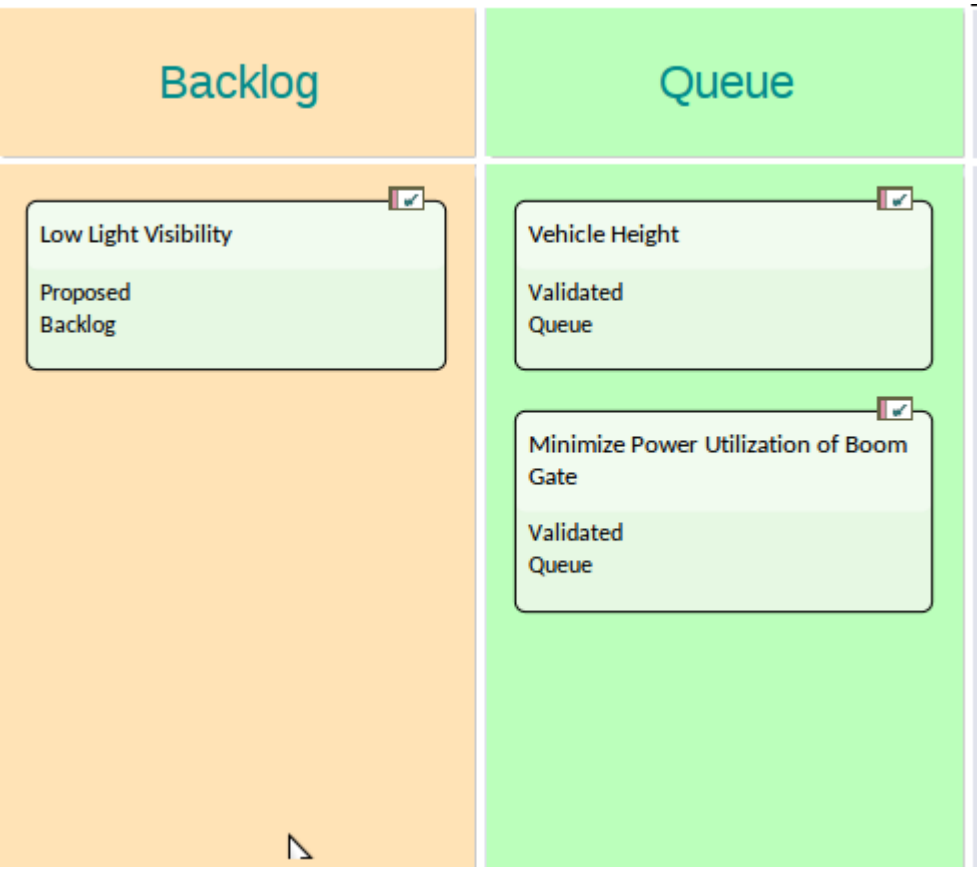
分析

需求开发的分析相确保在启发相发现的需求得到了正确的表达，并具有正确的格式、细节和属性，并形成一个正确的、有凝聚力的集合。由于引出的来源和方法不同，相阶段中记录的需求将需要一些按摩和平衡——例如，发现重复或重叠的需求或系统工程师无意中忽略了记录需求是很常见的。一个或多个利益相关者的担忧。关系矩阵和可追溯性窗口等工具将有助于揭示需求的遗漏和问题。 Discuss &审阅窗口和 Chat & Mail 窗口——包括模型邮件功能——还将提供　其他团队成员讨论、审查和聊天需求的机制。

Source \ Target	REQ011 - Manage User Accounts	REQ012 - Provide Online Sales	REQ013 - Manage Deliveries	REQ014 - ShoppingBasket	REQ015 - Process Credit Card Payment	REQ016 - Add Users	REQ017 - Remove User	REQ018 - Report on User Account	REQ019 - Manage Inventory	REQ020 - Receive Books	REQ021 - List Stock Levels	REQ022 - Order Books
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

优先满足需求

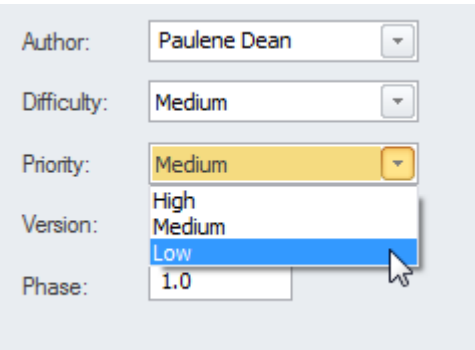
优先考虑需求对于项目的成功至关重要，因为它确保分析、开发、测试和实施资源集中在系统的最关键方面。优先级是为每个需求分配优先级的决策过程；最常见的分类标准是业务价值。业务价值通常由实施需求将为组织或其客户产生的价值的成本效益分析确定。其它因素可能是政策或法规遵从性、紧迫性、业务或技术风险和成功的可能性。需求可以在看板中可视化，该看板可以通过将项目从积压通道移动到队列泳道来指示优先级，并且还允许在通道内订购项目。有关更多信息，请参阅Kanban Boards帮助主题。



或者，可以使用搜索或模型视图来创建基于一些标准的需求列表，这些标准将使需求得到优先考虑。

需求优先属性

有多种标准可用于确定优先级，每个组织和项目通常会使用某种类型的加权平均值来确定优先级。 Enterprise Architect对需求优先级提供灵活和完成的支持，因为每个元素都有一个内置的 优先级"属性，可以设置为指示其优先级，允许用户从下拉列表中选择分配的优先级。

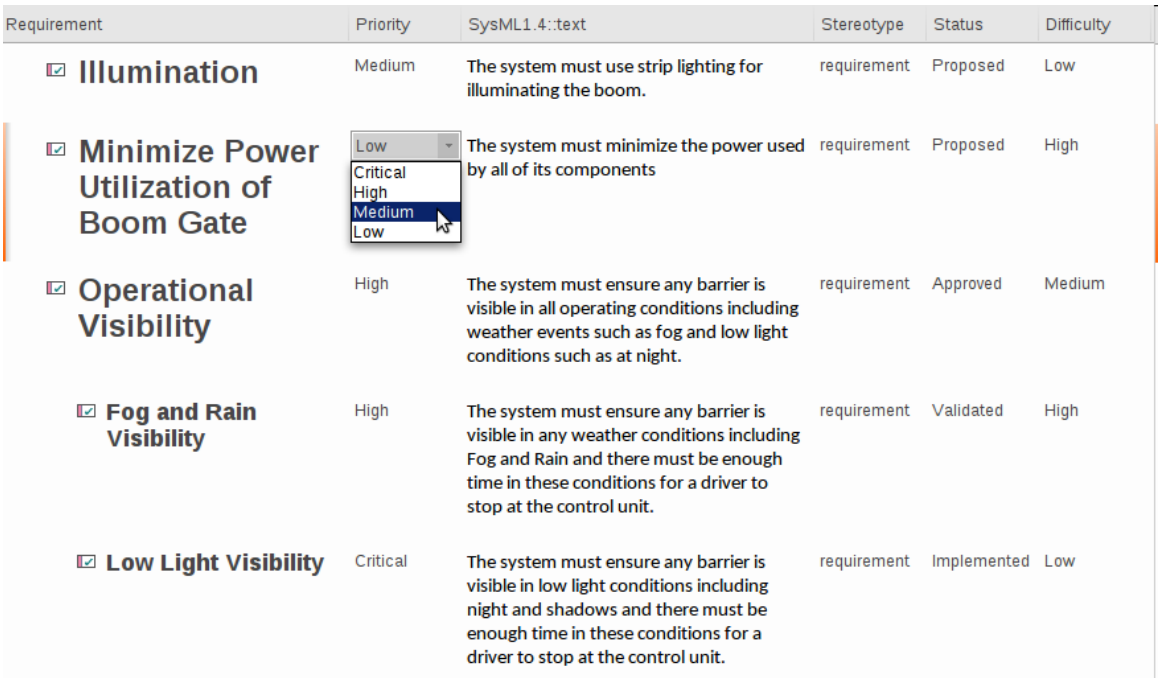


当您安装Enterprise Architect时，可以方便地预先加载优先级列表，但是可以编辑或完全修改这些优先级列表以适应组织或项目。它们甚至可以作为参考数据从以前的项目中导入，或者，如果当前项目是基于模板创建的，则可以从基础模型中预加载组织的优先级。可以使用此功能区选项设置它们：

设置 >参考>模型类型 > 常规类型 > 优先级

协同改变优先级

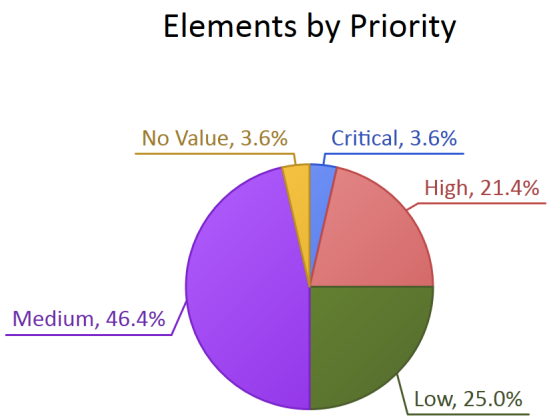
选择标准和分配优先级的过程通常是协作的，并且通常在与利益相关者或其代表辩论分类的研讨会中完成。在以前的时代，这是一个费力且困难的过程，但Enterprise Architect具有一些有用的特征来处理需求属性，包括优先级。有许多窗口——包括包列表和图表列表——支持处理需求和在线编辑优先级，根据新分配的优先级自动过滤或排序需求列表。规范管理器是一个有用的工具，它提供了一个基于文本的界面，可以查看需求及其注记，并可以从下拉列表中选择优先级。该界面还显示了许多其他通常对优先级有用的属性，例如状态和复杂性。有关更多信息，请参阅Editing Elements帮助主题。



当需求属性被更改并保存在任何窗口或图表中时，该属性将在所有其他视图中更改，并且查看存储库的任何其他用户将立即能够看到更改。

仪表板图表

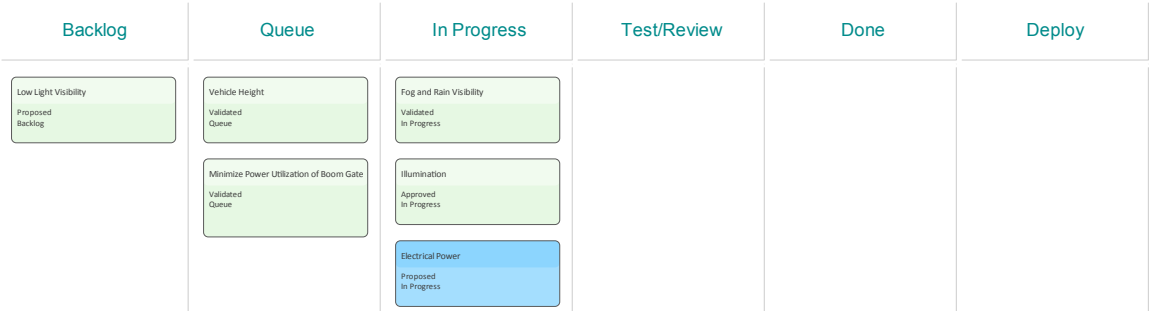
Enterprise Architect有一系列仪表板图，可用于创建包中需求优先级的引人注目的视图，并可选择包含子包。有许多预配置的图表可用于显示部分模型中需求优先级值的比率。过滤器添加了另一个级别的用户配置，例如，允许建模者排除特定状态的需求或相仅显示当前需求的需求。有关详细信息，请参阅Dashboard Diagrams帮助主题。



This diagram shows a Pie Chart element depicting element priorities for all the Requirements in a selected Package. It provides a useful summary for a Requirements Manager and is dynamically updated when the priority changes and the diagram is reopened. A range of other pre-defined Charts and user-defined Charts can also be added. A filter has been added to exclude all elements other than Requirements.

使用看板进行可视化

Enterprise Architect有一个看板图，可用于管理需求和其他规范或项目管理元素，例如更改。看板对于管理需求和其他元素的优先级特别有用。这些元素可以简单地拖到图表上，然后在列之间，允许团队管理和可视化需求在规范和实施之间的进展。

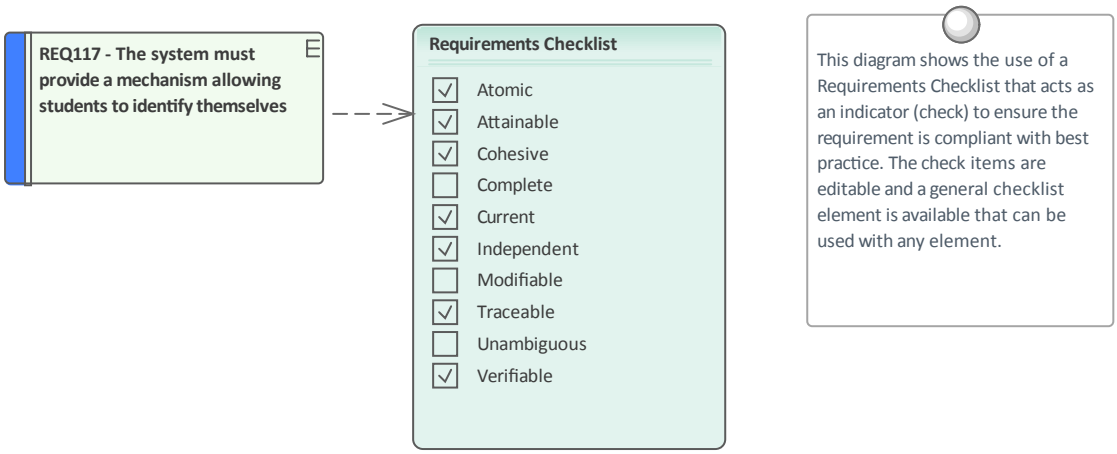


可以配置看板图，以便在列之间拖动元素时自动更改元素的优先级。有关更多信息，请参阅Kanban Boards帮助主题。

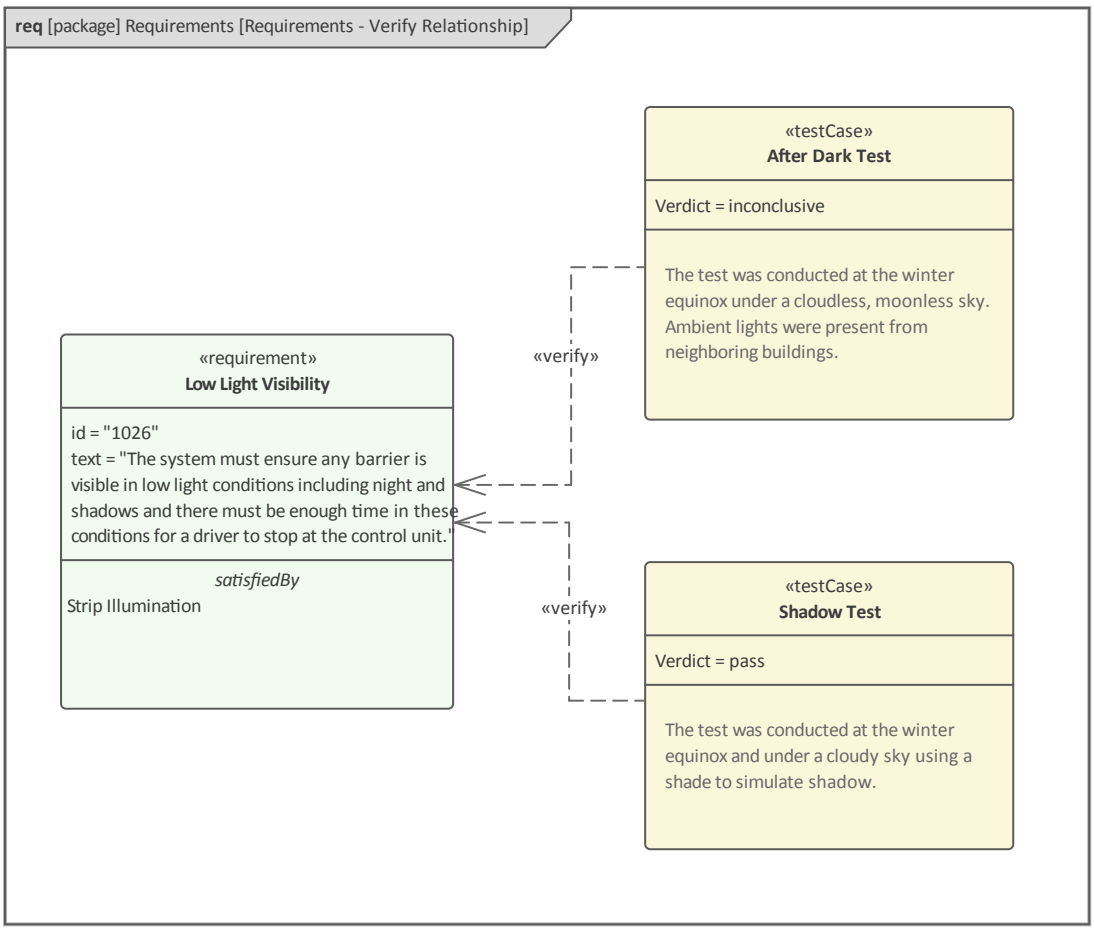
验证

需求验证是必要的，以确保需求是高标准的，适当地定义顾客的问题（或机会），并足以让实施团队设计和实施产品。要求必须具有所需的质量水平，并且是完成和必要的。有许多方法可以验证需求，但最常见的两种方法可能是执行团队评审和将测试用例分配给需求。

团队评审通常由对领域有一定了解的团队成员或其他分析师进行，但他们自己不负责需求开发或管理。Enterprise Architect有一个方便的工具来协助这个过程，称为团队图书馆，它适用于整个模型，并允许审阅者在讨论文档中记录他们的发现并参考模型元素。需求工具箱的 扩展需求”页面还有一个需求检查清单元素，它提供了一个有用的检查需求质量的机制。



测试用例可以定义在从用户验收测试到单元测试的多个级别。在需求开发过程的早期定义测试用例会创建对需求的双重检查，因为当定义测试用例时，通常会发现需求问题。 Enterprise Architect具有许多定义测试用例的功能，建模者可以选择最适合该工作的任何一个。

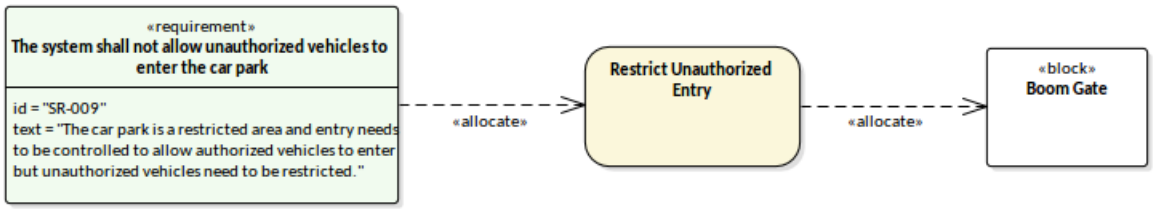

req [package] Requirements [Requirements - Verify Relationship]
«testCase»
After Dark Test
Verdict = inconclusive
The test was conducted at the winter equinox under a cloudless, moonless sky. Ambient lights were present from neighboring buildings.
«verify»
«requirement»
Low Light Visibility
id = "1026"
text = "The system must ensure any barrier is visible in low light conditions including night and shadows and there must be enough time in these conditions for a driver to stop at the control unit."
satisfiedBy
Strip Illumination
«verify»
«testCase»
Shadow Test
Verdict = pass
The test was conducted at the winter equinox and under a cloudy sky using a shade to simulate shadow.

可视化需求

需求可以使用产品的不同特征以多种不同方式可视化。一些方法允许用户创建新的和编辑现有的需求，而另一些方法只是提供一种查看需求的方法。需求，就像 SysML 中的所有元素一样，可以形成图形的一部分，许多重要的语义都在这些中表达关系；例如需求和验证它的测试用例之间的关系。然而，需求分析师、经理和其他利益相关者希望独立于需求可能参与的任何关系来查看需求是很常见的。我们现在将研究其中的一些功能，其他功能将在现在化中介绍需求关系话题。

需求图表

需求图可用于可视化需求及其与其他元素的关系，包括其他需求。您当然可以通过许多其他方式查看需求，但对于许多利益相关者来说，需求图将更具吸引力，因为它提供了一种图形方式来查看需求与模型其他重要部分的联系，包括利益相关者、架构、设计和测试。此图显示了如何查看需求以及它们与其他元素的连接。要求未经授权的车辆不应被允许进入停车场被分配给活动 限制未经授权的车辆”，而这又被分配给代表需求门A块。

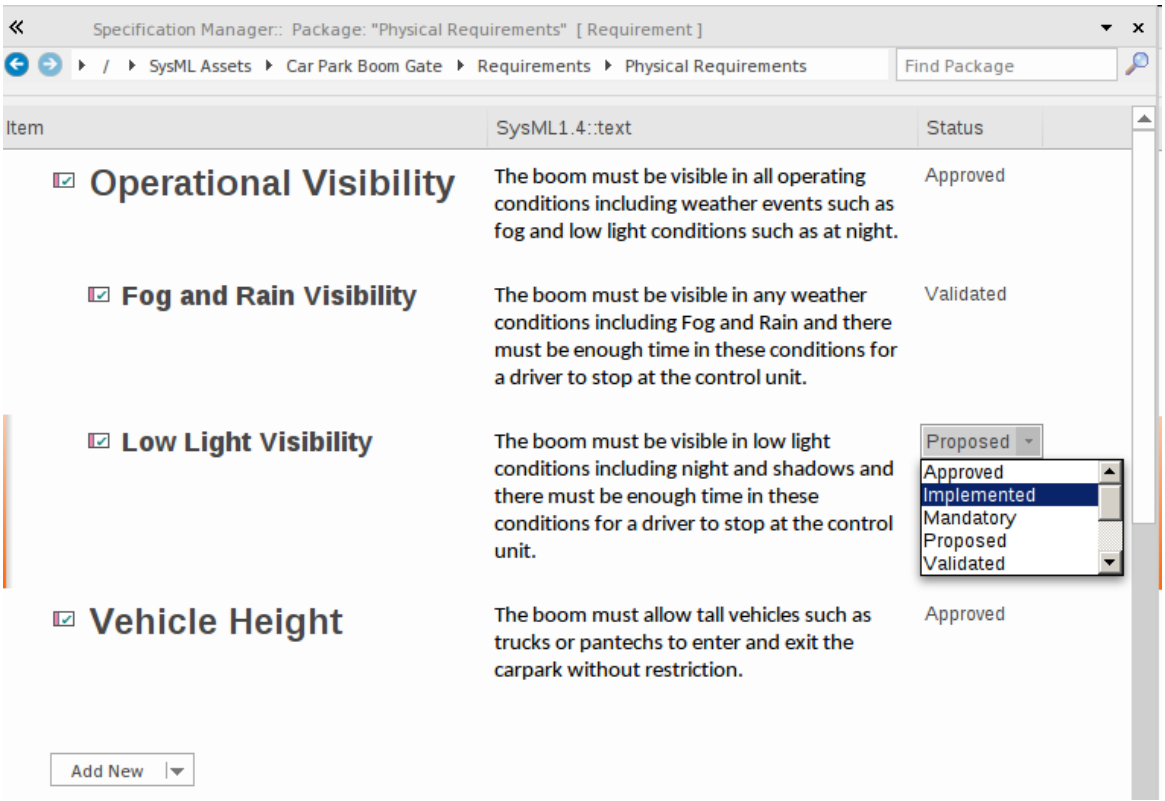


可以从用户界面中A许多不同位置创建需求图，包括从功能区选项 设计>图表>添加图表”。

将显示 新图表”对话框，可以从 SysML 图表类型列表中选择需求图表。有关更多信息，请参阅SysML Requirements Modeling帮助主题。

规范管理器

规范管理器是Enterprise Architect中处理需求的核心工具，从一开始就被设计成一个允许通过直观且功能齐全的界面创建和管理需求的工具。对于那些习惯使用电子表格或文字处理器文档的工程师或其他利益相关者来说，该工具看起来很自然，并模拟了这两种可视化模式，允许用户在电子表格模式和文档模式之间切换。



规范管理器可以从功能区打开： 设计>包>规范视图”。

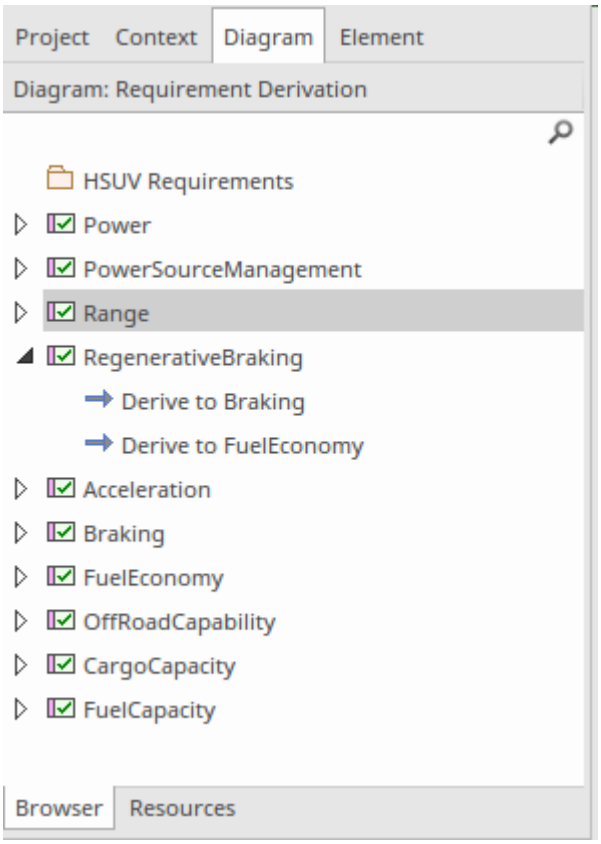
规范管理器可用于查看任何类型的元素，但特别适用于文本成分较大的元素，例如需求。在工具中更改的任何元素也将在需求图表、浏览器窗口和其他目录（如库表）中自动更改。更多信息参见Specification Manager 帮助话题。

浏览器和视图

浏览器是中央导航工具，可用于构建和探索存储库的内容，包括处理需求。浏览器有许多选项卡，允许以特定方式查看存储库的内容。我们已经在前面的主题中查看了浏览器，但将讨论它与可视化需求的相关性。

大多数系统工程师会尝试将他们对特定项目或努力的要求保持在一个位置，尽管可能存在需要将它们分开的情况；例如，出于合同或日程安排的原因。在 项目"选项卡中选择需求包后，用户可以切换到 上下文" 选项卡以获得焦点视图 - 有效地消除该时间之外的其他元素的上下文。然后可以选择一个单独的要求，并在检验员窗口上选择 详细信息"选项卡以聚焦所选要求的属性。

打开图表中包含的元素和关系也可以通过浏览器的 图表"选项卡进行可视化，提供查看图表内容的另一种方式。



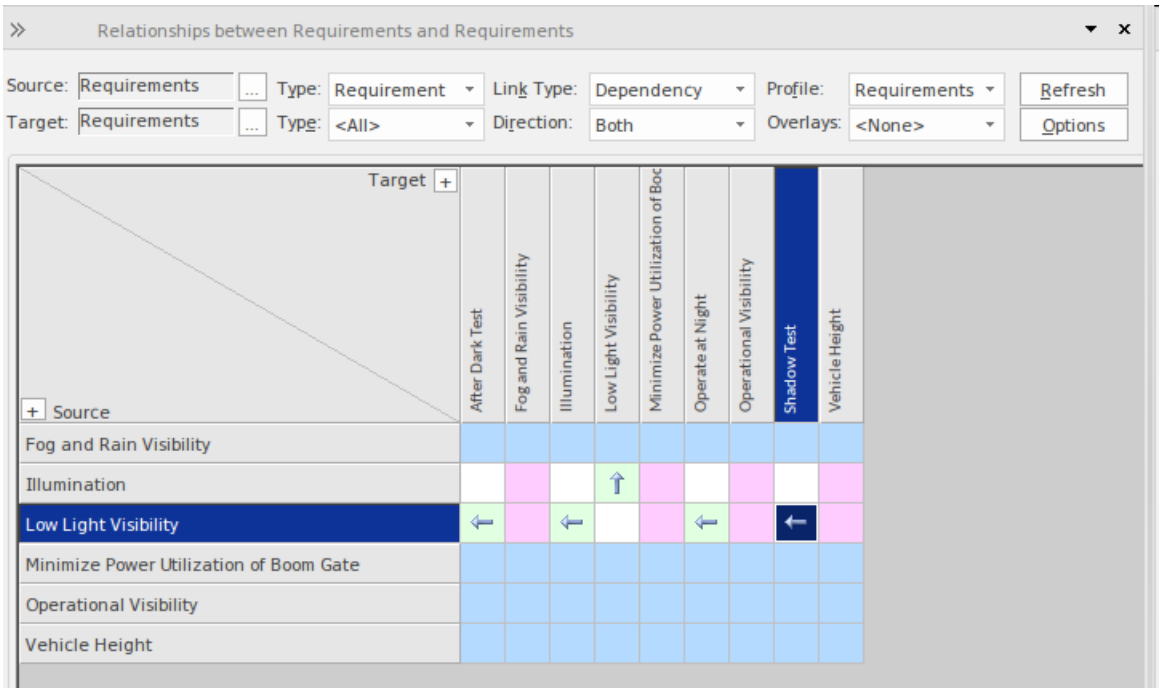
任何包或图表也可以通过多种不同的方式进行可视化，包括提供列表中元素的常用视图的列表视图，类似于电子表格，其中需求按行列出，其属性按列列出.

Status

Status	Name	Type	Modified
Status: Approved			
Approved	OffRoadCapability	Requirement	30/01/2020
Approved	FuelCapacity	Requirement	30/01/2020
Status: Implemented			
Implemented	Braking	Requirement	30/01/2020
Status: Proposed			
Proposed	Power	Requirement	27/03/2018
Proposed	Range	Requirement	27/03/2018
Proposed	CargoCapacity	Requirement	5/03/2019
Proposed	RegenerativeBraking	Requirement	27/03/2018
Proposed	PowerSourceManagement	Requirement	27/03/2018

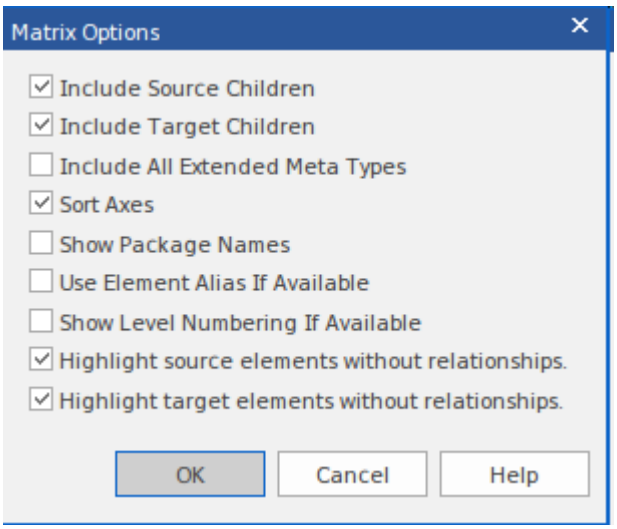
关系矩阵

关系矩阵是一个有价值的工具，用于在一个类似于具有行和列的电子表格的界面中可视化任何两个包中元素之间的连接。该工具在与需求一起使用时特别有用，并且允许工程师查看需求与其他元素的关系，**包括其他需求**。



关系矩阵可以从功能区选项 设计>**包**>**包**/矩阵"打开。选择当前包是源包、目标包还是两者都有。

如果存在关系，则在源元素和目标元素的交叉处的单元中会显示一个箭头图标，箭头表示关系的方向。矩阵还可以配置为以单独的颜色突出显示没有任何关系的行或列。可以在 选项"窗口中配置此选项和其他选项（单击关系矩阵标题中的选项按钮）。



这些选项允许您定制矩阵的显示方式，包括是否对元素进行排序或其名称以包名为前缀，以及是否突出显示源和目标元素的行和不连接的列。有关更多信息，请参阅Relationship Matrix帮助主题。

需求表

需求表是类似于电子表格的视图，可以使用#

创建#

基于 select 语句选择需求（或任何其他元素）的语句，该语句有效地过滤掉特定的需求组。例如，表可用于显示所有与电力子系统相关的、已获批准且具有高优先级或用于分解性能要求的需求。可以创建任意数量的库表，它们会随着存储库中基础属性的更新而动态刷新。这提供了比列表视图更大的灵活性，因为能够根据指定的标准从存储库中的任何位置选择一组需求。

req [requirement] Performance [Decomposition of Performance Requirement]

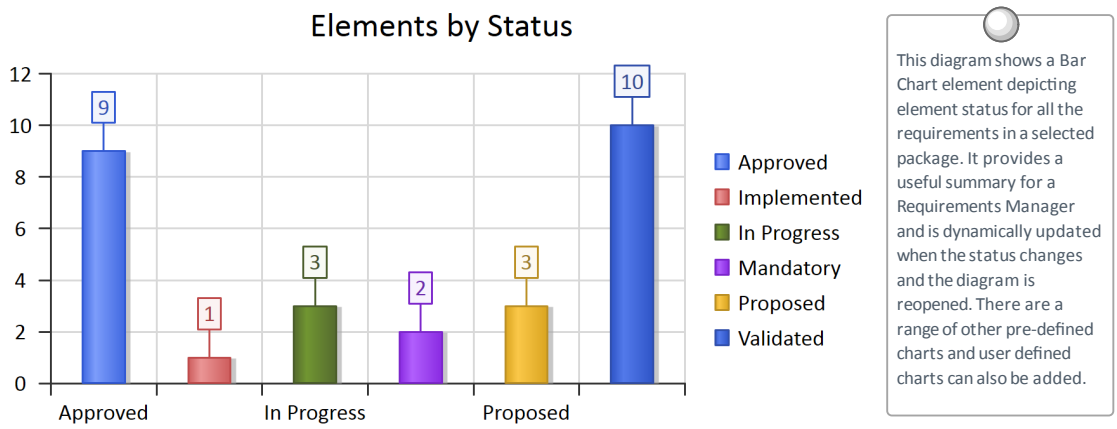
Decomposition of Performance Requirement

ID	NAME	TEXT
2	Performance	The Hybrid SUV shall have the braking, acceleration, and off-road capability of a typical SUV
2.1	Braking	The Hybrid SUV shall have the braking capability of a typical SUV.
2.2	FuelEconomy	The Hybrid SUV shall have dramatically better fuel economy than a typical SUV.
2.3	OffRoadCapability	The Hybrid SUV shall have the off-road capability of a typical SUV.
2.4	Acceleration	The Hybrid SUV shall have the acceleration of a typical SUV.

Showing 1 - 5 of 10 items

管理需求

这包括维护一组要求的活动，这些要求代表项目团队和客户之间的协议或协议。它还涉及确保设计和实施团队可以接受这些要求，并且它们足以使他们指定的内容可以实施到工作业务、软件或硬件系统中。 Enterprise Architect是一个复杂的需求管理平台，无论领域、项目规模或所遵循的方法如何，**都有一些工具可以直接管理**复杂项目中的大型需求存储库。

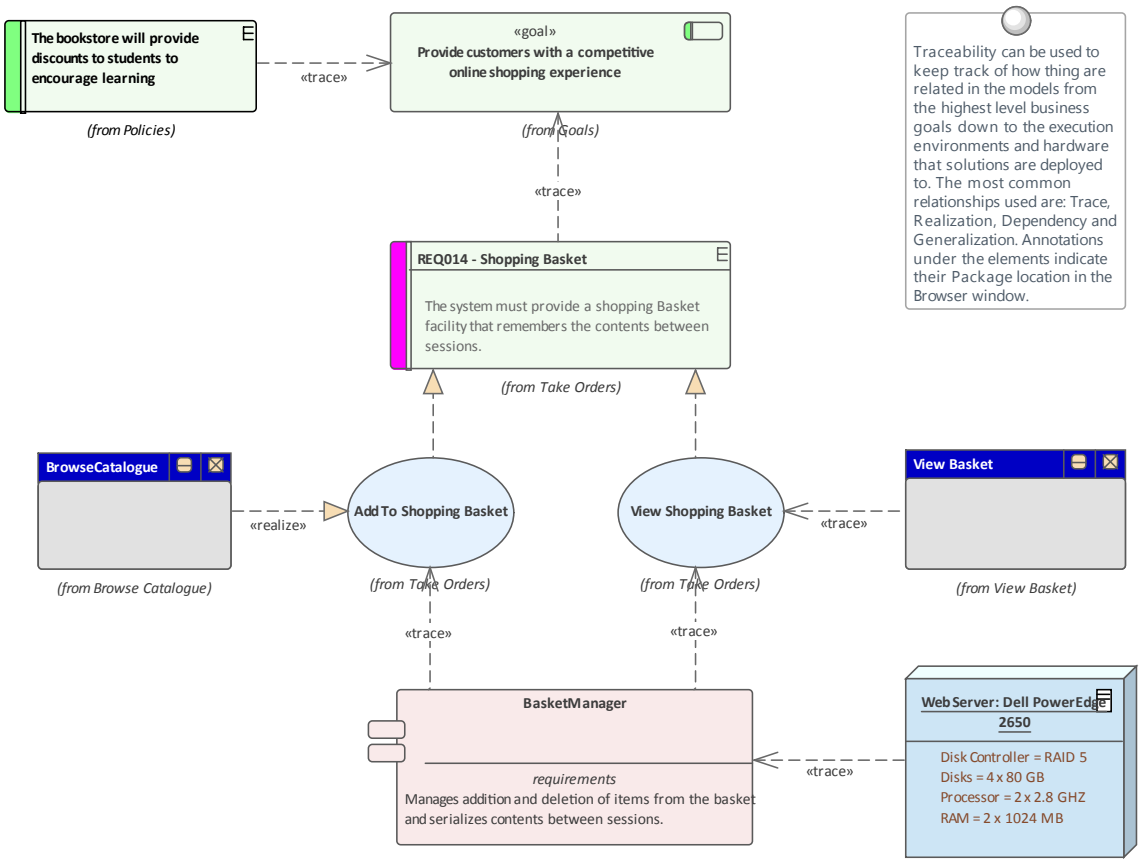


追踪需求

大多数需求流程要求需求从高级概念（如驱动业务驱动因素、愿景和目标）一直到实现它们的组件部分。对于许多项目来说，这是一个棘手的问题，因为大部分信息都存在于一组异构工具中，例如文字处理器文档、电子表格、图表工具、企业演示工具等。一些项目经理试图通过创建一个电子表格来解决这个问题，该电子表格作为所有不同信息的寄存器，但该文件的管理占用了相当多的项目资源，并且该文件几乎不可能保持最新状态。使用Enterprise Architect ，可以在一个工具中对所有项目信息进行模型，并在所有元素之间创建易于维护和可分析的跟踪，从组织的使命声明到编程代码级别（如果需要）.

在图表中可视化跟踪

无论您是使用图表输入项目的需求，使用基于文本的工具（如规范管理器），还是通过其他工具导入它们，在图表中查看需求跟踪都可以轻松理解它们的关系。可以通过从浏览器窗口拖放元素轻松创建图表，或者使用 插入相关元素”选项自动创建图表。此函数可以配置并用于绘制任意深度的元素图，并且可以限制为选定类型的元素和连接器。它是团队环境中一个方便的生产力工具，即使是对领域和存储库有深入了解的建模者也会对图表中显示的连接感到惊讶。



使用关系矩阵

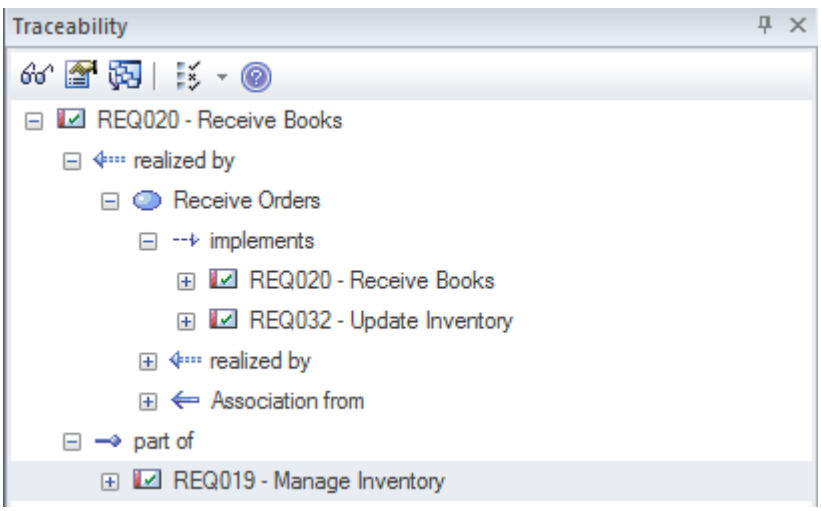
关系矩阵提供了一种可视化需求与其他元素之间，甚至不同层次或类型需求之间的关系的替代方式。一些利益相关者更喜欢电子表格形式的需求及其关系视图是很常见的，而关系矩阵提供了一种无需借助图表即可呈现关系的绝佳方式。在用例驱动的需求方法中，使用案例被称为实现一个或多个需求，这些关系可以在关系矩阵中直观地显示出来。使用案例列表将出现在矩阵的一个轴上，需求将列在另一个轴上。 A存在关系，则在行列交叉处显示标记，表示特定用例实现了需求。元素之间的关系可以通过关系矩阵创建或删除，并且可以随时保存和重新打开矩阵或保存到 CSV 文件，以便可以在电子表格中打开。还可以创建包含关系矩阵的文档，为无法访

问模型的人提供有用的交流工具。

Source \ Target	REQ011 - Manage User Accounts	REQ012 - Provide Online Sales	REQ013 - Manage Deliveries	REQ014 - ShoppingBasket	REQ015 - Process Credit Card Payment	REQ016 - Add Users	REQ017 - Remove User	REQ018 - Report on User Account	REQ019 - Manage Inventory	REQ020 - Receive Books	REQ021 - List Stock Levels	REQ022 - Order Books
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

使用可追溯性窗口可视化跟踪

虽然图表和关系矩阵允许建模者查看需求元素之间的跟踪，但存储库的这些视图的创建者可能故意从视图中省略了元素。例如，图表不需要显示特定利益相关者拥有的所有要求。但是，可追溯性窗口将关系元素之间的完成和完整的视图。无论元素在浏览器窗口中的位置如何，都会显示元素关系。



使用关系窗口可视化跟踪

建模者经常选择在图表上隐藏一个或多个关系，以使图表更易于理解或隐藏细节。关系窗口是一个有用的打开窗口，因为它将显示图表中元素之间存在的所有关系，指示它们在图表中是可见的还是隐藏的。

Relationships

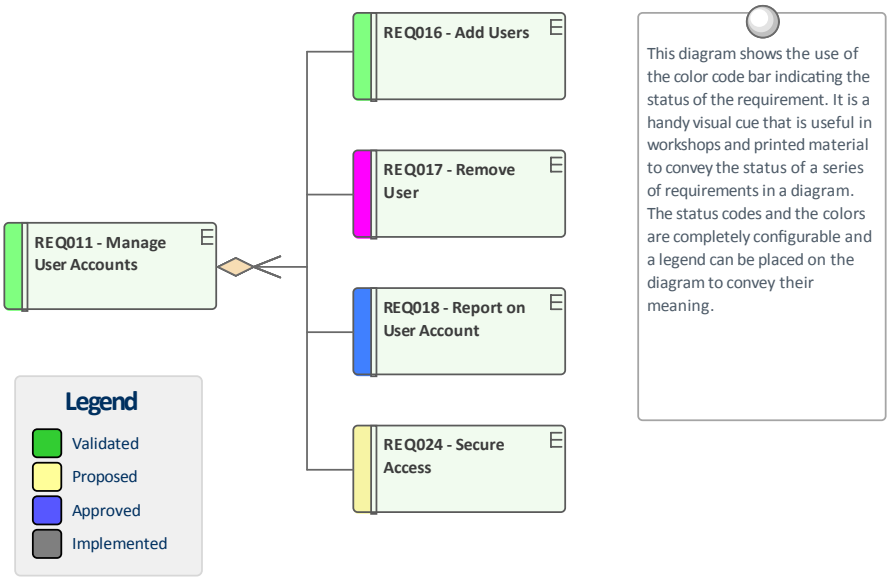
Relationship	Source	Target	View
Abstraction	REQ014 - Shopping Basket	Provide customers with a competitive ...	Visible
Realization	Add To Shopping Basket	REQ014 - Shopping Basket	Visible
Realization	View Shopping Basket	REQ014 - Shopping Basket	Hidden
Aggregation	REQ014 - Shopping Basket	(REQ012 - Provide Online Sales)	

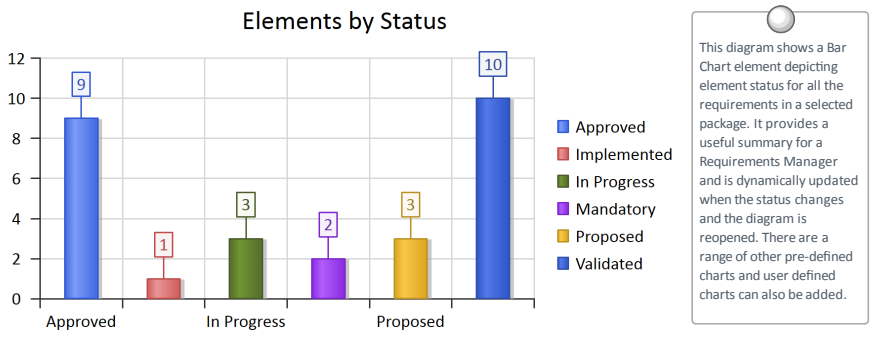
如果关系已隐藏在图表中，则可以通过在图表的 属性”对话框的 连接器”页面上选择 显示关系”选项使其可见。

追踪需求

需求状态是它在需求开发过程中所处位置的基本指标。例如，状态为 已提出"的需求表明它们还没有准备好并且可用于开始开发工作。 Enterprise Architect有多种工具可以状态、分析和管理状态，首先是可以为每个需求分配一个状态，并且状态代码列表是完全可配置的。状态方便地显示在需求列表视图中，包括使用规范管理器时。还有一组预定义和可扩展的仪表板图表和图形，可用于获得需求状态和其他属性的引人注目的可视化表示。

跟踪需求的工具

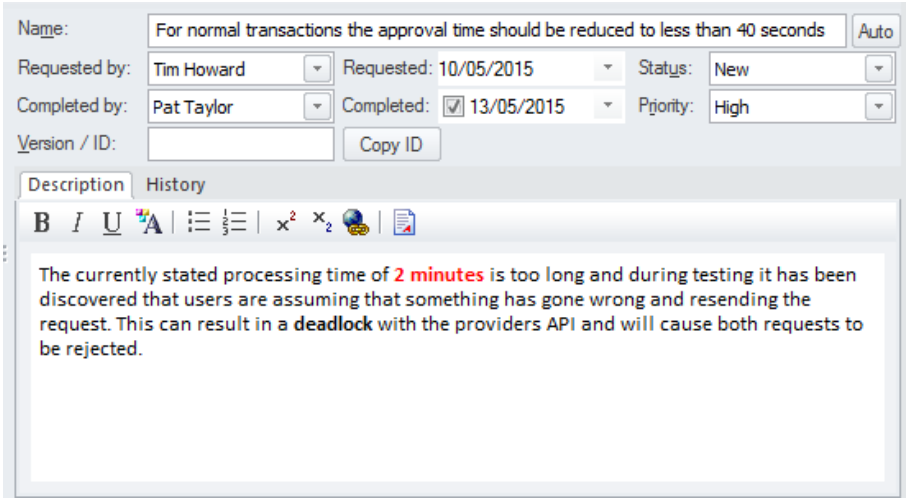
工具	描述
状态码	状态代码是可应用于任何元素（**包括需求**）的受控状态列表。 Enterprise Architect带有一个预定义的代码列表，但是可以配置该列表，并且可以更改和删除列表中的代码，还可以添加新代码。状态需求在图表中显示为元素侧面的颜色编码带。 
仪表板图表和图形	仪表板图是一种扩展的图表类型，允许创建高质量的图表和图形，以视觉上引人注目的方式显示存储库信息。可以创建任意数量的图表和图表，并且可以从存储库包层次结构中的任何级别获取数据。 Enterprise Architect带有一个预先配置的图表和图形的工具箱页面，但可以根据存储库中的任何信息创建新图表。

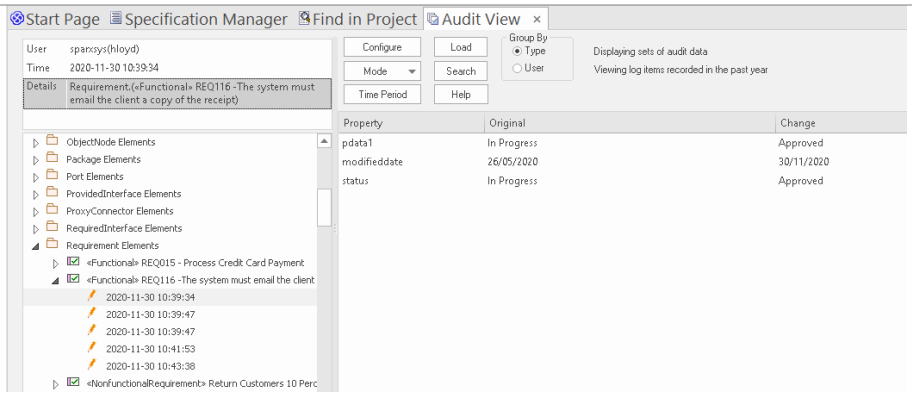
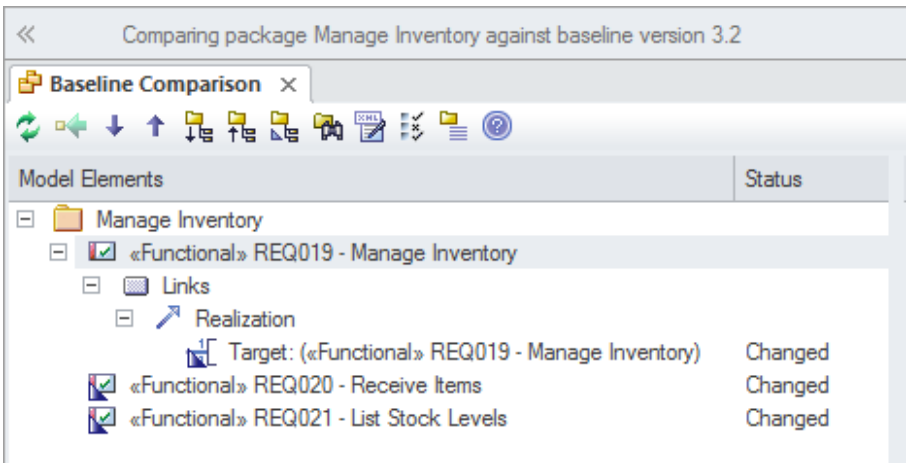

Elements by Status
12
10
8
6
4
2
0
9
1
3
2
3
10
Approved
In Progress
Proposed
Approved
Implemented
In Progress
Mandatory
Proposed
Validated
This diagram shows a Bar Chart element depicting element status for all the requirements in a selected package. It provides a useful summary for a Requirements Manager and is dynamically updated when the status changes and the diagram is reopened. There are a range of other pre-defined charts and user defined charts can also be added.

管理不断变化需求

在项目的规范和解决方案阶段，需求不可避免地会发生变化，并且大多数需求管理过程都有某种类型的机制来接受这些变化。通常，会为解决方案团队指定和整理一组要求来实施；任何后续更改都指定为更改请求。无论使用的流程有多严格，都会发生需要与更改请求一起管理的无意更改。 Enterprise Architect是一个复杂的需求管理平台，具有一系列工具来协助需求经理。更改请求可以在维护窗口中进行管理，它允许记录和描述所请求的更改，以及请求的人以及请求的时间done 以及完成更改的人。可以使用许多特征发现和分析意外更改，包括审计、基线版本和版本控制；这些工具有一些重叠的特征，可以单独使用，也可以一起使用。内置的安全系统还将通过允许建模者有意锁定模型中的包和元素来帮助防止对模型的无意更改。

管理不断变化的需求的机制

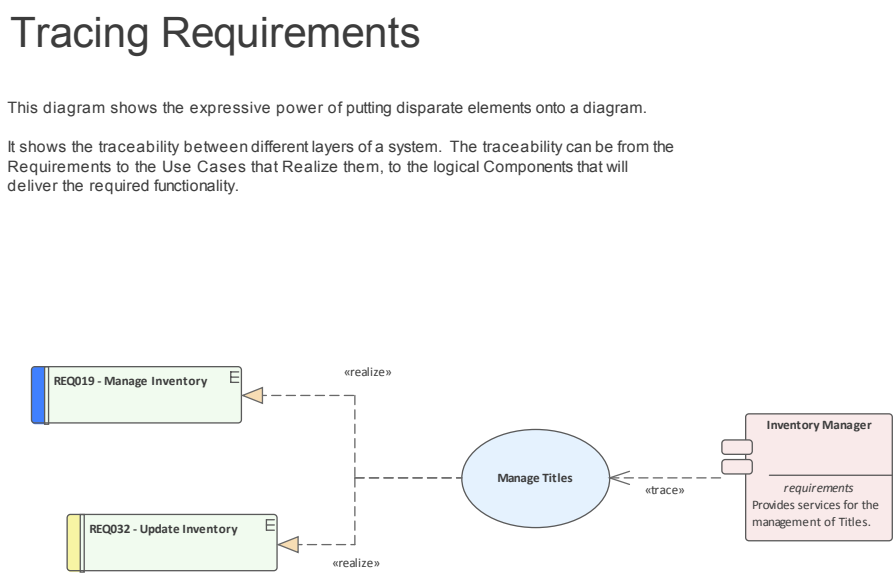
机制	描述
元素改变任务和努力项	对需求的修改可能会在不经意间发生，但更常见的是有意更改以响应各种因素，例如利益相关者修改他们的需求、业务变化或对问题了解甚少。可以使用许多工具来获取无意的更改，但是可以使用更改项来分配故意的更改，这可以针对每个元素进行记录。一旦分析了变更的影响，就可以创建任务来指定实施变更需要做什么，工作量可以使用需求工作量项来分配工作量。 
审计	审计是一个内置工具，启用后会自动记录对存储库的更改。它有许多不同的模式和视图，并且可以配置以协助管理需求。它可以跟踪模型中的更改、更改的人员和更改的时间，显示之前和之后的视图。因此，如果它的状态将被记录为与审计功能重叠的基线需求，但与基线更新或记录工具不同的是，每次更改都被记录在谨慎的变化中。相比之下，没有基线审计将不会对当前模型的基线进行比较，只是比较已发生的变化。一旦确定了更改，就可以使用关系矩阵等工具来确定影响。

	
版本控制	版本控制可以在Enterprise Architect中实施，以管理对任何包的更改和修订，包括需求包。一旦实施对需求的更改将被记录下来，需求分析师将能够查看以前的版本并在需要时回滚到这些版本。这个工具特征和审计和使用之间有一些基线。该功能与审计的区别在于，审计只记录变更，而不允许您恢复到以前的版本。版本控制和基线之间的区别是建模者必须有意创建基线，而版本控制在后台自动记录更改基线是当前记录的间隔变化与记录的要求之间的基线。
基线	用于管理需求基线的多功能机制。任何数量的利益相关者基线都可以被更改为项目的一个重要更改的要求，而这些基线的基线通常可以在一个开发迭代之后或在开始之前在一个开发迭代中创建。当发现差异并且不符合预期或违反从当前项目基线到当前项目的要求时，这些变更一旦恢复，这些基线将模型于评估变更的影响。关系矩阵和元素跟踪可用于确定更改的影响。 

修改的影响分析

当系统的开发已经开始并且需求发生变化时，变化就会产生影响，并且需要确定、理解和管理这种影响。建立对利益相关者和业务驱动因素等上游要素和使用案例、组件、测试案例和源代码操作等下游要素的驱动因素对于确定变更的影响至关重要。 Enterprise Architect具有许多可以帮助实现此目的的功能，包括在图表中可视化跟踪的能力、关系矩阵、可追溯性窗口、元素更改、任务和可用于记录影响和所需内容的工作量项目实施它。

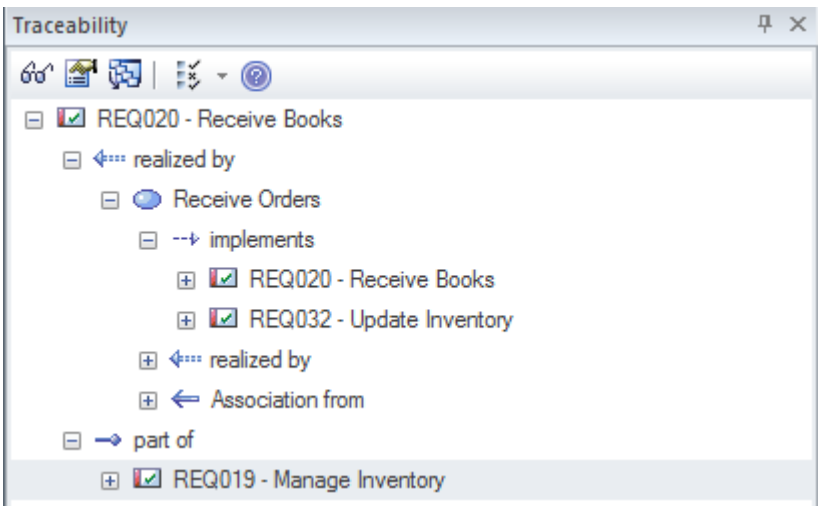
记录和分析变更影响的工具

工具	描述
分析使用需求跟踪	可视化需求及其与其他元素的连接方式的能力是分析变更影响的实用工具。需求通常形成层次结构，当一个需求受到影响时，它通常会对需求的子级产生连锁反应，并且能够在图表或层次结构中可视化这一点非常有用。需求通常也可以追溯到上游和下游流程元素，并且图表提供了一种查看和分析这些连接的方法。插入相关元素函数发现这些连接并自动绘制和布局图表，允许建模者花时间分析影响。 
使用关系矩阵进行分析	通过将需求放在矩阵的一个轴上，将连接的元素放在另一个轴上，关系矩阵可用于可视化需求及其连接。当与可能不熟悉UML等建模语言或更擅长使用电子表格类型视图的人一起工作时，此方法在研讨会中非常有用。可以创建任意数量的矩阵，并且可以存储它们的规格，以便轻松调用它们。

Target / Source	REQ011 - Manage User Accounts	REQ012 - Provide Online Sales	REQ013 - Manage Deliveries	REQ014 - ShoppingBasket	REQ015 - Process Credit Card Payment	REQ016 - Add Users	REQ017 - Remove User	REQ018 - Report on User Account	REQ019 - Manage Inventory	REQ020 - Receive Books	REQ021 - List Stock Levels	REQ022 - Order Books
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

使用可追溯性窗口进行分析

可追溯性窗口是一个方便的窗口，显示存储库中元素的层次结构。它特别有用，因为它无条件地显示元素如何相互关联。其它库的其他视图可以配置为仅显示特定元素以交流想法，而可可追溯性窗口将显示元素参与的所有关系这使得它对于分析变化的影响特别有用。



需求波动率

尽早发布产品和系统的市场压力越来越大，项目团队在越来越短的时间内开发、测试和部署产品的压力越来越大。近年来，需求流程发生了显着变化，以确保在架构师、设计师和开发人员需要时向他们提供稳定、正确和清晰的规范。已经转向迭代和增量过程，这需要为每次迭代提供一组稳定的需求。需求的搅动通常表明问题没有被清楚地理解，利益相关者没有受到损害并且存在未解决的政治问题，范围未定义或业务本身处于波动之中。 Enterprise Architect有许多机制可以用来帮助解决这个问题。 Enterprise Architect没有用于需求波动性（稳定性）的内置属性，但是使用一般标记值的通用UML扩展机制可以创建一个标签来记录这个属性。

注记：内部需求确实有稳定性属性，但外部需求没有。

管理需求波动的机制

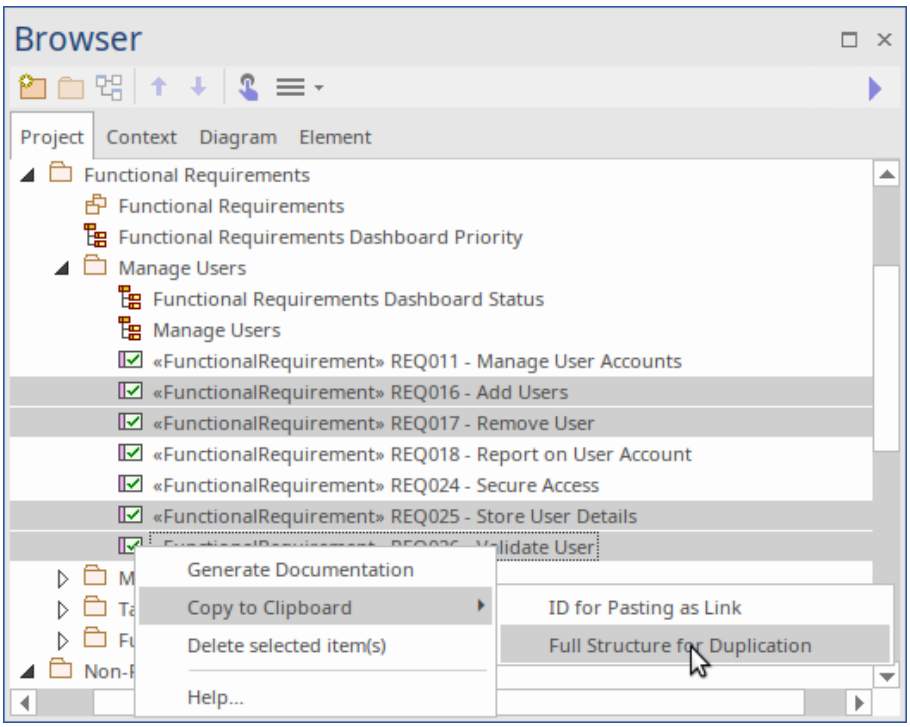
机制	描述
波动性作为一个标记值	Enterprise Architect为需求提供了一系列属性，但可以创建额外的属性来记录其他属性，例如需求的源性或需求的来源。这是使用UML标记值机制实现的，该机制允许包括需求在内的任何元素应用一个或多个标记，表示可以分配值的某些属性。 Enterprise Architect扩展了这个机制，允许建模者创建一个值列表，可以使用预定义的结构化标记值。这允许团队定义他们自己的波动率值列表，例如极端、高、中低、最小。 REQ021 - List Stock Levels *tags* Volatility = Medium *notes* *A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.*
使用基线	基线功能是一个有效的工具，它使用户能够拍摄一个快照或更典型的模型片段，然后随着模型的开发，将模型的新版本与模型进行比较，从而识别任何已更改的内容基线已被采用。基线具有一般，但对需求管理特别有用，如果要求通常被认为已签署或冻结，则对它们的任何更改都必须注册为更改基线模型和工具有一个比较实用程序，可以方便地列出当前基线之间的变化。 Comparing package Manage Inventory against baseline version 3.2 Baseline Comparison Model Elements / Status Manage Inventory «Functional» REQ019 - Manage Inventory Links Realization Target: («Functional» REQ019 - Manage Inventory) Changed «Functional» REQ020 - Receive Items Changed «Functional» REQ021 - List Stock Levels Changed
搜索搅动需求	Enterprise Architect具有复杂的搜索功能，允许用户在选定的包或整个存储库中进行搜索，以找到满足细粒度标准的元素。这可用于通过在指定日期之前搜索修改日期的更改来定位未更改的需求，从而提供稳定的需求列表。或

	者，如果使用标记值设置波动率，则可以定位具有指定波动率的所有元素。搜索功能返回可以位于浏览器窗口中的元素列表；搜索可用作模型视图视图的基础，用于查看易失性或非易失性需求。

重用需求

重用系统开发过程的工件的概念已经写在许多论文和教科书中，但传统上仅限于软件组件。近年来，重用规范（**包括需求**）的概念开始受到关注。当组织创建具有相似特征的产品系列时，或者在行业或领域内有用户社区时，其它尤为重要。安全和监管要求等其他要求类型通常适用于许多业务规则。业务规则和利益相关者的担忧通常也适用于许多项目，最好在单个项目结构之外进行分类。 Enterprise Architect提供了许多复杂的机制来管理项目间元素的重用，包括构建用于重用的存储库、从其他来源导入需求以及可重用资产服务。

需求重用机制

机制	描述
为需求重用构建存储库	设置存储库时，您可以选择为单个项目或多个项目构建它，而这些项目又可以由许多工作程序组织。 Enterprise Architect让建模者完成对存储库结构的控制，允许在项目级别之上设置包，**其中可以添加一些要求**，例如业务、监管和架构需求。
创建基础模型	当您在Enterprise Architect中创建新存储库时，您可以选择使用模型向导创建**空白模型**（首页'从模式结构创建'）来帮助设置存储库，或者您可以使用基础模型作为模板对于新模型。基础模型是存储业务、法规和架构需求、政策和业务规则等可重用资产的好地方。
从其他模型导入需求	在一个组织中拥有多个Enterprise Architect存储库是很常见的，并且很容易从一个模型复制和重用需求（**或任何其他元素**）到另一个模型。这可以通过简单地将选择的需求或整个包从一个存储库复制到另一个存储库，甚至从一个项目复制到同一存储库中的另一个项目来实现。 Enterprise Architect的工作方式与任何其他窗口程序相同，只需将选择复制到剪贴板，然后允许将其粘贴到同一模型的另一个位置或另一个打开的存储库中。 
使用可重用资产服务	可重用资产服务 (RAS) 对于分布式团队特别有用，它为建模者提供了一种简单方便的机制，可以通过共享存储库分发或下载可重用模型结构和元素，**例**

	如需求，可通过专业云服务器连接访问。企业或组织需求级别可以存储在 RAS 中，不同的团队可以将它们合并到他们的模型中，资产的治理通常由资产所有者（注册）在可重用资产服务级别进行管理。

需求关系

Enterprise Architect支持所有 SysML需求关系，这些关系可以在用户界面中的多个不同位置进行可视化，提供了使用这些重要连接器的灵活方式。关系元素（包括需求）在浏览器窗口中不可见，因为当有更有效的查看连接方式时，这会使元素混乱。

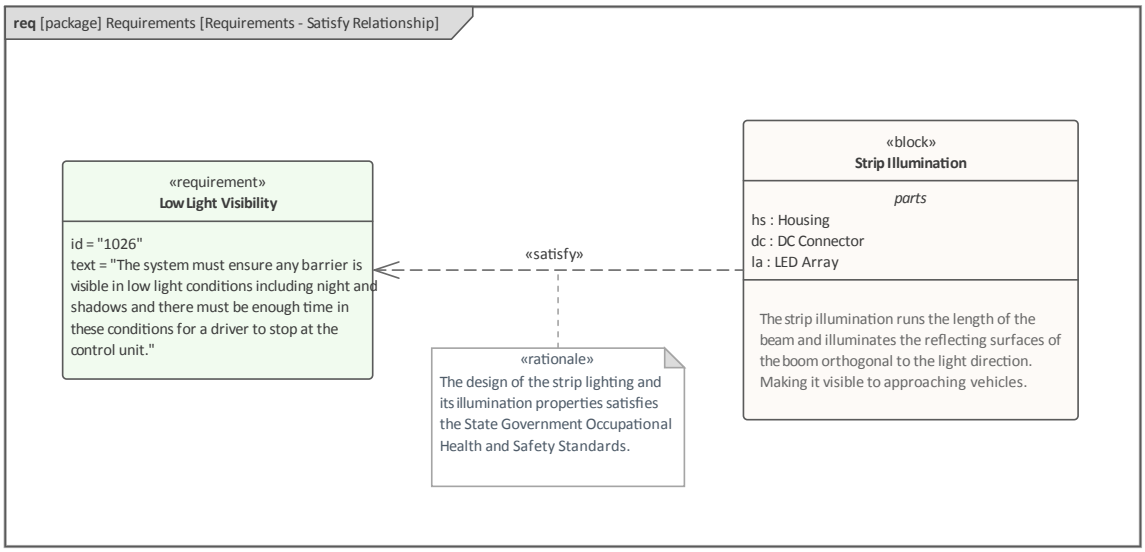
关系窗口对于在浏览器窗口或图表中查看元素时停靠非常有用。由于在浏览器窗口或图表中选择了元素，上下文变化和关系窗口将仅显示所选元素与模型中其他元素之间存在的关系，包括其他需求。

需求与其他元素（包括其他需求）之间的关系可以在包括需求图表在内的任何图表中以三种不同的方式可视化：

- 两个元素之间A连接器
- 需求元素中的A隔间
- 以注记形式附在需求或其他模型元素上A标注符号

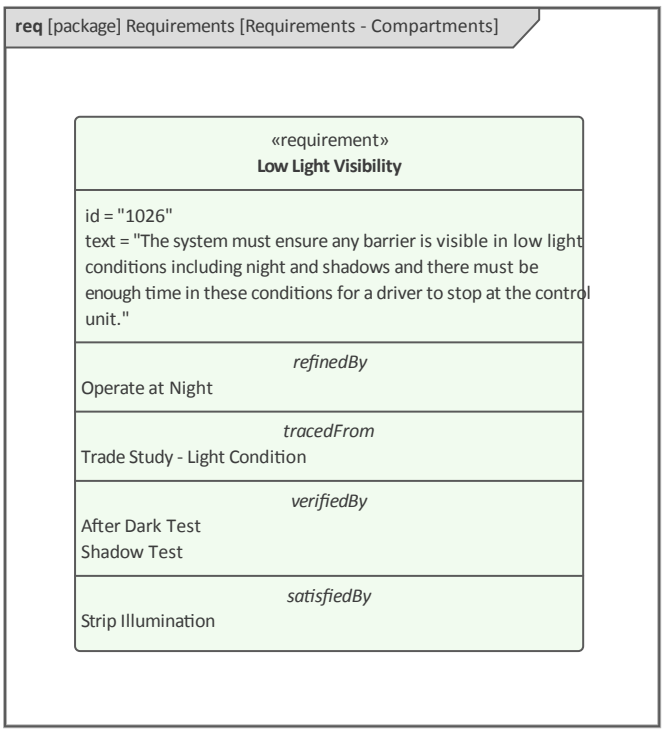
所有三种图表符号都有其用途，并为建模者提供了为特定目的和受众选择适当表示的灵活性。

在图表中可见的两个元素之间绘制关系是最常见的可视化需求关系的方式；虚线是从客户（依赖元素）到供应商（提供元素）。所以在这个例子中，'条码照明'块是客户，它取决于'低光能见度'需求，所以箭头从块（客户）指向需求（供应商）。

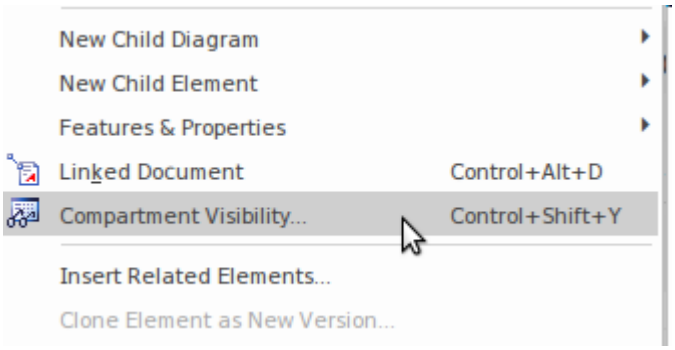


另请注意，在此示例中，建模者已选择显示 Parts 隔间，显示构成块的 Parts 和描述该块的注记隔间。还添加了A基本原理来限定 满足”关系，并解释上下文在标准时间中选择该块。

隔间可用于显示需求参与的关系，这是一种可视化需求关系的紧凑且有用的方式，无需在图表中包含相关元素。

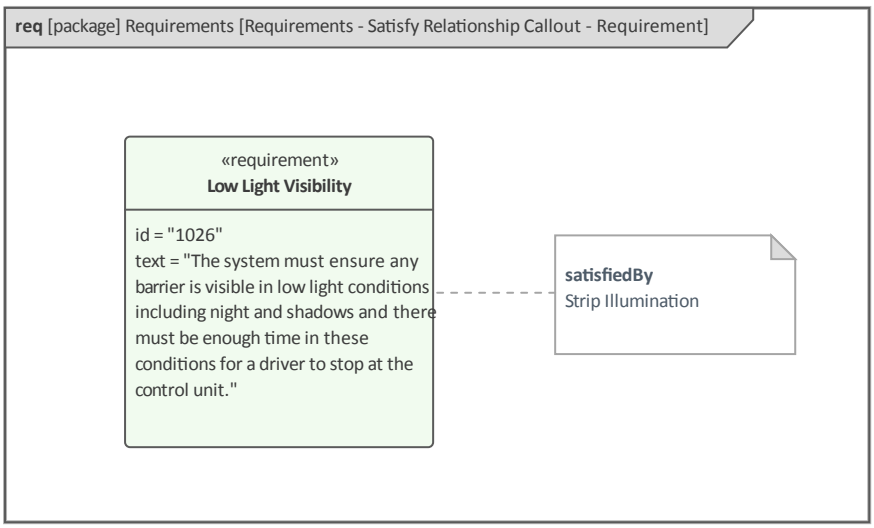


可以为每个图表元素或整个图表配置可见隔间列表，从而提供对关系如何可视化的精细控制。

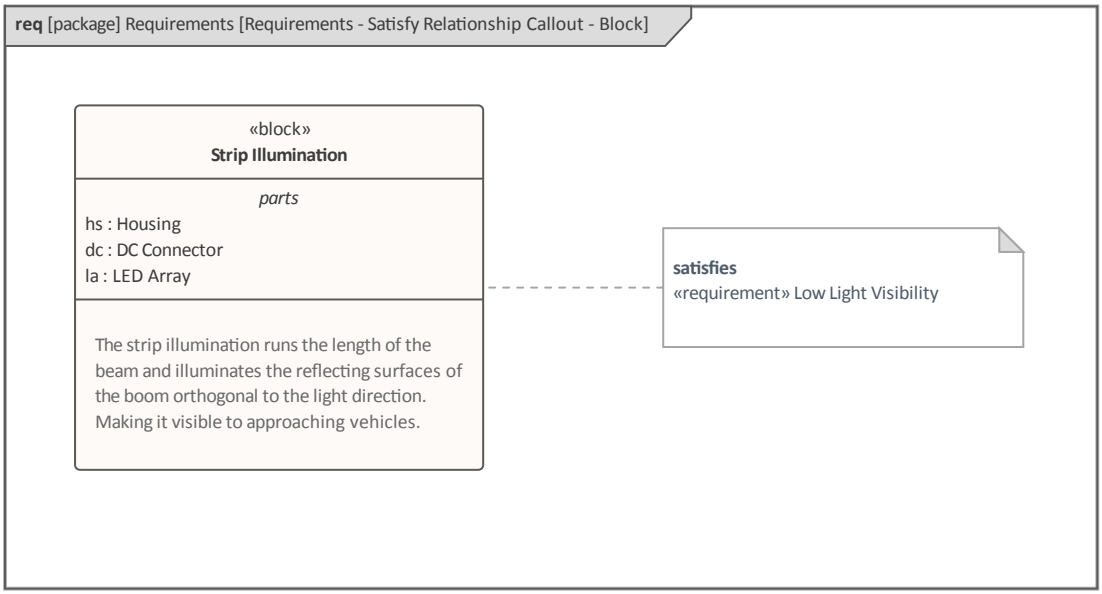


或者，标注符号可用于在注记到需求元素或关系相关的依赖项的元素注释中显示关系。当元素出现在连接器或隔间显示不合适的图表（例如内部块图、序列图或用例图）或建模者认为合适的其他图表中时，此表示法特别有用。需求关系是二元的，意味着它们有两端：供应商和客户。这意味着标注可以附加到需求或相关模型元素，根据关系，可以是块、测试用例、用例或其他模型元素，包括另一个需求。

在这个图中，**建模者的聚焦集中在需求上**，并且在注记中以<<satisfiedBy>>的形式列出了块元素。



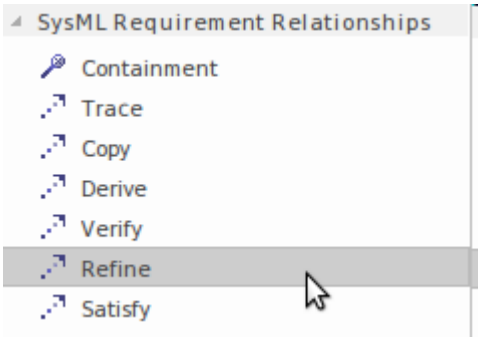
在这个图中，建模者的聚焦已经切换到一个块，需求元素在注记中被列为<<satisfies>>。



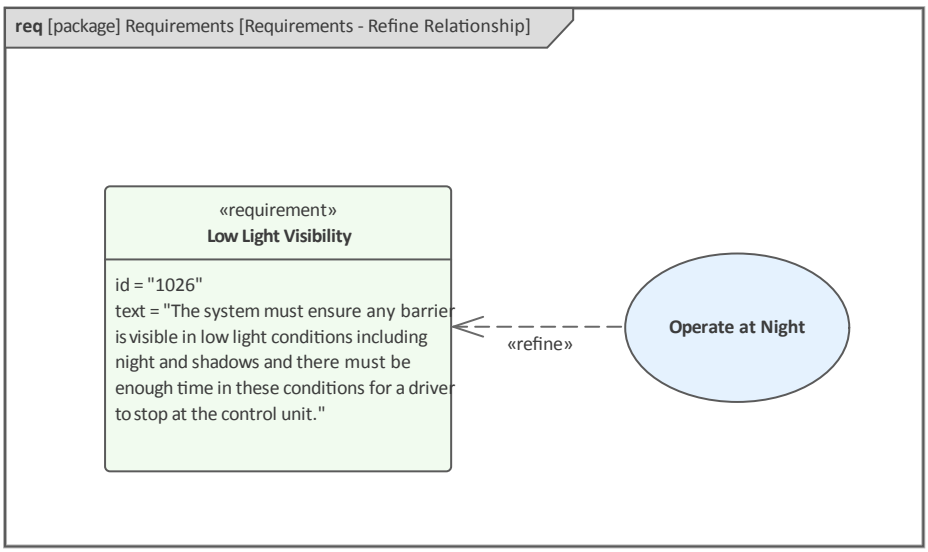
下一节详细说明需求关系，并提供每个关系的示例。

为需求添加细化

细化关系是需求与另一个模型元素之间的关系，它添加细化或附加信息，有助于阐明需求，使其含义更加明显。 Refine关系可从 SysML需求工具箱的 关系”页面获得。



Refine关系可以在需求和任何模型元素之间绘制，例如用例、状态机或活动。模型元素的选择将取决于需求中表达的信息以及建模者或工程师的判断。

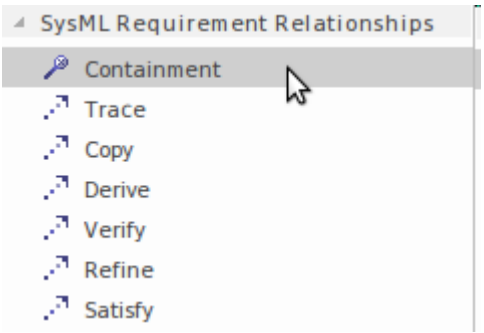


在这个例子中，用例并没有扩展或修饰需求，而是以描述和场景的形式添加细节，使需求的含义更容易理解。

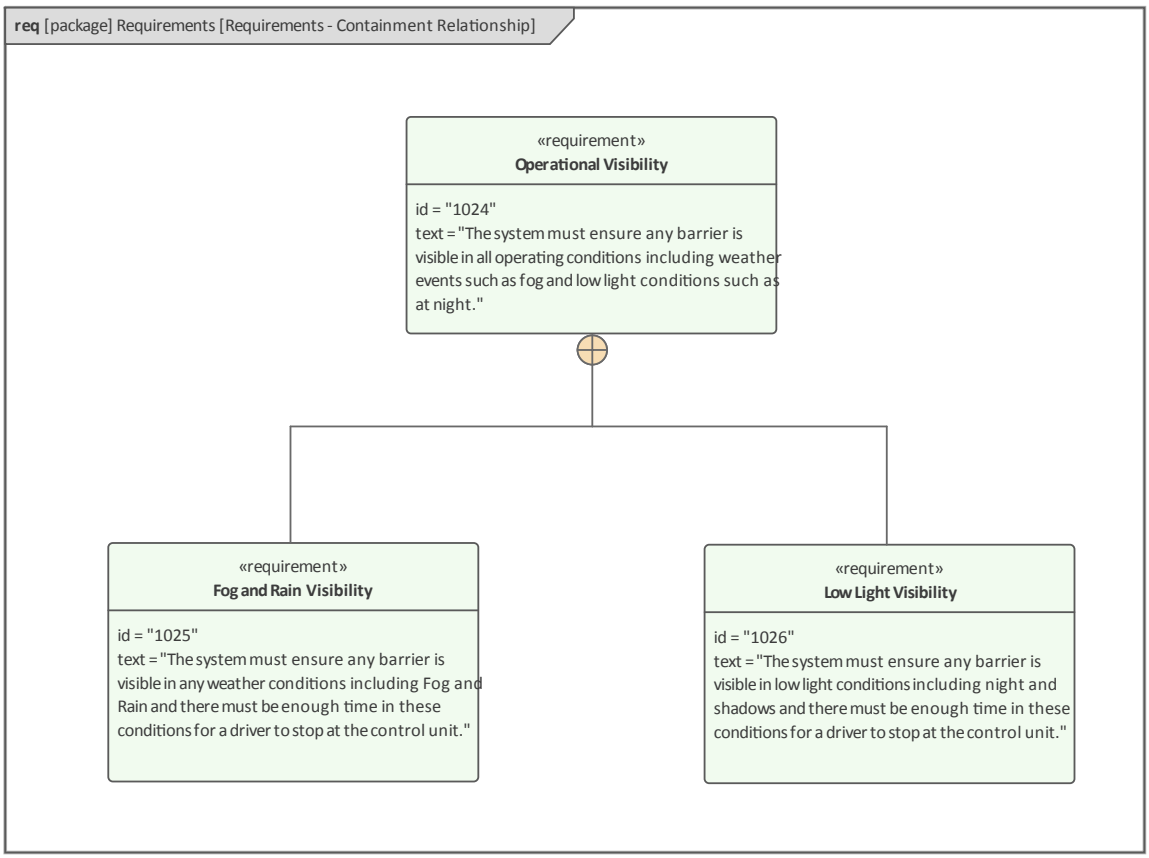
遏制关系

遏制关系的名称很好地描述了它的目的——它用于表明一个或多个需求包含在另一个更高级别的需求中，**或者**由另一个更高级别的需求分组。在为任何中等复杂度的系统建模需求时，它是一个基本且高度使用的关系。A大型系统可能有数千个（如果不是数万个）**需求**，这些需求最好在层次结构中组合在一起。使用Containment关系的替代方法是使用包对需求进行分组。当层次结构中只有两个级别时，或者当您按类型（**例如**利益相关者或物理）对需求进行分组时，**此方法有效**，但在更广泛地使用时会受到限制。

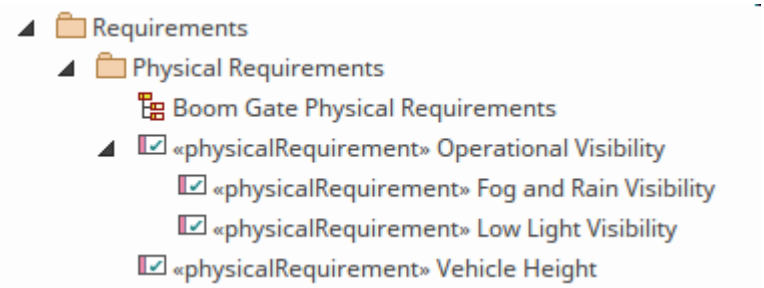
遏制关系可从 SysML需求工具箱的关系页面获得。



此图显示了使用包含关系来显示两个较低级别的需求，这些需求被较高级别的需求 包含”。



需求的包含也可以在浏览器窗口中可视化，**其中包含由嵌套的元素表示**，或者更正式地说，**包含的元素是另一**个需求的子元素。这种平衡对于存储库中的任何元素都是可能的，但对需求具有特殊意义。此图像显示与上图**中相同的需求**，但在浏览器窗口中。

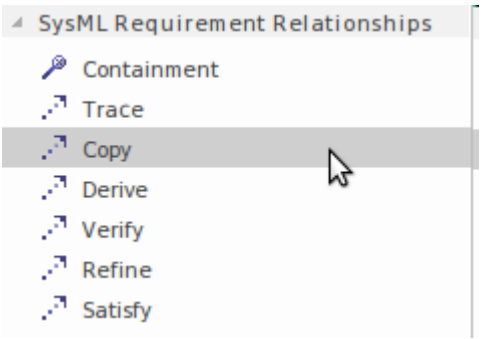


注记嵌套的需求之间的浏览器不会需求关系。事实上，这两种不同的方法可能彼此不同步，因为它们是独立的机制。

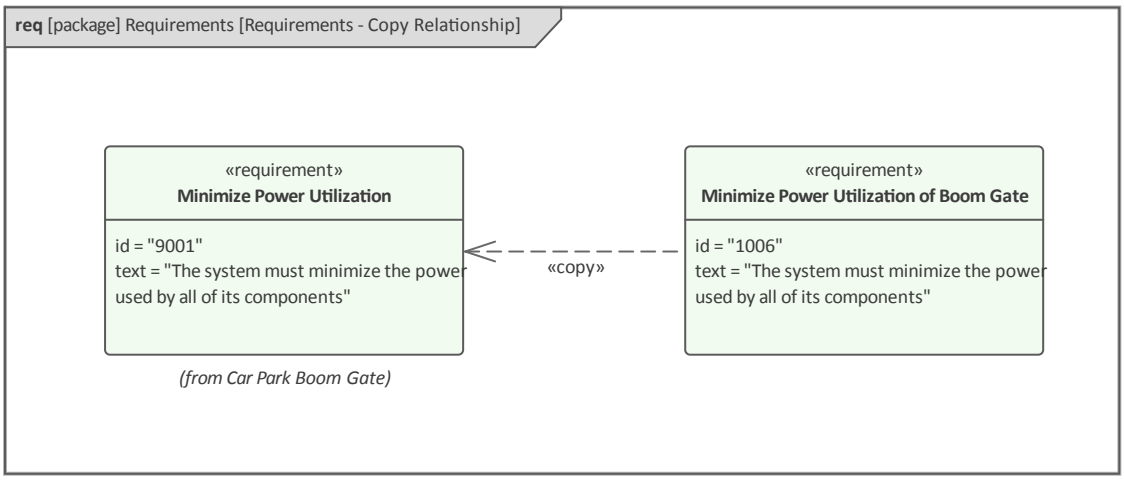
复制现有需求

*Copy*关系是两个需求之间的关系；它用于表明一个需求是另一个需求的副本。关系是 Dependency 的一种类型，由带有关键字 <<copy>> 的虚线表示，带有一个从 Copied（客户）指向 Base需求（Supplier）的空心箭头。鉴于需求获取和管理是昂贵且耗时的活动，并且许多项目经常存在利益重叠，因此重用需求是有用的； Copy关系提供了执行此操作的机制。基本需求通常存储在另一个项目的命名空间中，但将公共（基本）需求移动到位于单个项目级别之上的命名空间被认为是一种好的做法。

复制关系可从 SysML需求工具箱的关系页面获得。



此图描述了已复制以在多个项目中重复使用的电源使用需求。

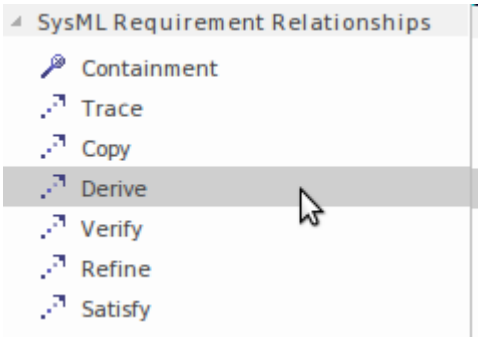


当使用 Copy关系时，新需求被分配一个新的 id，但新需求的文本将是基本需求的只读副本。

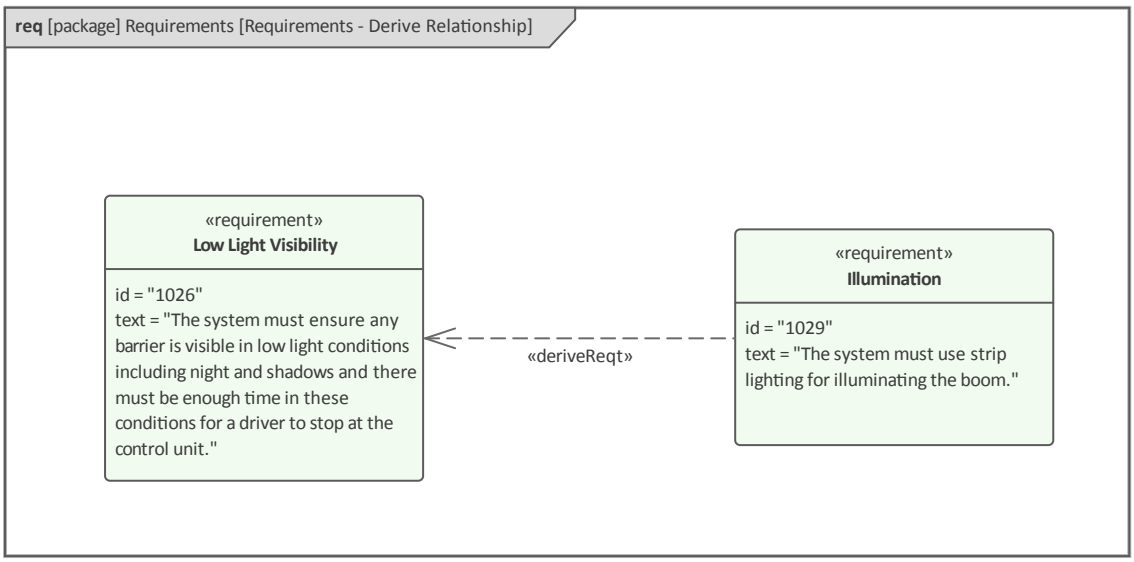
从另一个人那里得到一个需求

派生关系是两个需求之间的关系，用于描述一个需求是基于另一个需求或者是另一个需求的扩展或派生的事实。

派生关系可从 SysML需求工具箱的关系页面获得。



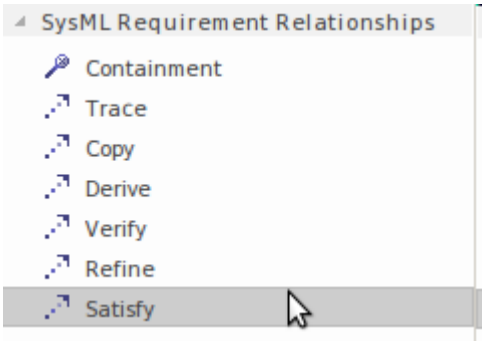
这两种需求通常处于不同的抽象或分辨率级别。低级别规范中A需求可能与更高级别规范中的需求有派生关系。作为调查、阐述或分析的结果，较低级别的需求通常源自较高级别的需求。这种关系的重要方面是，如果关系箭头末端的需求发生变化，则很可能需要重新分析派生的需求。



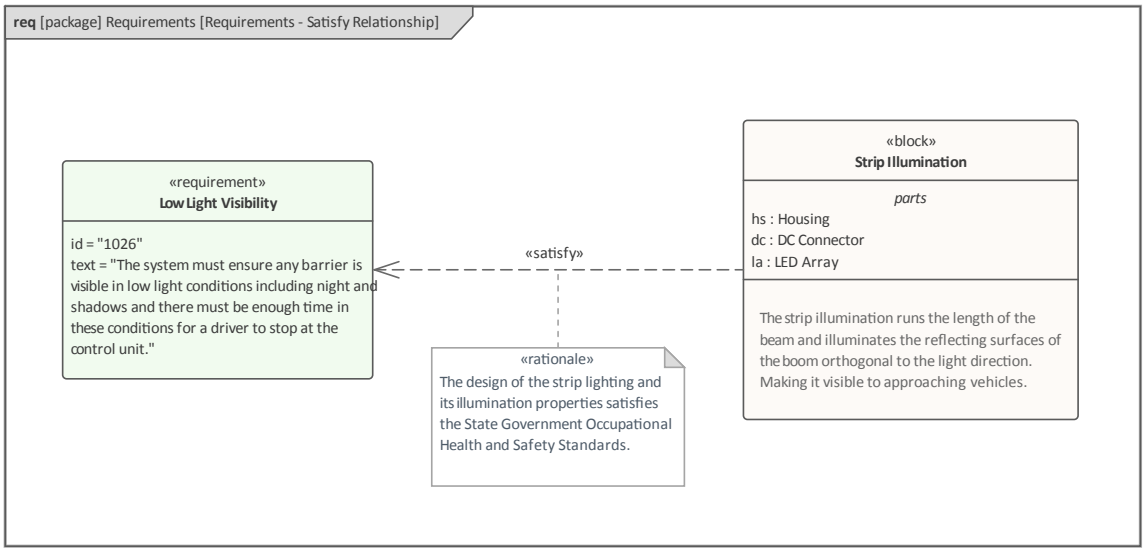
确保需求得到满足

满足关系用于表明架构或设计中的一个或多个模型元素满足需求中表达的概念。它是可能被描述为问题或机会与架构、设计和 - 经验证 - 实施之间的重要联系或桥梁。

满足关系可从 SysML需求工具箱的 关系"页面获得。

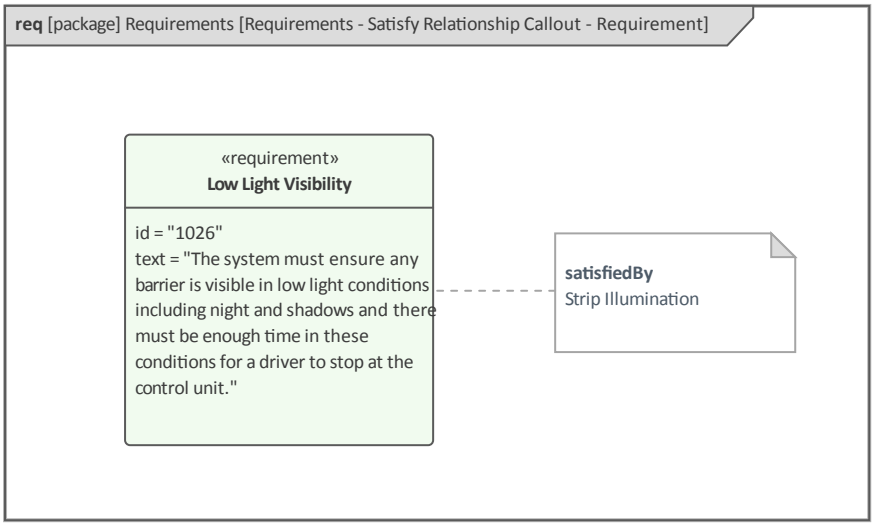


关系是一种依赖关系，由带有关键字<<元素>>的虚线表示，带有一个从设计（客户）指向需求（供应商）的开放箭头。

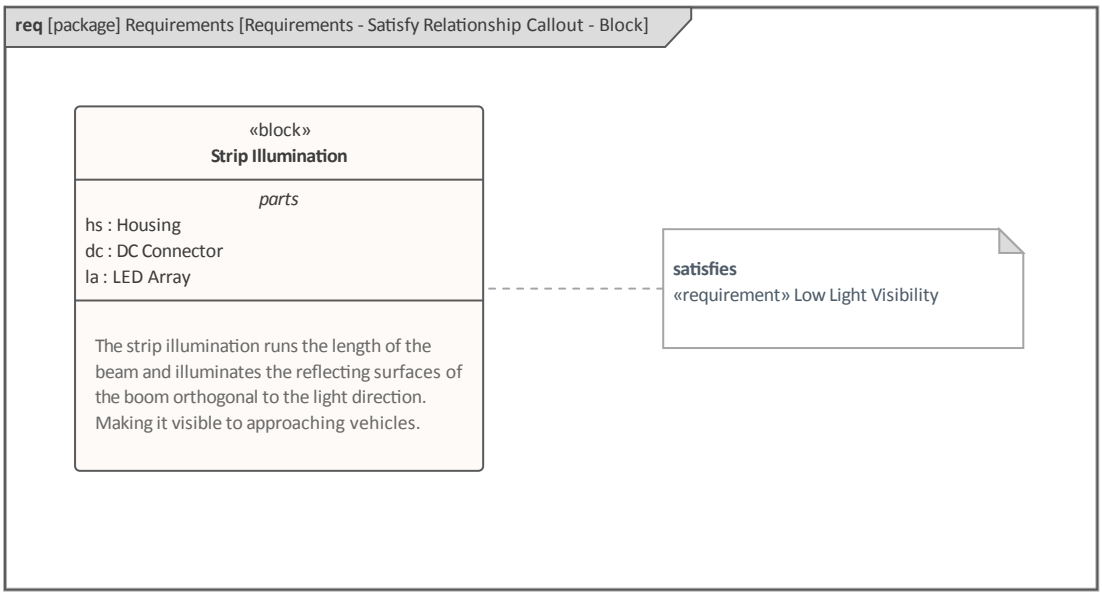


在该图中可以看出，块 条码照明"满足了在弱光条件下吊杆可见度的需求。还添加了一个基本原理，描述了对状态政府法规的遵守情况。

在此图中，满足关系已使用标注符号显示。



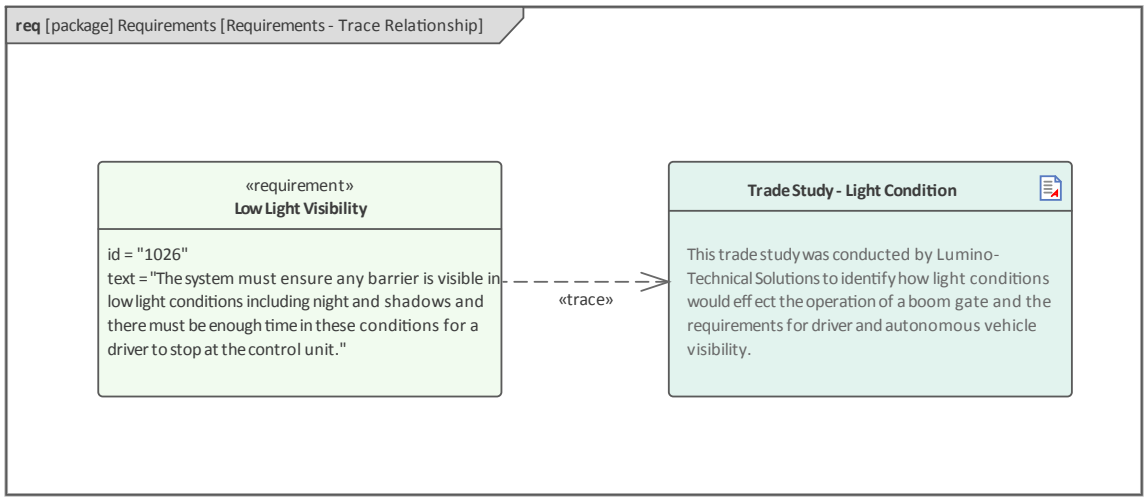
在下图中使用了标注符号，但这次标注中引用了块。



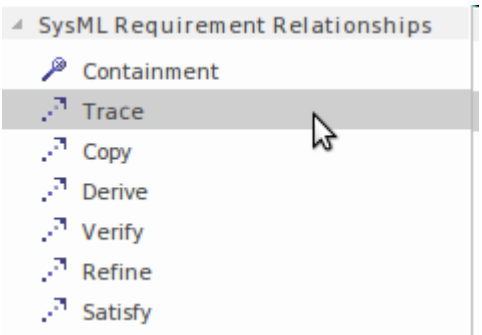
可追溯性到模型元素

跟踪关系是一种一般的、广泛使用的关系，它将需求连接到任何其他建模元素。关系用虚线表示，关键字<<trace>>表示含义；箭头指向一个向上流程元素（在流程早期创建的元素）。

在这个图表中，关系者想要展示一个由文档工件代表的贸易研究代表的需求。该文档可能是用Enterprise Architect编写的，也可能是链接的外部文档。



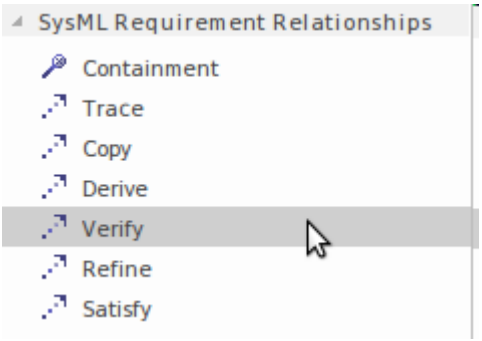
跟踪关系作为一个包罗万象的方法，当建模者想要表明需求与元素有语义关系，而其他关系都不合适时，它很有用。



核实关系

验证关系用于指示需求已被满足。关系是 Dependency 的一种类型，由带有关键字 <<verify>> 的虚线表示，并带有一个从测试用例（Client）指向需求（Supplier）的空心箭头。

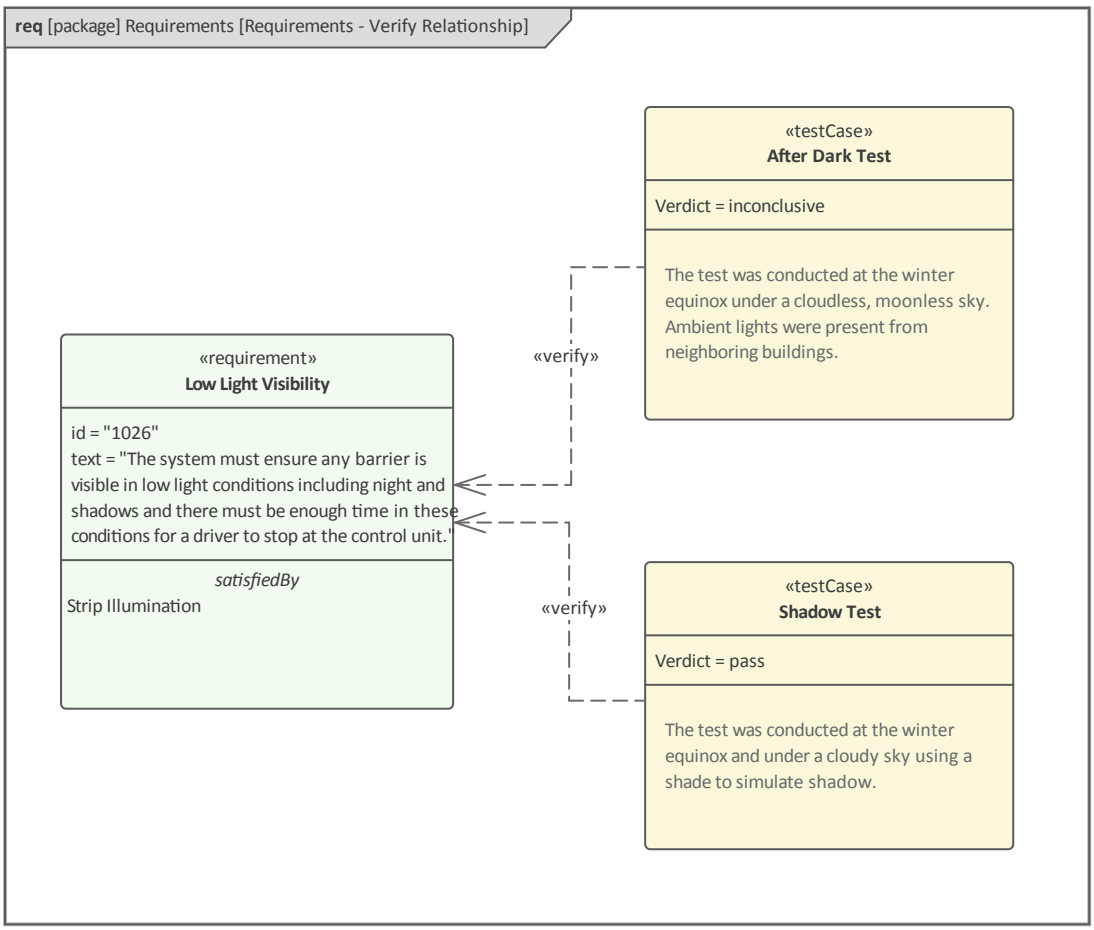
验证关系可从 SysML需求工具箱的关系页面获得。



测试用例可以描述方法或测试过程；它包含一个定义判断（测试结果）的标签，可以是：

- 经过
- 失败
- 不确定的
- 错误
- 用户定义的值

满足关系有一个相关的目的，它描述了设计或系统的哪一部分实际用于执行需求中表达的概念。

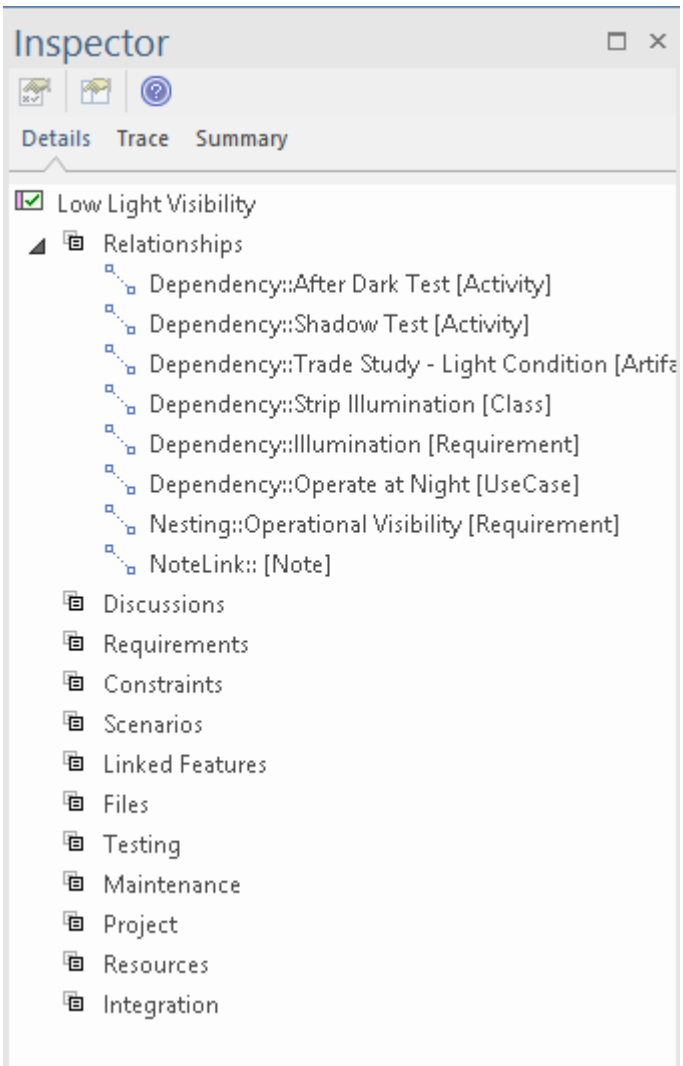


在此图中，需求描述了低光照条件，并且有两个单独的测试案例用于验证需求。建模者选择显示 "satisfiedBy" 隔间，以帮助阐明实施的哪些部分正在接受测试。请注意，每个测试用例的判断是不同的：After Dark测试是不确定的，而 Shadow测试通过了。建模者可以选择A单个图表中显示或隐藏 Verdict 标记。

可视化需求关系

需求与其他元素（**包括其他需求**）之间的关系是基于模型的系统工程的一个关键方面。在许多方面，这些关系是规范和设计或问题和解决方案之间的重要桥梁。这些关系可以在范围广泛的专用窗口和用户界面机制中查看。 Enterprise Architect的新手首先会注意到这些关系并没有明显地嵌套在浏览器窗口的 项目"选项卡中的元素下。这是一个有意识的设计决定，不要让这些关系使 项目"选项卡变得混乱，而是让它们在其他显示中可见，这些显示可以与它们相关的元素同时停靠和查看。

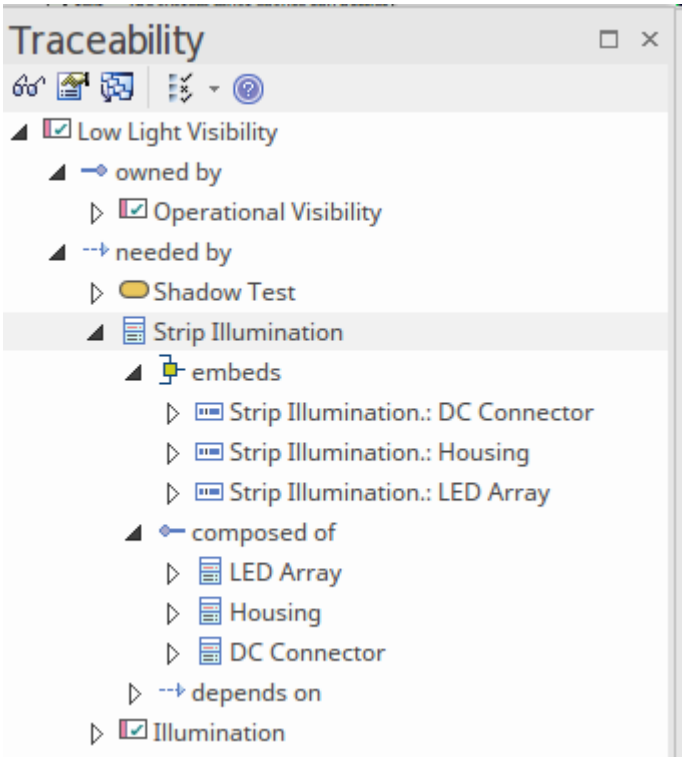
此插图显示了检验员窗口的 详细信息"选项卡，其中 关系"节点展开以显示与 低光能见度"需求相关的所有关系，该需求已在 项目"选项卡中选择。还有一个专门的关系窗口。



还有很多其他可以可视化关系的窗口。有关详细信息，请参阅The Browser Window主题。

可追溯性窗口

可追溯性窗口是Enterprise Architect的一个有用且独特的特征，允许探索关系，以便模型可以有效地遍历元素及其连接图。当您想查看元素如何连接以及连接的元素连接到什么时，打开它是一个有用的窗口。因此，**例如**，在选中 低光可见性"需求时，在门的此模型中，可追溯性窗口将显示它已连接到满足需求的 条码灯光"块。



可设计可追溯性窗口可以从功能区选项 “**元素**>跟踪”中打开。

这种情况下的建模者也可能对 “条码照明”块的结构方面感兴趣，因此可以按照这个元素的关系来发现它的结构关系，走图寻找问题的答案并探索模型。

如果您不关心查看关系，最好不要打开此窗口，因为每次更改聚焦到另一个元素时都必须渲染其内容，并且对于连接良好的元素，这可能需要一些时间，从而增加花费的时间移动模型。有关更多信息，请参阅Traceability Window 帮助话题。

窗口关系

关系窗口是一个有用的窗口，在处理横切关注点时可以打开，这通常是需求的情况。当在窗口、浏览器窗口或图表中选择了一个元素时，关系窗口会显示一个连接器列表，这些连接器要么以所选元素（目标）为目标，要么从元素（**源**）发出。此窗口的另一个有用方面是 “视图”列指示关系是否在当前打开的图表中可见。

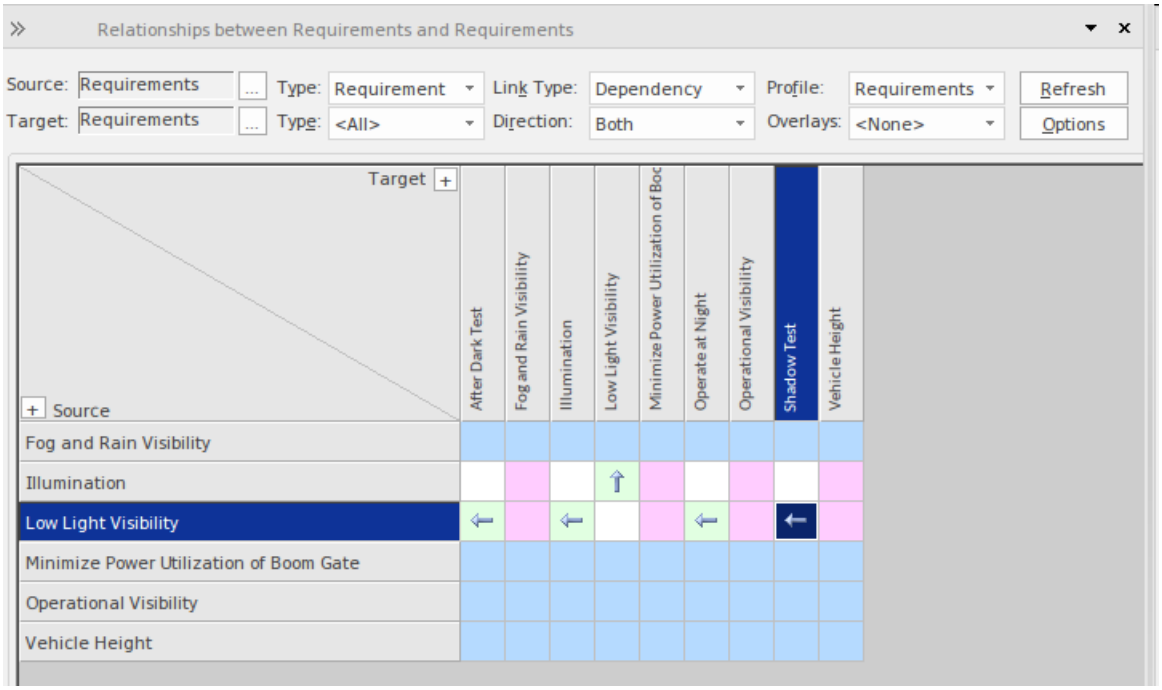
关系窗口可以从功能区选项 “开始>**所有窗口**>属性>职责>关系”打开。

建模者还可以通过从上下文菜单中选择 “在所有图表中查找”选项来定位包含所选关系A所有图表。在该图中可以**看出**，连接 “低光能见度”需求和 “照明”需求的 “deriveReqt”关系存在 两个图中。有关详细信息，请参阅The Relationships Window帮助主题。

Usage Type	Diagram Type	Diagram
Link	SysML Requirements	Low Light Visibility
Link	SysML Requirements	Requirements - Verify Relationship

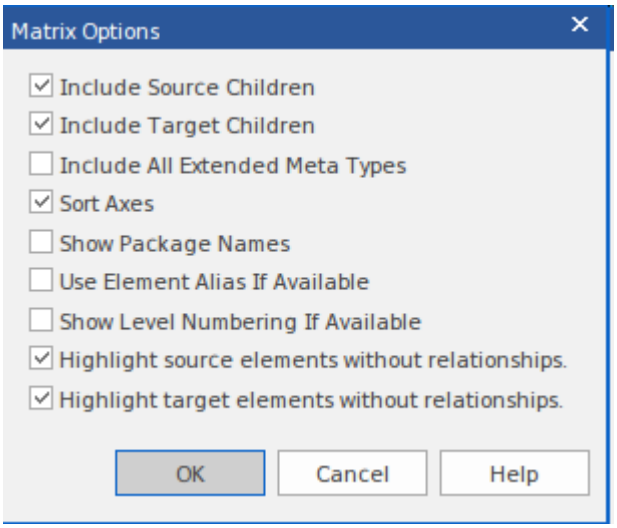
关系矩阵

关系矩阵是一个很好的工具，可以在一个类似于带有行和列的电子表格的界面中可视化任何两个包中元素之间的连接。该工具在与需求一起使用时特别有用，并且允许工程师查看需求与其他元素的关系，**包括其他需求**。



关系矩阵可以从功能区选项 设计>包>包/矩阵"打开。选择当前包是源包、目标包还是两者都有。

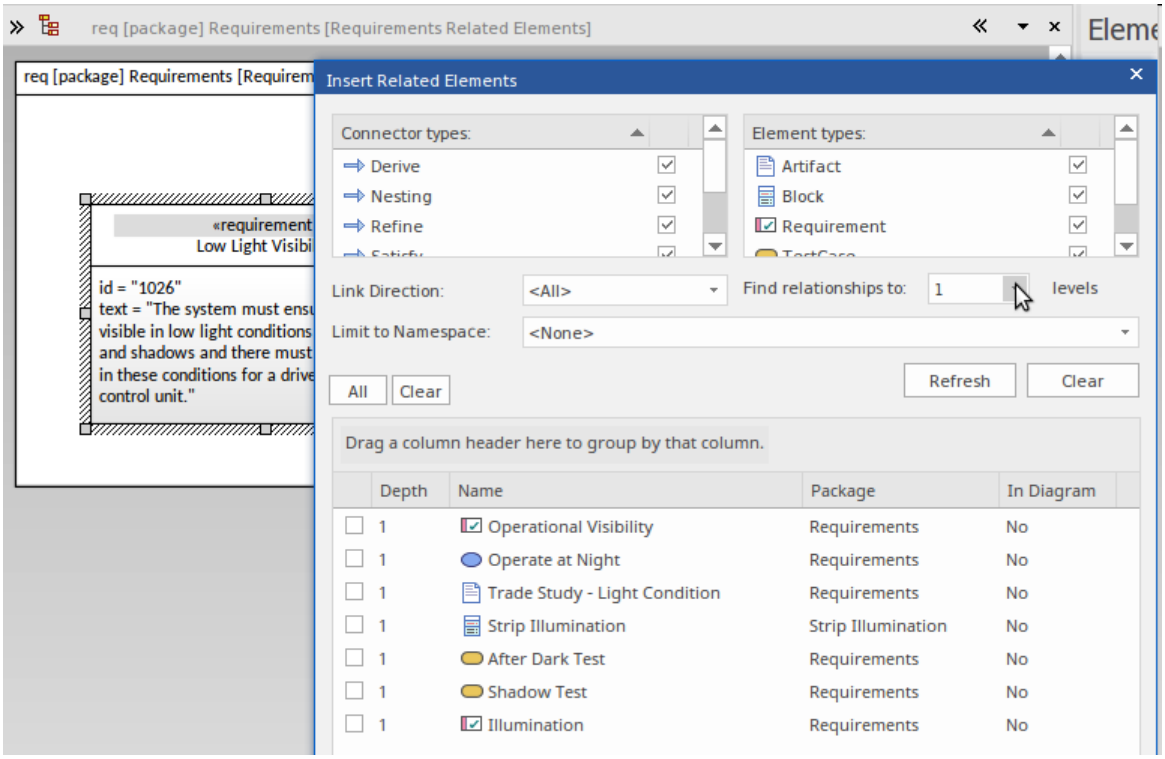
如果存在关系，则在源元素和目标元素的交叉处的单元中将显示一个箭头图标，箭头表示关系的方向。矩阵还可以配置为以单独的颜色突出显示没有任何关系的行或列。可以在 选项"窗口中配置此选项和其他选项，该窗口可从关系矩阵标题中的 选项"按钮获得。



这些选项允许您定制矩阵的显示方式，包括是否对元素进行排序及其名称是否以包名为前缀，以及是否突出显示源和目标元素没有连接的行和列。有关更多信息，请参阅Relationship Matrix帮助主题。

插入相关元素

插入相关元素特征是一种生产力工具，它允许工程师或其他利益相关者人员通过插入中心元素快速构建图表，然后要求工具查找与此元素相关的所有元素，直至任意深度的连接。这有助于工程师有效地探索元素图并创建一个图表，显示存储库中的其他元素如何直接与此中心元素相关以及其他元素如何与这些元素相关。元素和连接器类型可以指定包含或排除，并且可以更改深度以将更多或更少的元素和连接器带入图表。图表可以自动布局，也可以使用图表布局工具重新组织布局，使其更具吸引力或相关性。

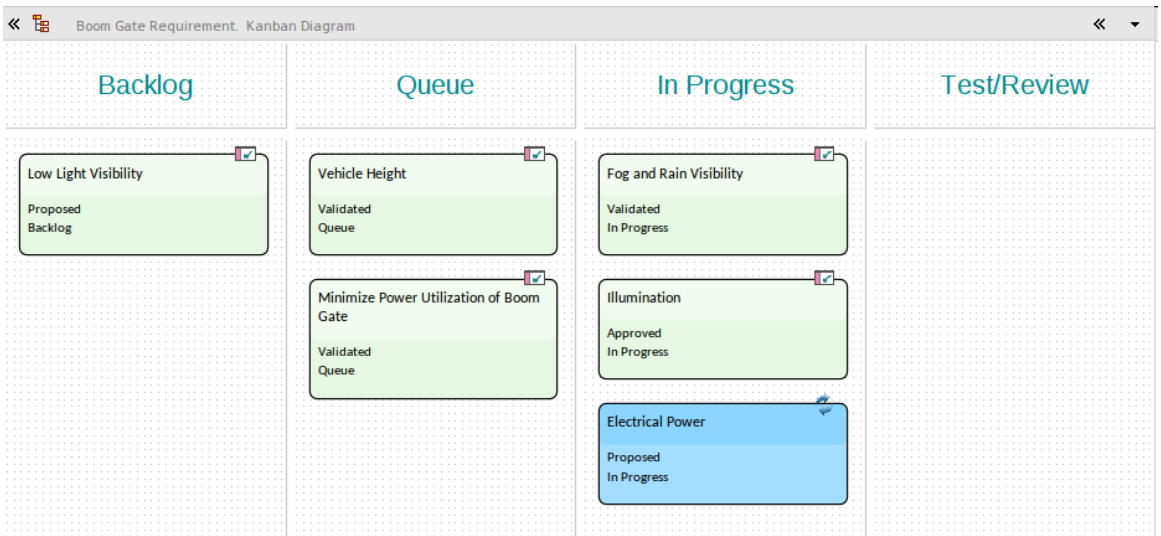


插入相关元素特征通过在图表中选择一个元素然后使用功能区选项 设计>**元素**>**添加元素**>相关元素"来使用。

特征可以用作探索性工具，其中呈现的图表作为查询过程的一部分构建，并在达到其目的后被丢弃。或者，**特征**可用于创建更永久的图表，可以保存和重复使用以进行可视化。无论哪种方式，该工具都将节省工程师的时间，并能够创建准确而富有表现力的图表，这些图表势必会给利益相关者留下深刻印象，否则他们将无法可视化元素之间的连接。有关更多信息，请参阅Insert Related Elements 帮助 话题。

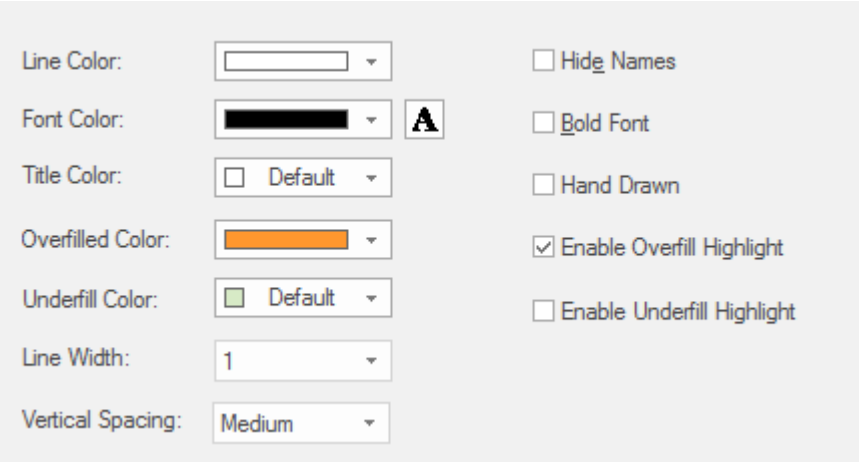
看板图表

看板图可用于在需求（**和其他元素**）通过行业标准或专有流程或方法的各个阶段时将其可视化。看板的概念起**源于** 20世纪 日本后期汽车行业的过程效率分析。从那时起，它已被许多软件和系统社区采用，作为管理从积压或队列到实施的需求的有效方式。 Enterprise Architect有一个完整而实用的看板实现，可以支持许多不同的需求**流程**。

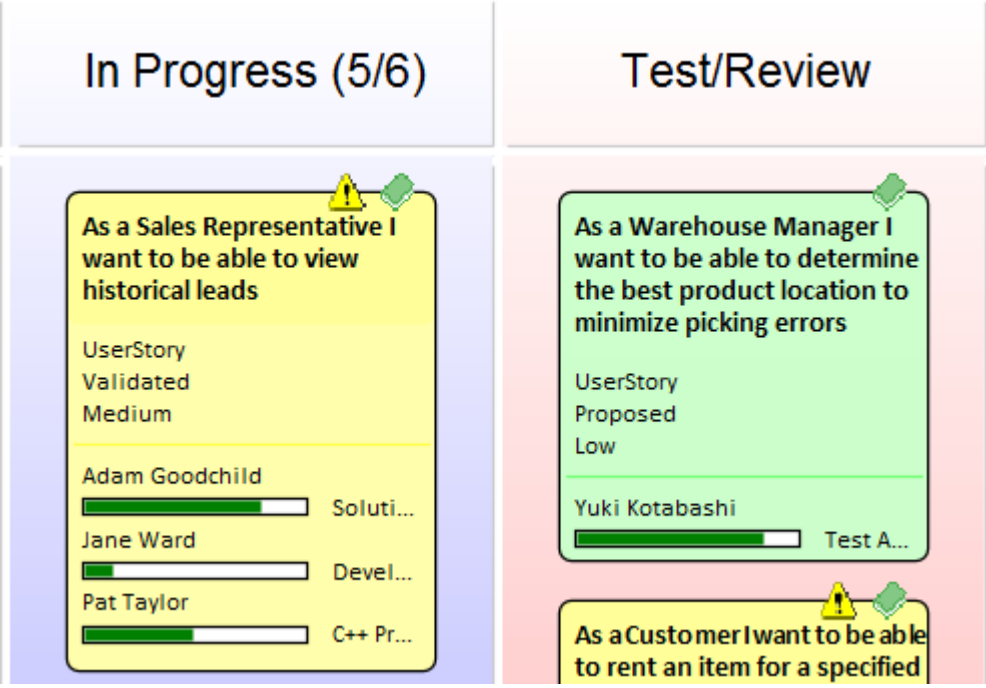


在此图中，我们看到许多列代表需求过程中的各个阶段，允许在列之间拖动元素——通常是从左到右，但有时可以将元素返回到积压中。该图表完全可以由工程师配置，允许配置列的数量及其名称以及图表的广泛其他方

面，包括绑定属性、正在进行的工作限制和颜色等等。



通过应用资源分配并将每个资源显示为显示任务完成百分比的进度条，还可以显示针对特定需求的进度。在这个图中，我们看到了许多元素，其中一个显示了三个资源在同一个元素上工作。有关更多信息，请参阅Kanban Boards 帮助话题。



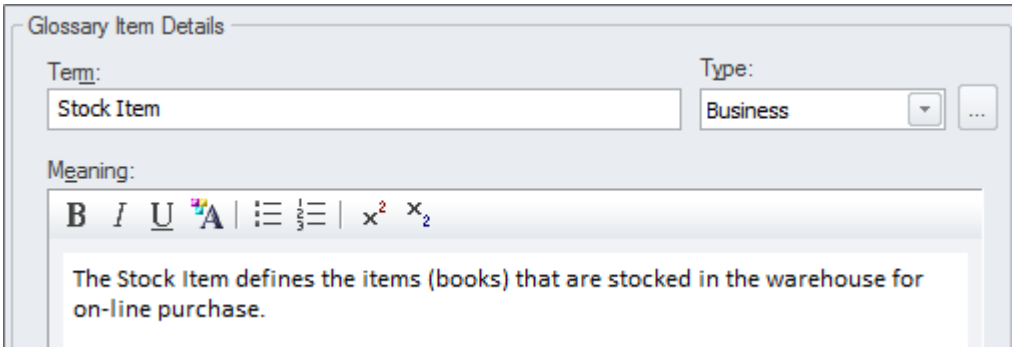
记录需求

A需求工程学科的一部分，通常会生成许多文档，例如软件（系统）需求规范和用例报告。这些可以使用内置模板从需求模型中自动生成。此外，可以使用内置或自定义模板生成范围广泛的其他文档。 Enterprise Architect中的文档功能是高度可配置的，并且可以使用模板系统生成许多报告，但是对于更复杂的报告，有一个称为虚拟文档的功能，它允许发布者模型文档的结构并从任何地方挑选内容在存储库中，将不同的模板应用于文档的每个部分。在模板或文档生成级别也可以应用多种选项，脚本引擎可用于将内容注入文档或生成整个报告。

还有一个自定义文档功能，它允许建模者通过简单地从浏览器中以元素、图表或包的形式拖动内容并将内置或用户定义的模板应用于指定的每个项目来定义动态创建的文档内容将如何呈现。这允许文档被可视化，并且除了从浏览器注入的内容之外，还可以包括任何手工制作的内容或图像。

项目词汇

项目词汇列出并定义A项目或工作计划很重要的术语。项目词汇可以作为一个单独的文档生成，也可以作为一个或多个其他文档的一个部分包含在内。它为重要的项目术语及其含义提供了一个单一的事实；生成新文档时，条款将自动更新。词汇表可以生成为 DOCX 或 PDF 格式，或者生成为可以包含在项目或组织级网站中的 HTML。词汇表允许建模者将术语分类为用户定义的类型，当它们在文档中生成时可以应用样式。

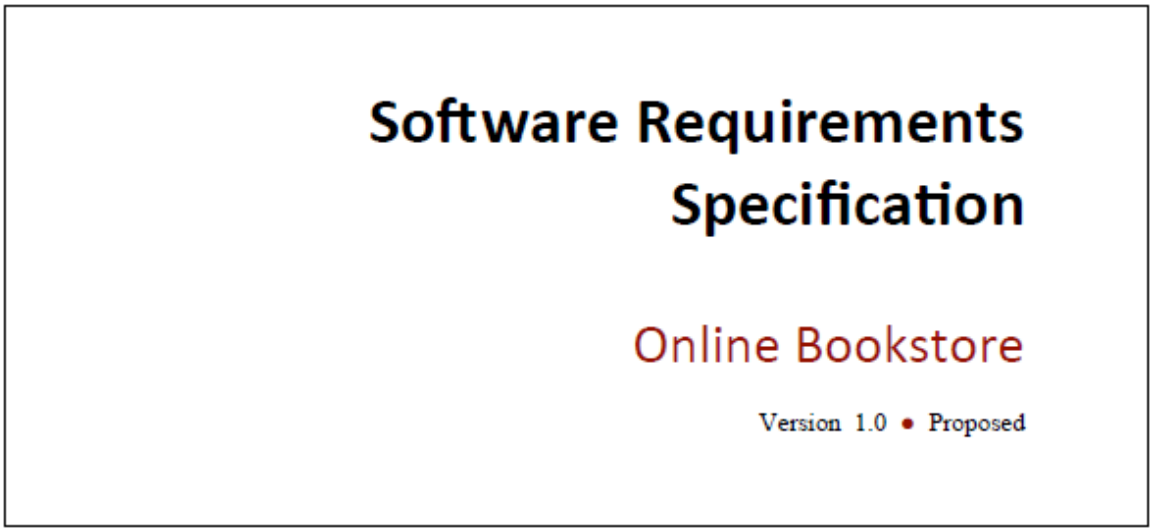


可以从这个功能区位置查看和管理项目词汇：

设计> 词典 > 词汇表 > 词汇表视图

软件需求规范

该文件描述了系统的需求、它在定义条件下的行为以及它必须在其下运行的约束；它通常会被各种利益相关者阅读。有一个内置的需求模板可以用来生成文档，尽管建模者可以自由地创建一个新的模板，可以基于这个模板或从一个空白模板创建。当文档在浏览器窗口中包含来自多个位置的内容时，使用虚拟文档功能是最方便的，它允许用户创建文档的模型（类似于文字处理器中的主文档）有许多部分称为模型文档。这些可以从浏览器窗口的任何地方挑选内容。



使用案例描述用户目标

使用案例最初是由瑞典电气工程师 Ivar Jacobson 设计的，他也是Unified Modeling Language (UML) 开发的重要支持者。使用案例被用作从用户角度表示功能需求的一种方法。据说它们是目标驱动的，因为用例定义了用户在与系统交互时试图实现的目标。 Enterprise Architect全面支持用例图的开发，同时也全面支持用例文本的建模和管理；它有一个用于处理使用案例的独特且高效的工具，称为场景生成器。

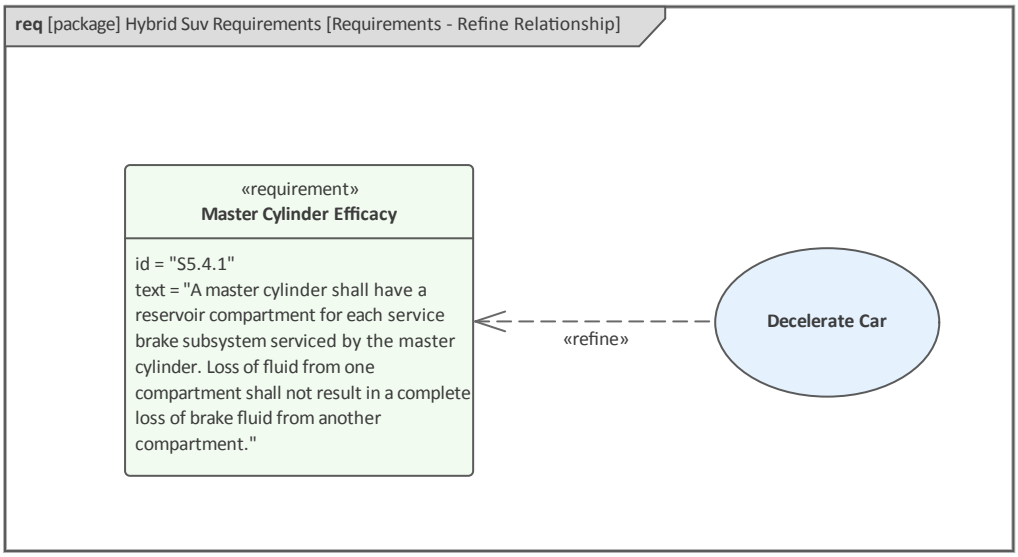
不仅可以对使用进行细节，还可以自动创建行为模型，允许详细的步骤和使用示例的任何用例工具和系统之间的参与者都可以与使用示例的其他部分相关联进行可视化模型

用例是用户故事的近亲，它被用于许多敏捷软件开发技术。最初在瑞典语中创造的术语更自然地翻译为用途，它为该方法提供了更令人信服的解释。

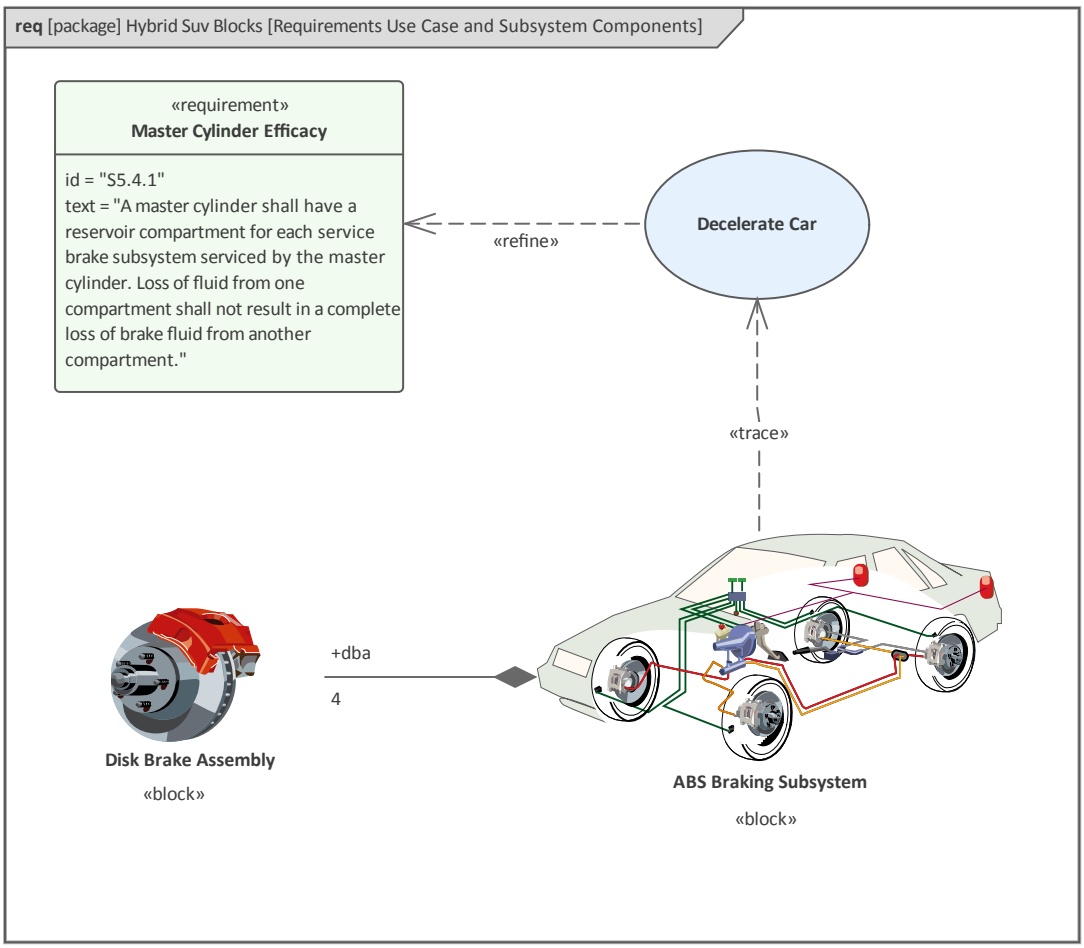
需求和使用案例

用例技术从根本上说非常简单，最初的设计目的是确保从用户的角度编写功能需求。这种观点有助于确保部署的系统适合用途并被不同的用户社区所接受。然而，有大量相互矛盾的文献和同样大量的用于定义使用案例的样式。这导致了混乱和不确定性，并倾向于削弱可以从这种有效且简单的技术中获得的价值。

在软件工程中，许多方法规定使用使用案例作为需求开发的替代方案，因为Unified Modeling Language (UML) 不包括正式的需求元素。相比之下，大多数使用 SysML 的基于模型的系统工程方法结合了使用案例和需求的应用。这是因为 SysML 定义了一个用例和一个需求元素，因此这两个元素可以相互关联并补充系统规范，从而为需求工程和管理的重要学科带来清晰和精确。



在这两个图中，建模者使用了 <<refine>>关系来表示*Decelerate Car*用例提炼或添加额外的解释来阐明需求 *Master Cylinder Efficacy* 。这提供了一种机制来跟踪与使用示例相关联的实施用例组件，以备份到需求并最终到利益相关者。

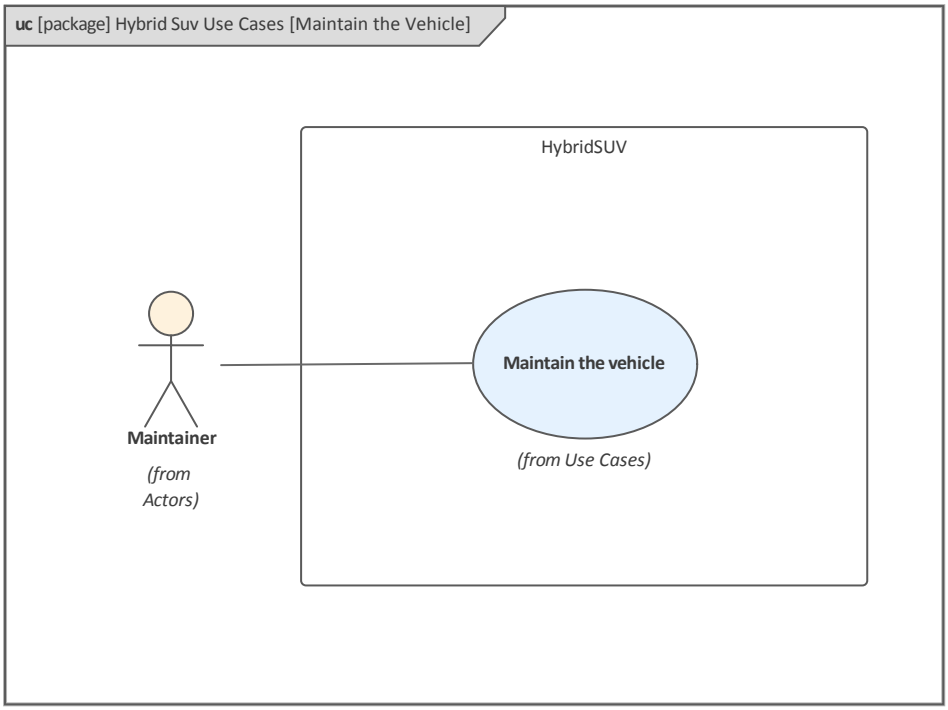

req [package] Hybrid Suv Blocks [Requirements Use Case and Subsystem Components]
«requirement»
Master Cylinder Efficacy
id = "S5.4.1"
text = "A master cylinder shall have a reservoir compartment for each service brake subsystem serviced by the master cylinder. Loss of fluid from one compartment shall not result in a complete loss of brake fluid from another compartment."
«refine»
Decelerate Car
«trace»
+dba
4
Disk Brake Assembly
«block»
ABS Braking Subsystem
«block»

介绍用例图表

用例图是一个简单的图表，它直观地描述了用户对系统或部分系统的目标。这可以解释为 系统为演员提供的价值”。用例图看起来非常简单，只有少量元素：

- 主题
- 演员
- 使用案例

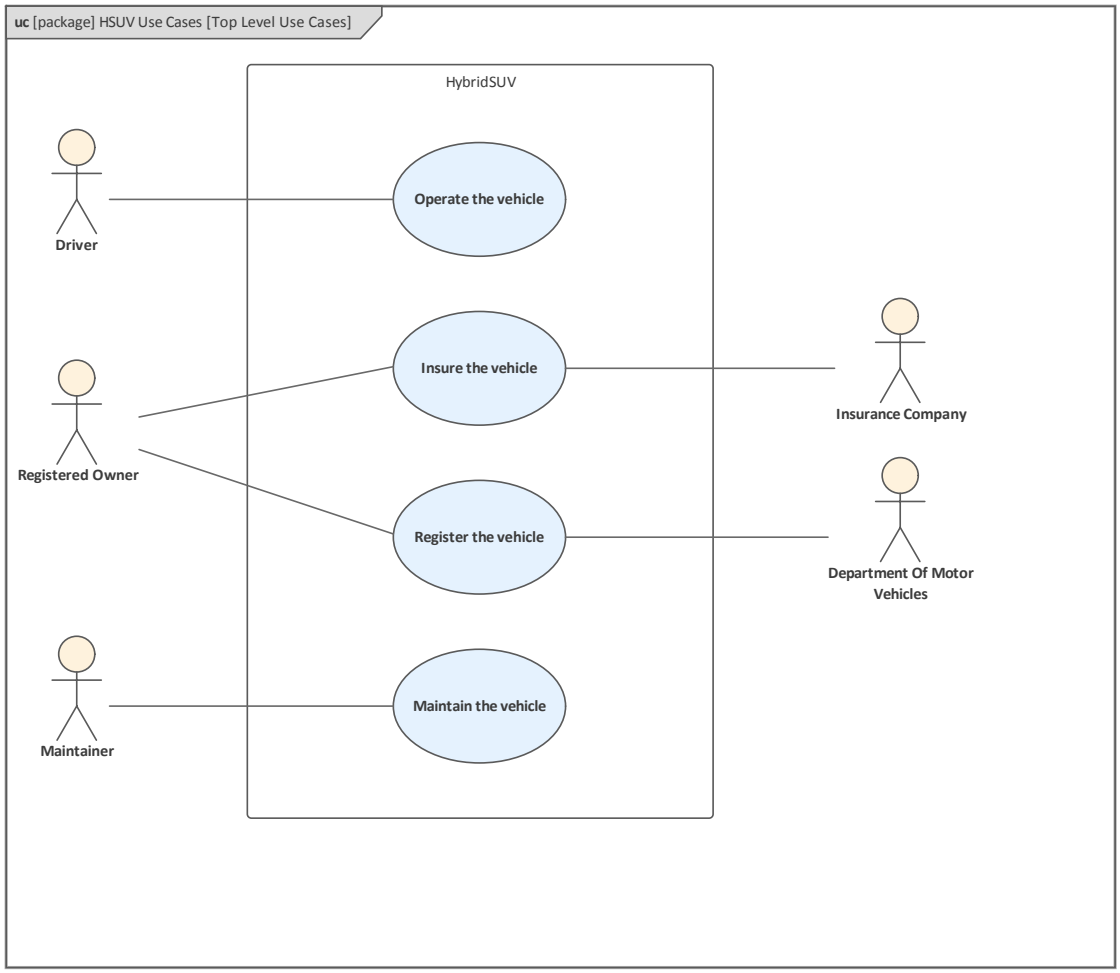
这些是由一系列关系连接起来的。



上下文（边界）为定义提供时间，代表一个系统或系统的一部分；根据定义，Actor 位于 Subject 之外，而使用 Cases 则位于其中。根据定义，通讯路径关系跨越了主题的边界，因为它连接了参与者和用例。同样，关系的数量非常有限，但每个关系在图中都有特定的含义。

- 通讯路径
- 延长（也条件）
- 包括
- 概括

与所有 SysML 元素一样，元素既有图形方面也有文本方面，在使用案例的描述中，通常更强调文本或叙述方面。



可以创建任意数量的用例图来表示用户与系统或系统的一部分的交互。重要的是要理解使用案例旨在描述系统为其用户提供的价值，而不是通过功能分解来分解它们。毫无疑问，这是新手建模者最常犯的错误，导致该技术可以获得的深远好处减弱。

用例模型可以通过一种称为 构建用例模型"的机制进行美化，该机制将重复的文本分解出来，对角色和使用案例进行分类，并指定扩展点。这一机制将在本章后面讨论。有关更多信息，请参阅SysML Use Case Models帮助主题。

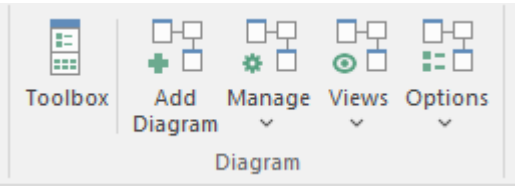
创建一个用例图表

可以从用户接口中A多个位置创建用例图，方法是选择：

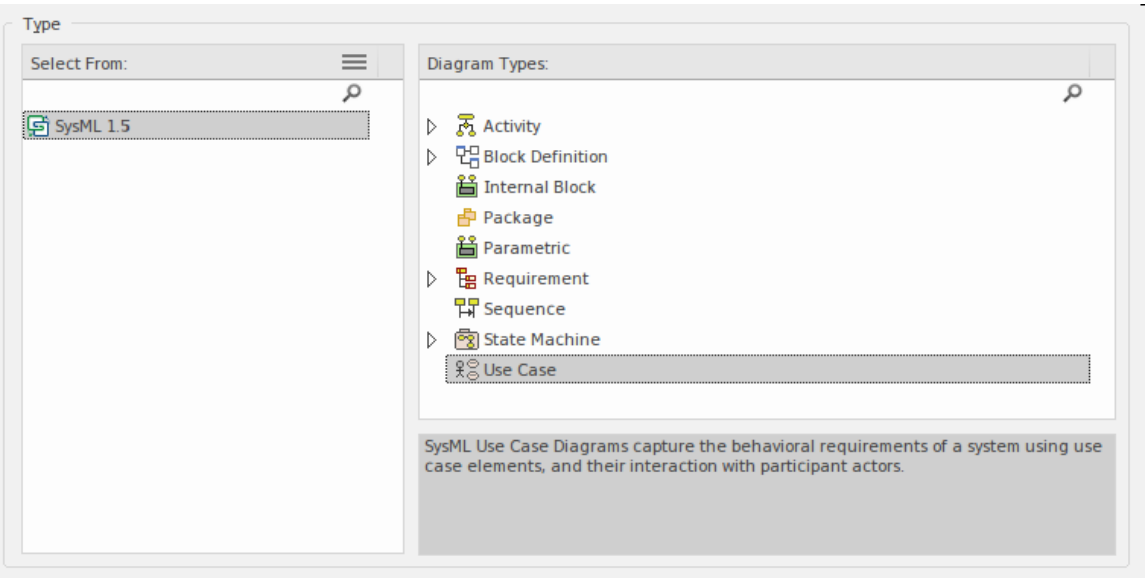
- 设计ribbon - 在图表面板上添加图表图标
- 浏览器窗口工具栏 -新的图表图标
- 浏览器窗口上下文菜单-图表图形

我们将使用设计功能区创建一个用例图。首先，在浏览器窗口中选择您希望用例图所在的位置。与所有图表一样，这可以是包或元素，但通常将用例图插入包中。在浏览器窗口中选择包位置后，选择：

设计>图表> 添加图表



选择此选项将打开*New*图表对话框，允许您命名图表；该名称最初默认为包含图表的包或元素的名称。选择 SysML 透视图并选择 SysML 版本后，将显示图表列表，允许您选择使用用例图表。单击确定按钮在浏览器窗口中选择的位置创建一个新的用例图。将打开图表视图，允许您开始添加元素和连接器来描述系统将为其用户提供的价值。 Enterprise Architect还将显示工具的 用例"页面，工具箱包含图表规范定义的元素和关系，以适用于构建用例图。如果需要，除了始终可用的公共元素和公共关系工具箱页面之外，还可以打开任意数量的其他工具箱页面。



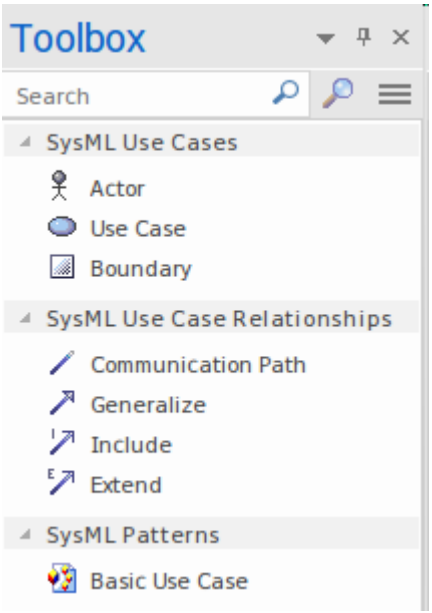
与用例图一起使用的最重要的元素和连接器是：

元素

- 尊重参与者- 用于描述用户扮演的角色
- 用例——用于描述系统为其用户提供的价值
- 边界显示系统的范围（通常每张图一个）

连接器

- 通讯路径- 用于连接 Actor 和使用案例
- 这 - 用于两个概括之间或两个使用案例之间
- 包括 - 用于在 to用例之间重用场景步骤
- 扩展 - 用于用额外的细节修饰一个用例

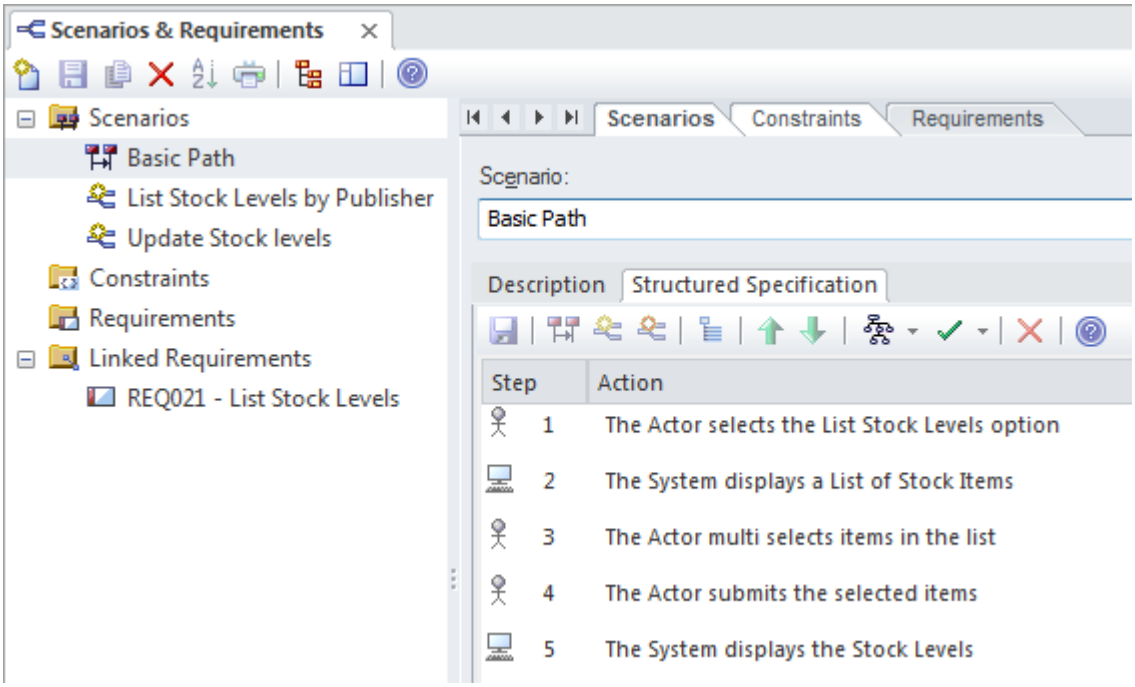


可以通过将元素从工具箱到图表视图视图中来将元素添加到图表中。以边界元素开头被认为是一种良好的做法，它应该被适当地命名以描述被用例图建模的系统、子系统或实体。将名称留空，或者给它起一个名称，使读者无法清楚地了解正在建模的系统或系统的一部分，可能会导致对图表的误解。通过在边界中添加并适当调整大小，可以添加 Actor 和使用案例 - 定位外部的边界和内部的使用案例。下一步是在 Actor 和使用案例之间添加通讯路径关系，从而定义 Actor 从系统中派生的值。

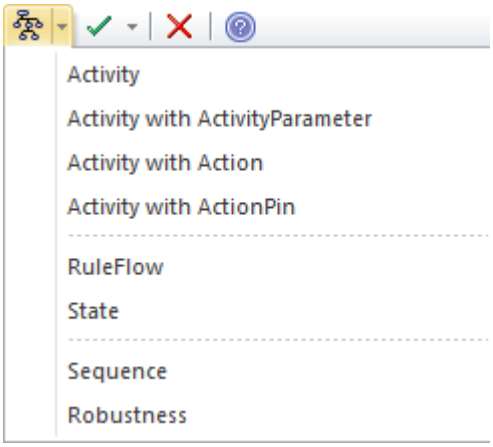
一旦创建了基本图，并且进一步揭示了领域和系统的知识，就可以使用包含、扩展和附加行为的关系来构造或概括图。警告新手不要过于随意地使用这些关系，任何使用功能分解的尝试都会削弱用例模型的价值，该模型的描述有意宽泛，以便利益相关者获得系统提供的服务的 10,000 米视图，子系统或实体被建模。

认识场景构建器

有很多工具可用于处理使用案例，但没有一个比场景生成器更重要和有用。这个独特的工具弥合了传统上在文字处理器中完成的工作或将用例图与其角色和使用案例分开的孤立工具与场景步骤之间的差距。



场景构建器还提供了一种机制，可以直接从场景步骤中自动生成行为模型，从而允许架构和设计的元素与各个步骤相关联。



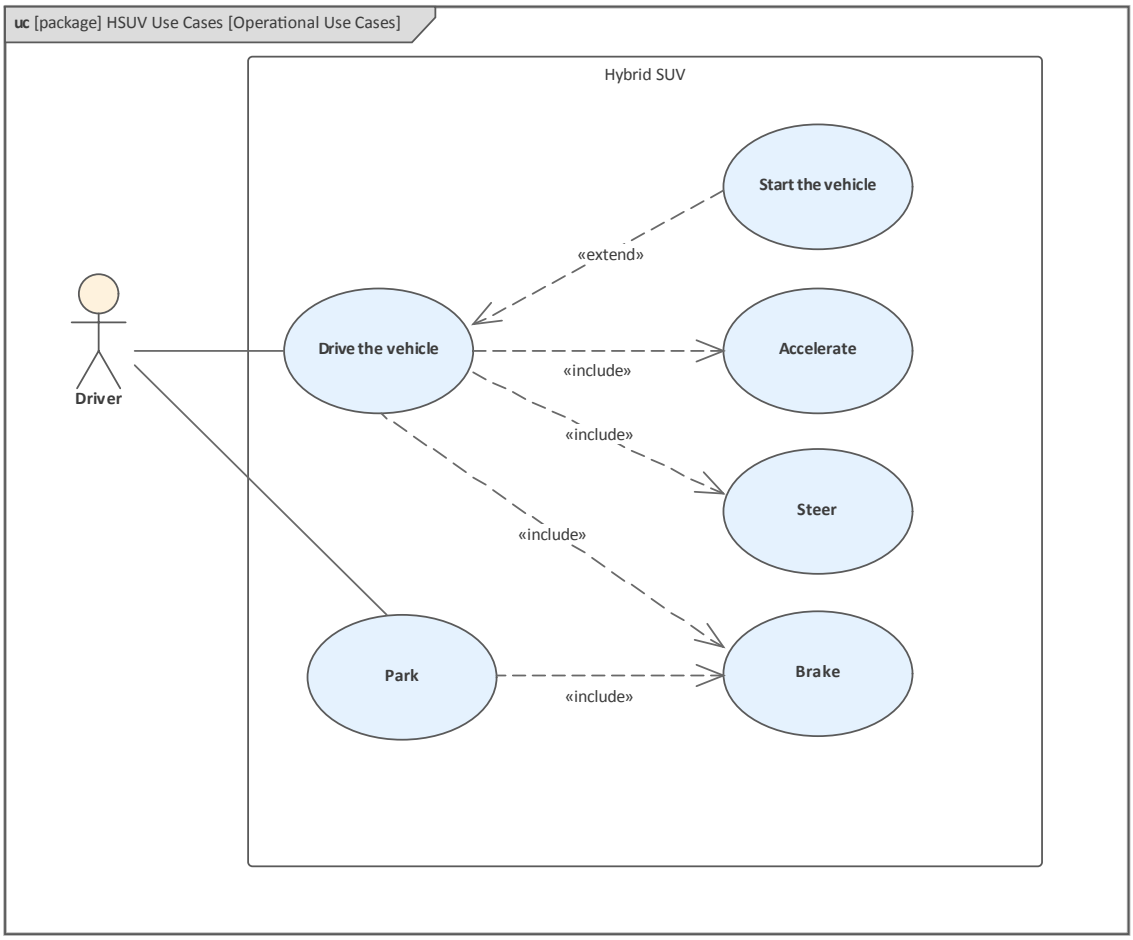
工具提供了一个完整的范围选项，**适用于用例用例或需求流程**，范围从通常称为基本流程的选项，其中任何参与者和使用案例的命名和使用案例都给出了描述，到部分穿着完成基本流程的过程。 A完整的流程通常会详细说明基本细节中的所有步骤，并定义和详细说明替代方案和细节方案的步骤。有关详细信息，请参阅Scenario Builder主题。

此外，可以添加任意数量的约束，例如前置条件和后置条件以及不变量。

Constraint	Type	Status
The customer has signed the workshop agreement.	Pre-condition	Approved
The customer's driver settings and preferences are unchanged	Invariant	Approved
The vehicle has new parts and lubricants in compliance with the service schedule.	Post-condition	Approved
The vehicle is road worthy.	Pre-condition	Proposed
The vehicle is safe, road worthy and performing optimally.	Post-condition	Approved

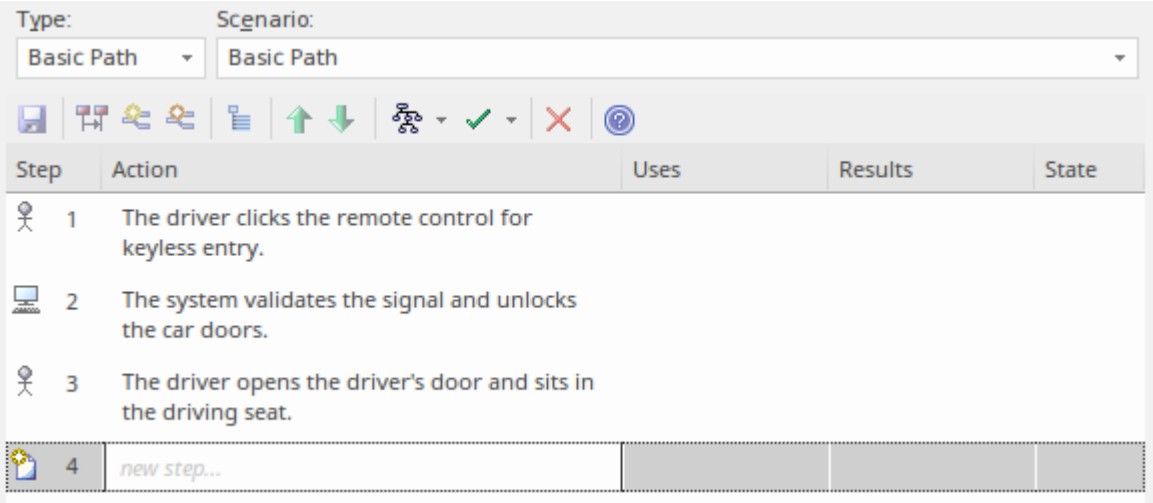
构建一个用例模型

虽然用例模型提供了高水平的可视化，并且系统工程师被警告不要应用功能分解，但 SysML 确实提供了许多有助于构建用例模型的机制，以确保可以重复使用离散的功能片段。这些机制由<<include>>Dependency、<<概括>>Dependency 和相关关系组成。



生成行为图表

Enterprise Architect有一个有用的生产力工具，它允许根据场景构建器中定义的用例规范自动生成行为图。这提供了一种可视化这些文本描述的方式。它还允许在用例描述中的步骤和其他建模元素之间绘制关系。



用例报告

用例文档的创建传统上是一个手动过程，许多项目中的文档长达数百页，它们的生产消耗了宝贵的项目资源。这些手工制作的文档变得难以维护，并且与项目的其他部分（例如需求、业务规则和解决方案组件）保持隔离。 Enterprise Architect有一个名为 Scenario Builder 的多功能工具，它允许建模者在模型中指定使用案例和场景，并且这些可以使用内置模板自动生成为高质量的文档。有两个内置模板可用于生成用例报告：一个在摘要级别记录用例，另一个在详细级别记录用例。

来自用例报告的示例内容

详细的用例报告将列出所有用例的详细信息和详细步骤，包括基本路径、备用和其它情况。包括内部需求、前后条件和其他约束在内的信息也将包含在报告中。如果已经自动创建了行为图（例如活动图），则该图也将显示在报告中。

Alternate. List Stock Levels by Publisher

The List Stock Levels by Publisher allows a user to obtain stock level information for a selected publisher. The Stock Control Manager and Storeroom Worker need this information to plan logistics and to ensure that stock remains at adequate levels to service incoming requests. There is also the need to predict the date that the stock items will fall below an acceptable level

SCENARIOS

based on purchase cycles and promotional periods.

1. User selects "List Stock Levels by Publisher"
   Uses:

2. System returns a list of publishers to select from
   Uses:

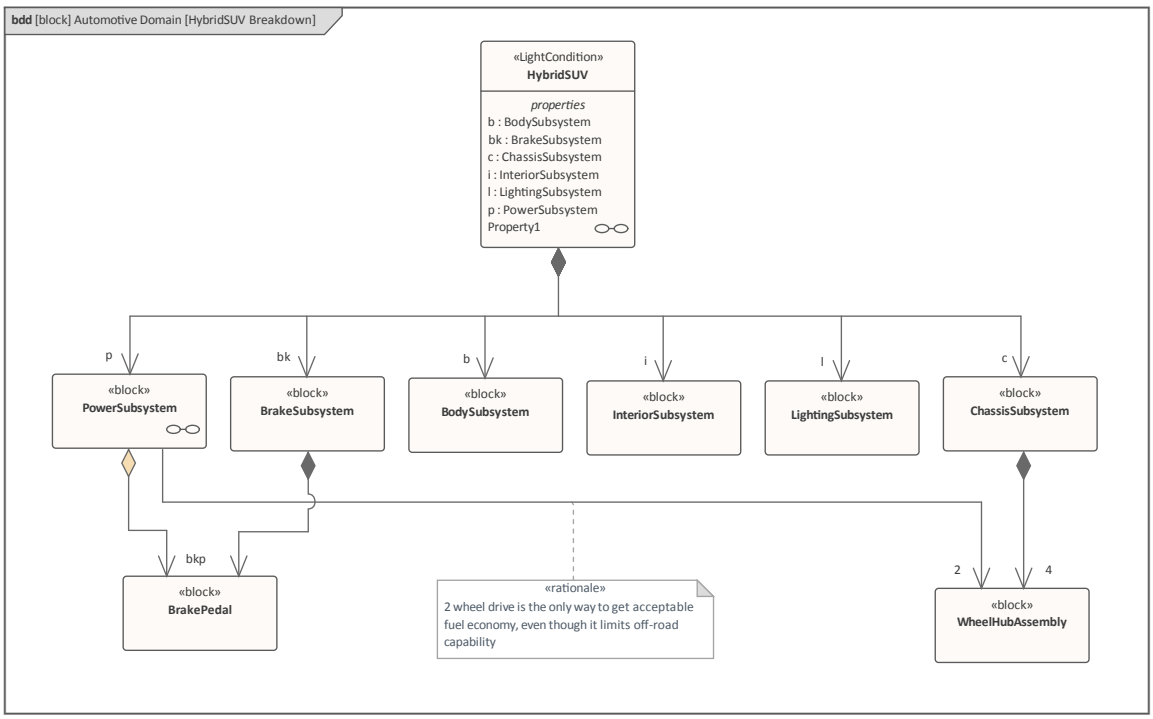
3. User Selects a publisher
   Uses:

4. System returns a listing of titles and quantity in stock for the publisher
   Uses:

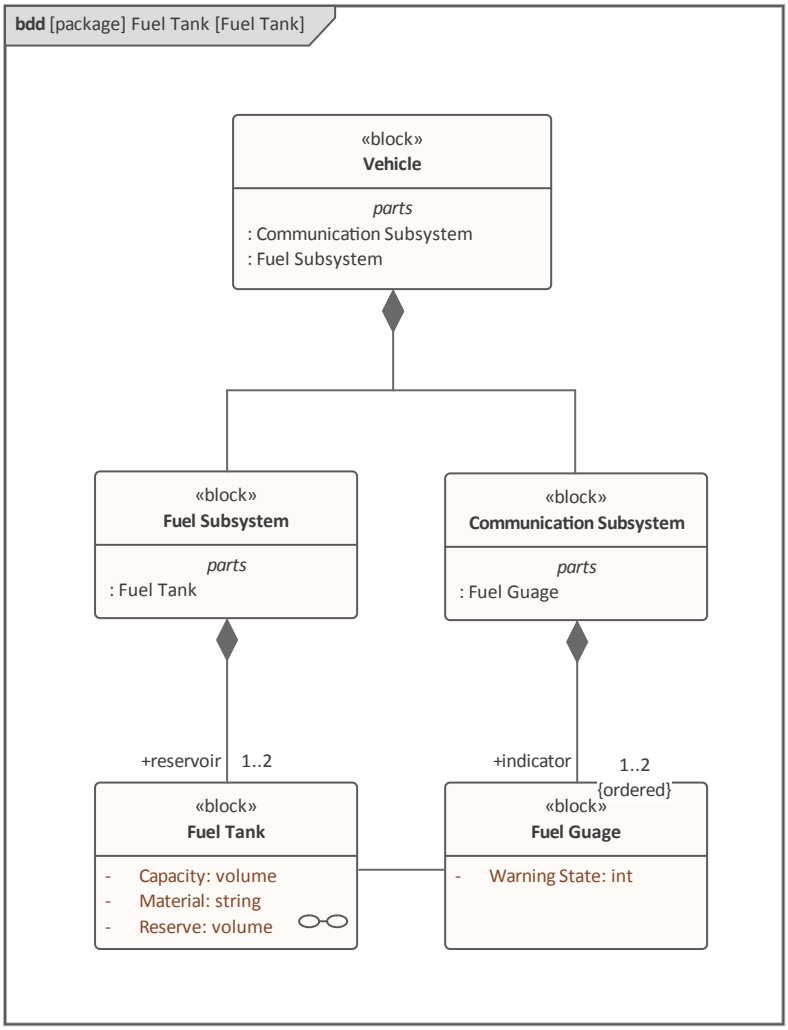
使用模块构建和约束模型

SysML 中的语言结构和表达方式与我们的自然语言一样，可以分为结构类型和行为类型。在英语、德语或日语等语言中，名词描述结构，动词描述行为。句子通常包含名词和动词的组合，可以揭示说话者世界的某些方面。 SysML 具有类似的划分，具有描述结构的元素和描述行为的其他元素。在 SysML 中，结构事物（名词）使用块来描述。当工程师创建图表时，他们通常会混合使用行为或结构元素，并且他们将描述系统的特定方面 - 揭示正在建模的系统的某些方面。

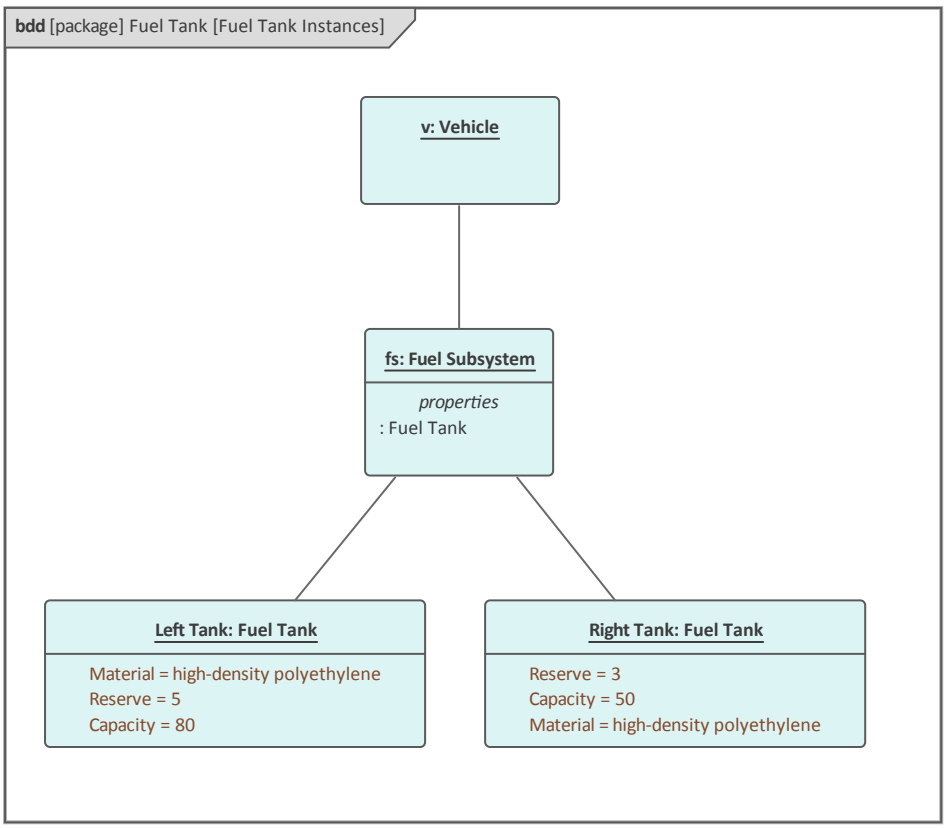
块是系统结构的基本单位；它可用于描述整个系统、子系统、组件、流经系统的项目、约束或驻留在系统外的实体。与我们的自然语言类似，块可以代表抽象的、逻辑的或物理的东西。这是一个重要的概念，SysML 的编写者和读者必须清楚表示的意图。例如，在逻辑架构中，通常有代表概念想法或设计的块，这些块在详细设计和构造时可能由物理和有形组件实现。系统架构师可能会定义A称为碰撞检测子系统的块，它是一个逻辑系统组件的表达，在详细设计相，它可以部分地由一组雷达和激光发射器、探测器和摄像机实现。



我们A许多自然语言都有一个称为分类器的语法术语，它将词典中的事物（名词）分组为具有共同特征和行为的事物类别。同样的原则也适用于块，它本质上是一种分类器，它将共享相同结构和行为特征的实例集合分组。块的实例可以以通用方式建模，也可以给定精确值，例如旅途中特定点或事故发生时油箱中的汽油量。



在油箱图中，汽车被建模为分类器（块）级别，其中模型描述通用车辆并表示车辆可能有一个或最多两个油箱的事实。但是，此油箱实例图描述了具有两个具有不同容量和储备容量的油箱的特定车辆。

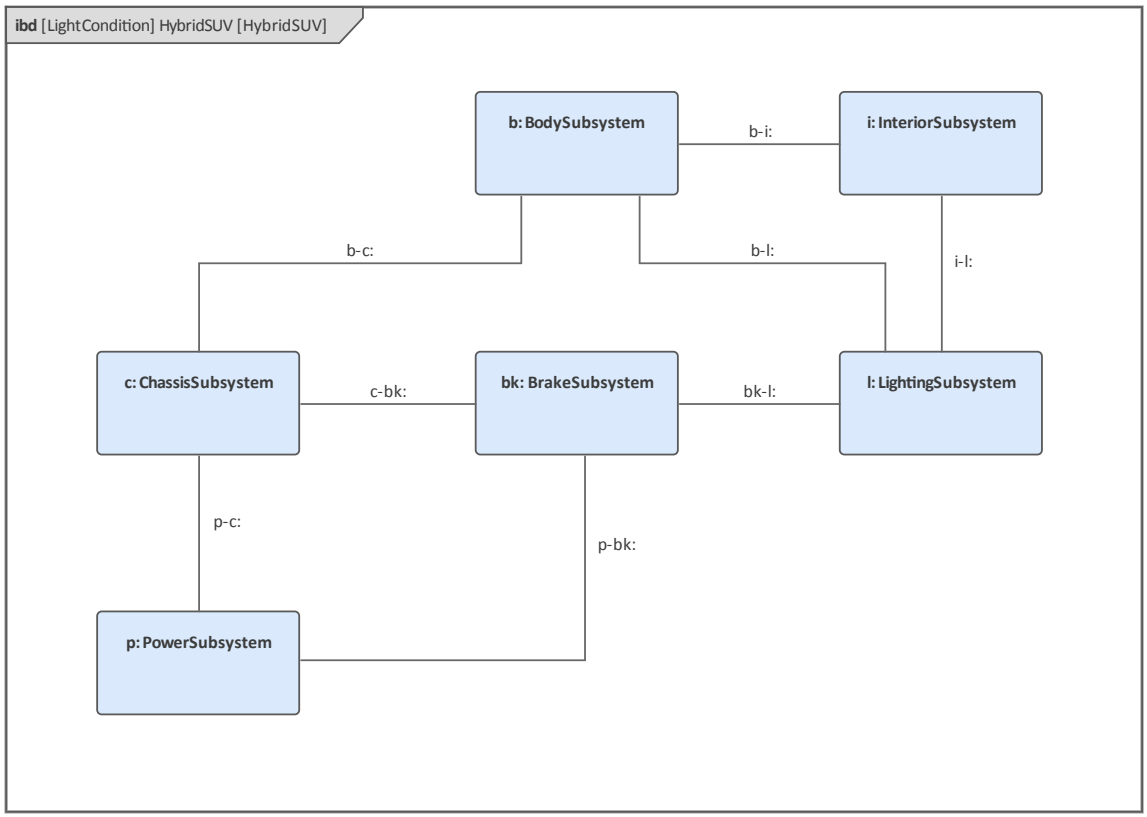


块定义A用于描述系统、子系统、组件或其他感兴趣的元素的特征集合（**或集合**）。这些特征可以包括结构和行为特征，例如属性、操作和接收，以表示系统的状态和系统能够展示的行为。

Enterprise Architect**有一套工具**，可帮助系统工程师使用 Blocks 并可视化系统定义中这些最重要元素的结构和行为。这些功能包括：

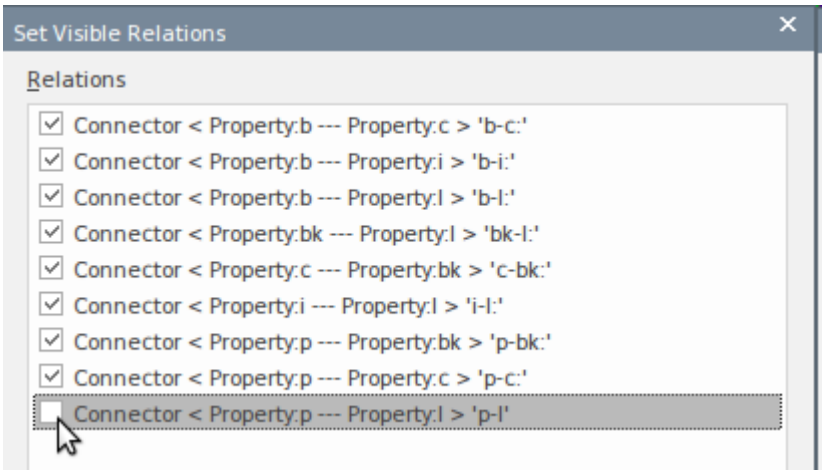
- 块定义图表，描述块、它们的特征点和结构关系
- 内部块图表，它根据属性和属性之间的连接器捕获块的内部结构

这个内部块图表显示了许多子系统如何协作来创建车辆的结构。例如，照明子系统与内部子系统有连接，而内部子系统又与体子系统有连接。



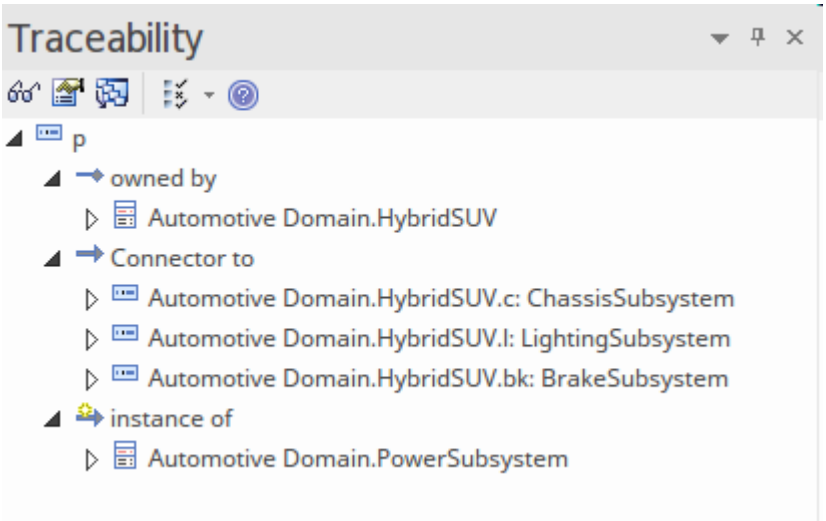
图表中已隐藏了一些关系；**例如**，电源子系统通常会连接到照明子系统。这一点很重要，因为 SysML 和 Enterprise Architect的新手通常认为每个定义的关系都应该显示在图表中。虽然这句话似乎是正确的，**但重要的**是要记住，建模师，就像创作漫画的漫画家一样，经常会在插图中聚焦细节，以将观众的注意力集中在其他主观上更重要的元素和连接器上。

此屏幕截图显示了工程师如何设置图表的可见关系。



如果在此对话框中未选中连接器，它将不会显示在当前图表中。但是，它可能在显示连接元素的其他图表中可见。这可以从 布局>图表>外观>可见性>设置可见性关系”功能区选项进行设置。

无论图表中显示哪些连接器，建模者始终可以通过选择图表中的元素并查看可追溯性窗口来查看元素的所有连接器。在此屏幕截图中，电源子系统已被选中，即使电源子系统和照明子系统之间的连接器在图中设置为 不可见”，也可以在可关系可追溯性窗口中看到。



一个块的特征要么是结构的，要么是行为的。

结构特征分为三种：

- **部件**- 描述块的组成；**例如**，车辆的底盘由两个车轴和四个车轮组件组成
- **参考**- 描述该块与其他块（**包括其自身**）的关系；**例如**，都市列车与车站和架空布线系统有关系
- **价值**——描述一个块的可量化方面；**例如**，尺寸、温度和亮度等

行为特征包括：

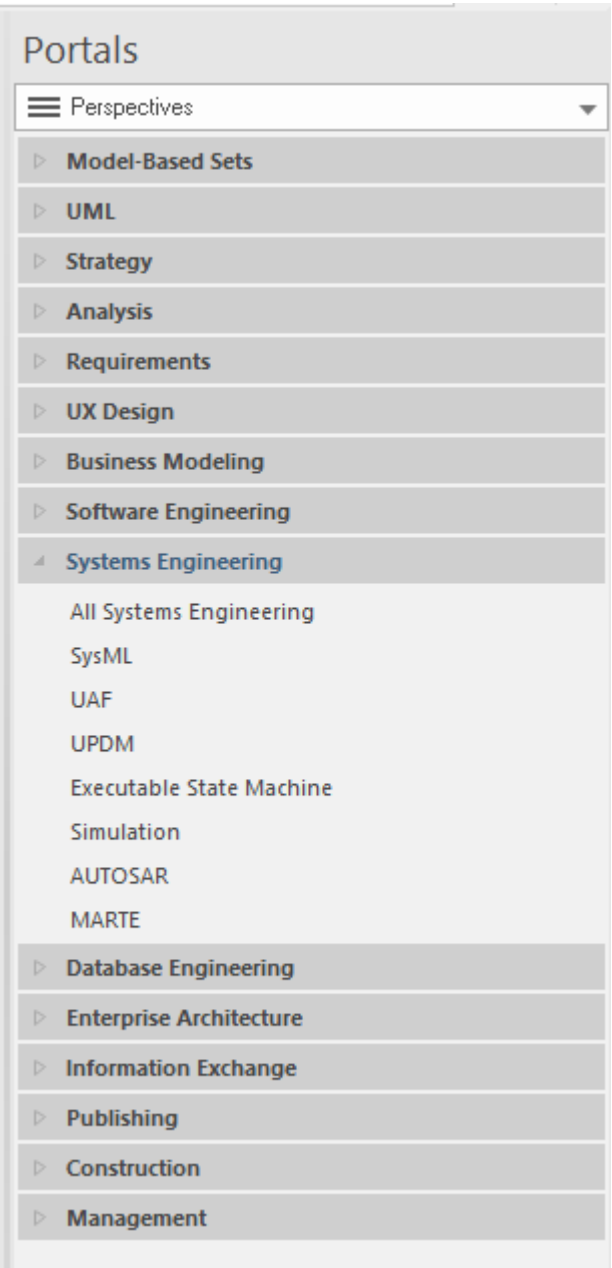
- **操作** - 通常表示同步请求
- **信号接收**- 表示来自信号的异步请求

用块开始

Enterprise Architect的设计师为系统工程师和其他学科创造了极大的灵活性，他们认识到团队成员经常扮演各种角色，需要在一个项目中，甚至在一天内有效地多次更换帽子。蓝图和工作空间提供系统工程师无需启动其他工具即可有效更改角色的便捷有效方式。

设置蓝图和工作空间

使用该工具一段时间的系统工程师会从系统工程蓝图集中选择一张蓝图；通常这可能是 SysML蓝图，让他们可以访问所有模式和工具箱页面来创建任何一个 SysML 图，包括块定义和内部块图。



正如介绍性主题中所解释的，可以设置工作区以促进特定类型的建模；在块定义图的情况下，核心工作空间集中包含的任何一个工作空间都将对建模有用，包括：

- 基本图表

- 核心建模
- 默认

在项目级别工作的工程师也可能会发现一些有用的施工工作区，包括：

- 路线图
- 看板
- 文件出版
- 评论和讨论

创建块定义或内部块图表

使用 Blocks 时通常会创建两个图表：

- 块定义图 (BDD) - 用于显示块之间的结构关系，包括零件和类型的层次结构，以及与其他块的参考连接
- 内部块图 (部件) - 用于显示组件属性如何直接连接，或通过交互点（如端口和接口）连接

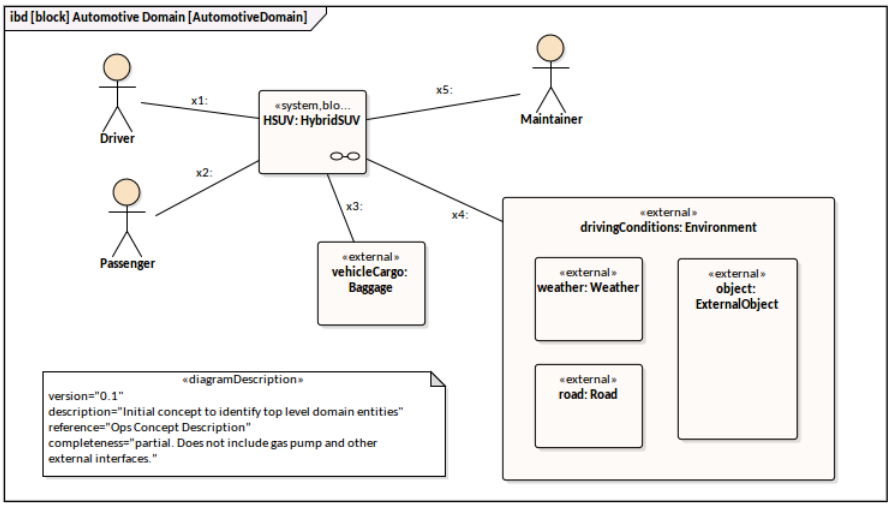
我们将在指南的这一章中讨论块定义图；在下一章中，我们将讨论内部块图，它将演示如何在给定时间使用上下文。

内部块图表-设置上下文

在计划早期创建的最重要的图表之一是上下文图，它描述了在其环境或领域的时间范围内上下文的产品或服务。这有助于模型查看者清楚地了解和了解产品如何在一个或多个需要运行的环境中。元素可以及早指示项目范围内和范围外的内容。在图中已标记为 外部”（使用刻板印象），表明它们构成产品或服务环境的一部分或上下文。

图像已被用于软化图表，使其更吸引广泛的利益相关者，包括商业和非技术观众。

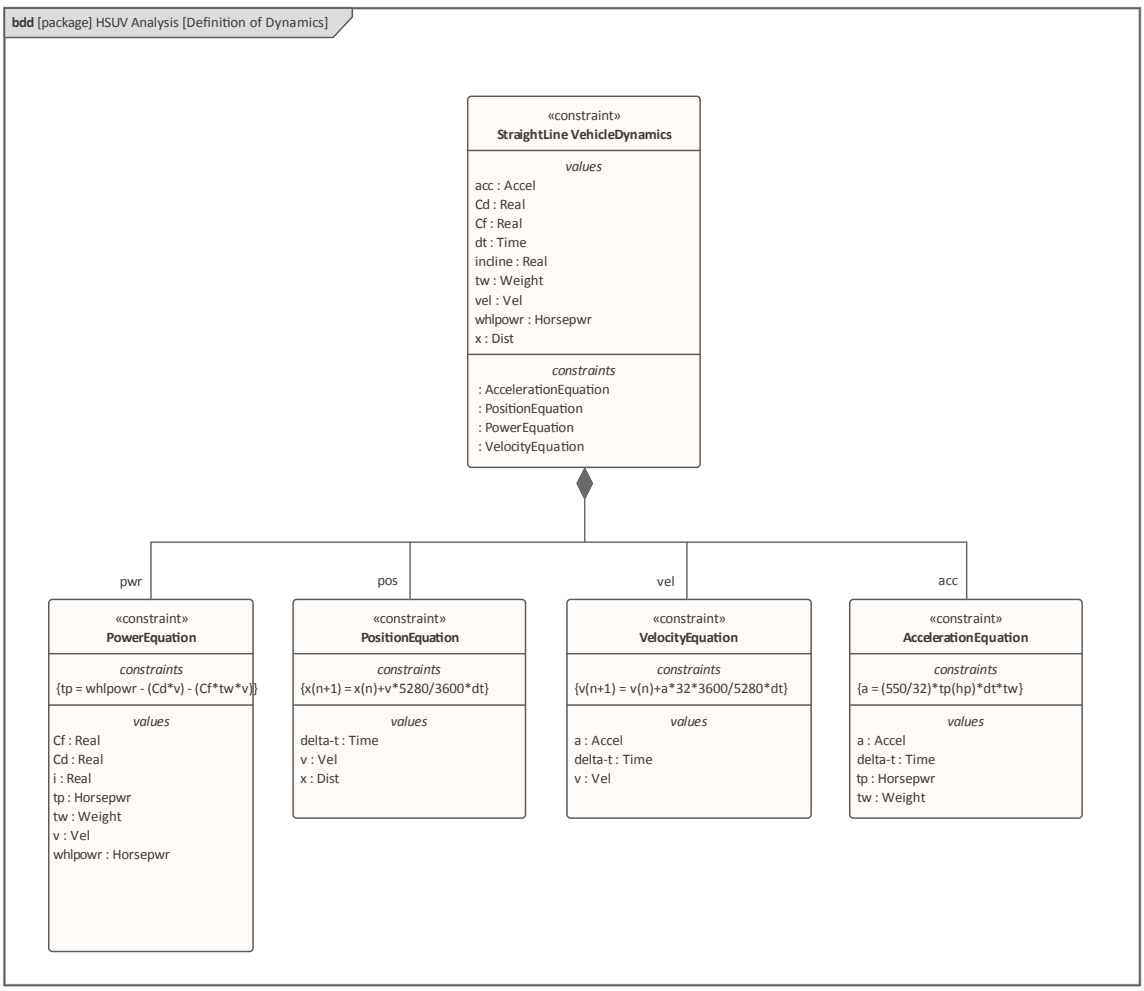
这张内部块图显示了混合动力上下文在典型城市环境的时间。设想至少可以创建一个其他图表，显示一个越野环境，其中道路将被土路和河流取代，建筑物被山脉和树木取代。



造型约束作为积木

工程分析要求工程师执行各种功能，包括构建性能和可靠性模型、权衡分析、替代分析和交易研究。这些分析通常需要使用用于约束分析元素的数学表达式和方程。 SysML 以 ConstraintBlock 的形式提供了一种语言机制，用于以图形方式对方程进行模型。这是一个有用的机制，它允许表达式连同它的参数和它们的类型一起被表达。然后可以在许多不同的环境中重复使用建模方程，允许工程师定义牛顿第二运动定律 {F=m*a} 或卡诺定义的公式，从而得出热力学基本定理 {p=W/ t =(mg)h/t}。以图形方式对这些方程建模的一个有用结果是它们可以与其他模型元素相关，例如利益相关者的需求、任务目标和较低级别的元素，例如块和实施工件。

Enterprise Architect允许对这些 ConstraintBlocks 进行建模，然后将其用作参数图上的约束属性。约束定义可以分组到库中，不仅用于当前计划，而且在多个项目和计划中重复使用。在后面的主题中，我们将看到如何将约束构建到方程网络中并在参数图上用于评估备选方案，并进行权衡和备选方案分析。该工具的精度和卓越的技术将确保以这种方式定义的方程可以被严格地创建、维护和使用。



介绍块定义图表

块提供了一个统一的概念来描述元素或系统的结构，包括：

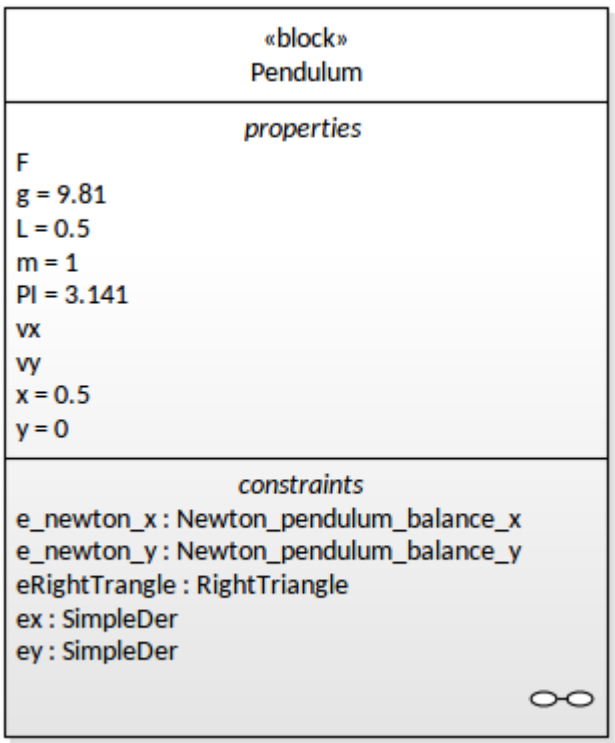
- 系统
- 硬件
- 软件
- 数据
- 过程
- 功能
- 人们

块可以有多个标准隔间，可以在图表上显示以描述块的特征，包括：

- 属性（部件、引用、值、端口）
- 操作
- 约束
- 从/到其他模型元素（例如活动）的分配
- 需求块
- 用户定义的隔间

可以抑制任何隔间。没有为缺少A隔间绘制分隔线。如果隔间被抑制，则无法推断其中是否存在元素。在此图中，已对 Pendulum 进行建模，并且已使多个隔间可见，以准备创建参数模拟。

-



额外的隔间可以作为工具扩展提供，以显示其他预定义或用户定义的属性模型（例如，显示业务规则、职责、变化、处理的事件、引发的事件等）。

每个块必须有一个在其命名空间内唯一的非空名称。名称的范围是它的包含包和其他可以看到包含包的包。

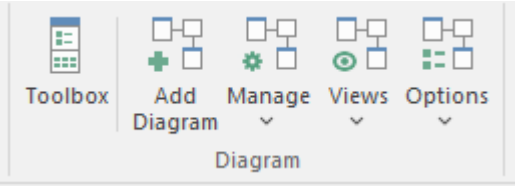
创建一个块定义图表

可以A以下任何选项从用户接口中的多个位置创建块定义图：

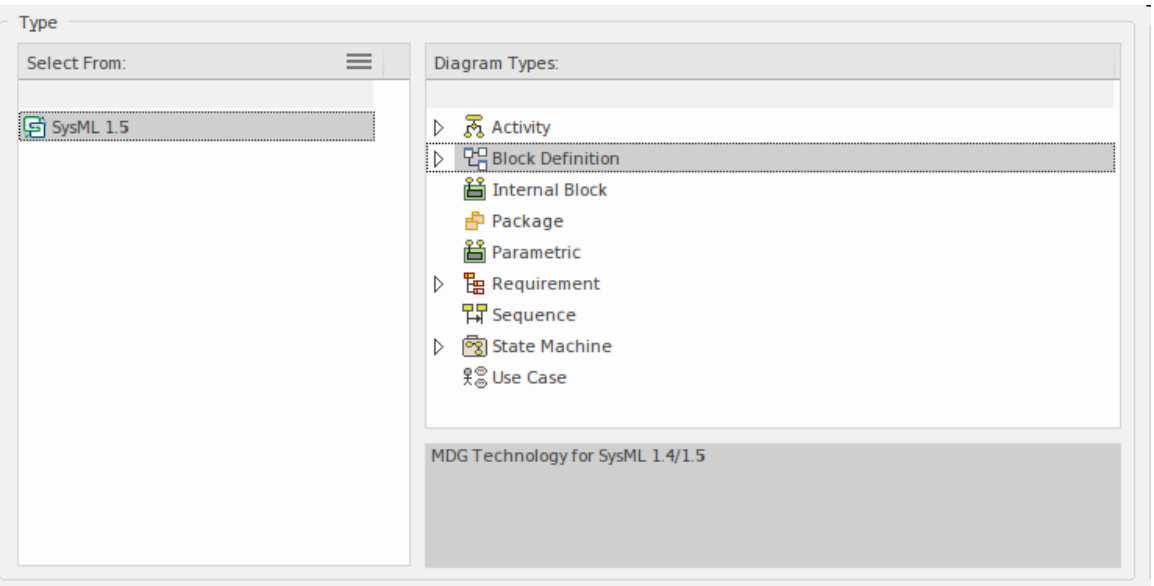
- 设计ribbon - 在图表面板上添加图表图标
- 浏览器窗口工具栏 -新的图表图标
- 浏览器窗口上下文菜单-图表图形

我们将使用设计功能区创建一个块定义图。首先，在浏览器窗口中选择您想要放置块定义图的位置。与所有图表一样，这可以在包或元素块下，但通常将定义图插入包中。在浏览器窗口中选择位置后，选择功能区选项：

设计>图表> 添加图表



选择此选项将打开 新建图表”对话框，允许您命名图表。该名称最初默认为包含图表的包或元素的名称。选择 SysML 透视图并选择 SysML 版本后，将显示图表列表；选择块定义图并单击确定按钮。将在浏览器窗口中选择的位置创建A新的块定义图。将打开图表视图，允许您开始添加描述块和其他重要结构元素（如端口、接口和值类型）的元素和连接器。 Enterprise Architect还将显示工具箱的 块定义”页面，其中包含图表规范定义的元素和关系，适用于构建块定义图。如果需要，除了始终可用的 公共”元素和 公共关系”页面之外，还可以打开任意数量的其他工具箱页面。



与块定义图一起使用的最重要的元素和连接器是：

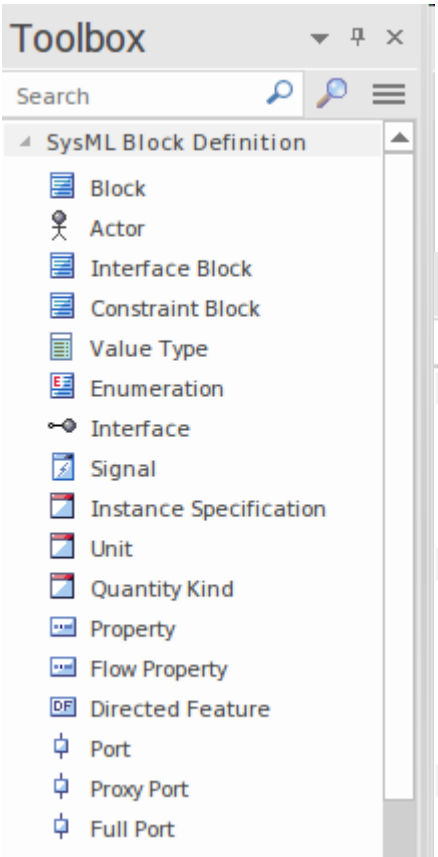
元素

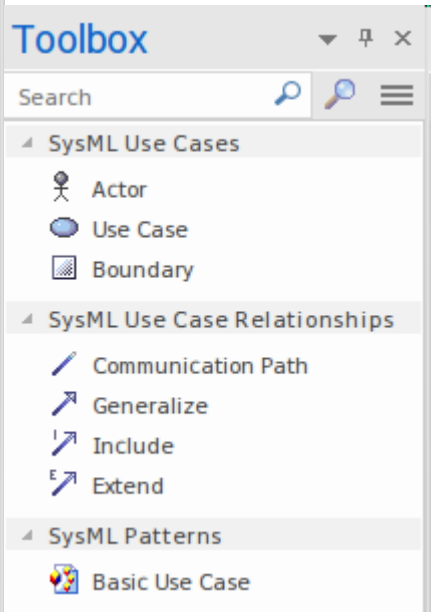
- 块
- 约束块
- 值类型
- 属性

- 单元
- 数量种类
- 代理端口
- 全端口

连接器

- 参考关联
- 部件关联
- 共享关联
- 概括
- 依赖
- 项目流

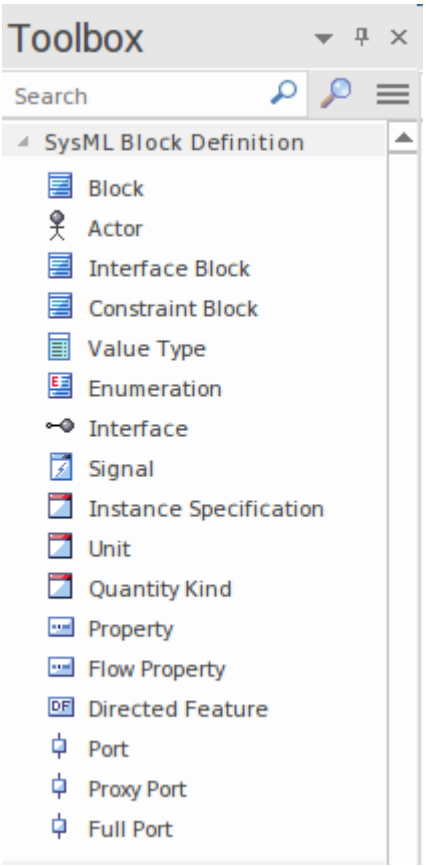




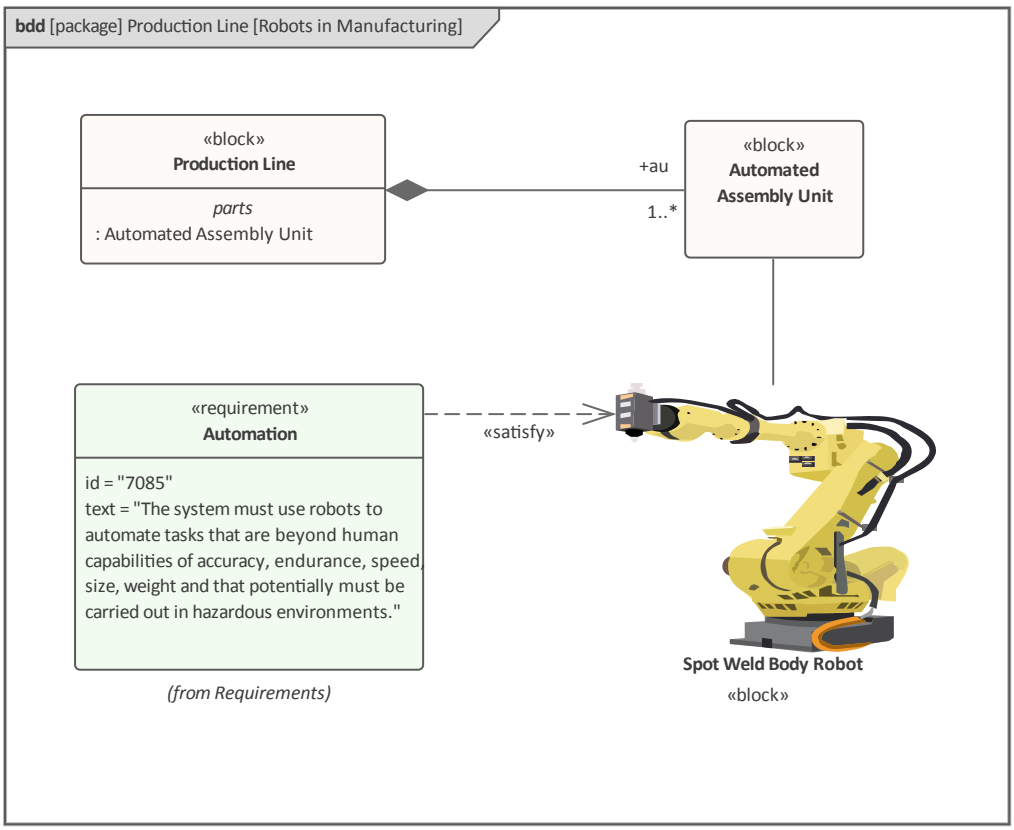
可以通过将元素从工具箱到图表视图视图中来将元素添加到图表中。有关更多信息，请参见Block Definition Diagrams (BDDs) 帮助话题。

创建元素块

块元素，与任何其他类型的元素一样，可以使用包上下文菜单上的 添加元素”选项创建，或者使用图形工具箱的 图表块定义”页面在块定义上放置块图（BDD）。



块通常出现在多个 BDD 上，其中每个图表都旨在解决特定利益利益相关者或利益相关者群体的问题。Enterprise Architect在单个块（**或任何元素**）级别或图表级别具有广泛的显示选项。例如，这些可用于决定要显示哪些隔间，甚至要为单个元素显示哪些特征，还有广泛的通用元素和图表设置来设置元素和图表的样式。例如，可以设置元素颜色，包括填充、边框和文本，或者通过应用更好地传达块功能的图形图像来更改元素的函数。在此示例中，建模者决定为点焊机器人使用替代图像，以更清楚地传达生产线上发生的自动化。

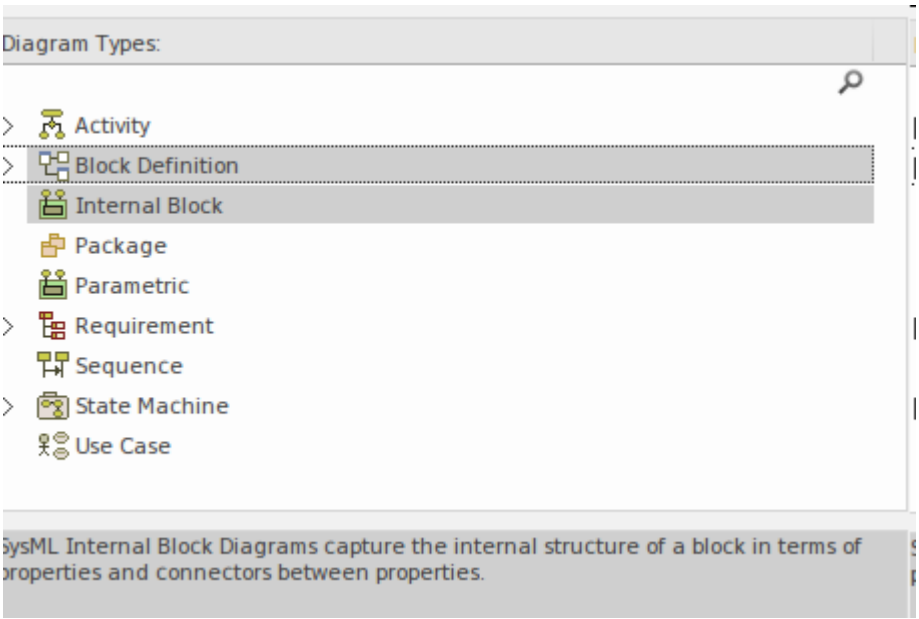

bdd [package] Production Line [Robots in Manufacturing]
«block»
Production Line
parts
: Automated Assembly Unit
+au
1..*
«block»
Automated
Assembly Unit
«requirement»
Automation
id = "7085"
text = "The system must use robots to automate tasks that are beyond human capabilities of accuracy, endurance, speed, size, weight and that potentially must be carried out in hazardous environments."
(from Requirements)
«satisfy»
Spot Weld Body Robot
«block»

基本结构建造块

块是系统描述的基本和离散的模块化单元。块定义了A组特征，这些特征用于定义系统的一个方面或系统本身。特征有两种基本类型：结构（一个块由什么组成）和行为（它做什么）特征。块与其他块（包括它自己）以及与其他类型的元素A关系，有助于描述系统、子系统或组件的结构。

系统建模者使用块定义图形（图表）来定义块的结构，并使用内部块图形（图表）来描述它们的用法。

可以从浏览器窗口工具栏访问的 新图表”对话框中创建这些图表。



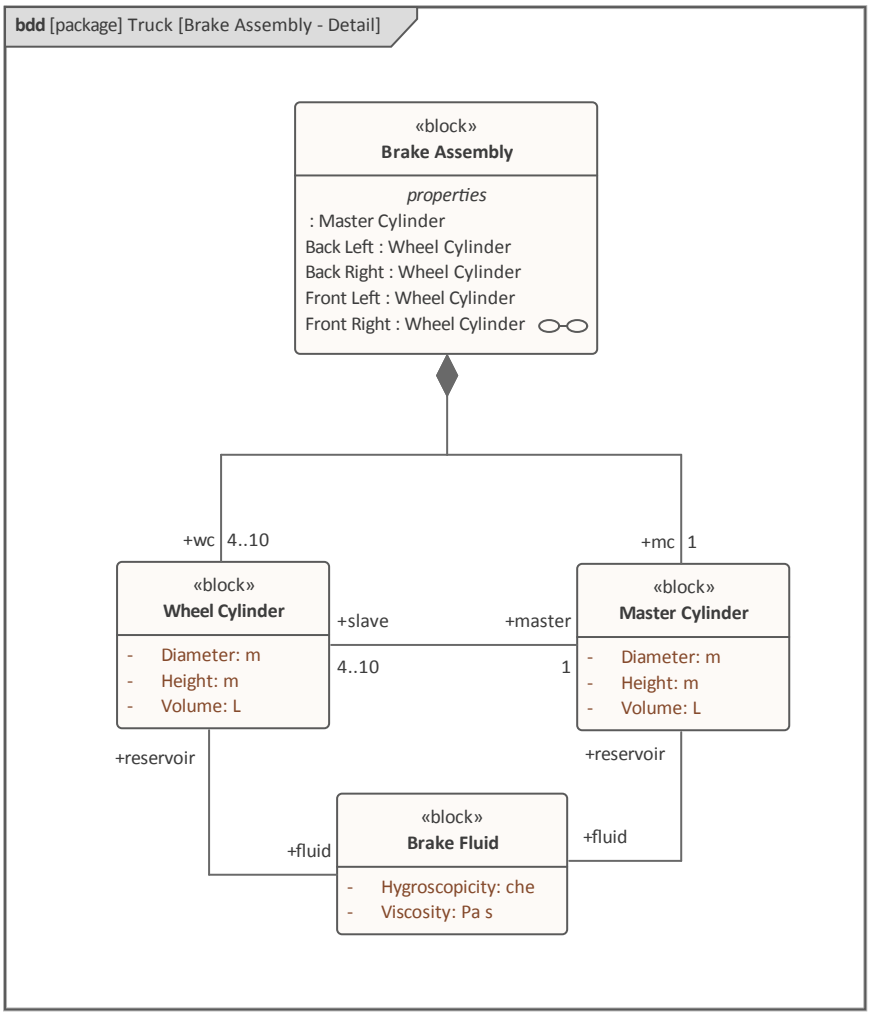
建模结构特征

块通常使用一系列结构特征来定义。这些是块的属性并定义块的性质。例如，火车引擎（机车车辆）将具有属性类、标识号、车轮组件数量、原动力、电机和一系列其他属性等属性。要记住的重要一点是，块是描述一组引擎的分类器。您暑假乘坐的火车前部的发动机是发动机的一个实例，它将具有特定的类，例如 OSE 类 660，标识符为 SM-09873，以及 8 个车轮组件。

Enterprise Architect支持三种基本类型的结构特征，每种对于建模块结构的不同方面块重要。我们将在这些部分中查看它们中的每一个。

- 部分 -一个块由部分组成
- 参考资料 -参考其他区块的特征
- 值 -描述数量

该图显示了所有三种类型的结构特征。

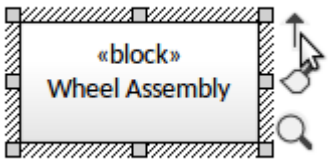


制动系统由多个部分组成，图中显示了其中两个。连接部件已用于表明主缸关联和轮缸s 是制动系统的基本组成部分。参考关联已用于A两种类型的气缸之间以及气缸与制动液之间的关系。已作为属性输入的值与其随附的值类型一起显示；例如，Volume 的类型为L ，即 SI Unit 为 Litre 的 Volume 的 Dimension 的符号。

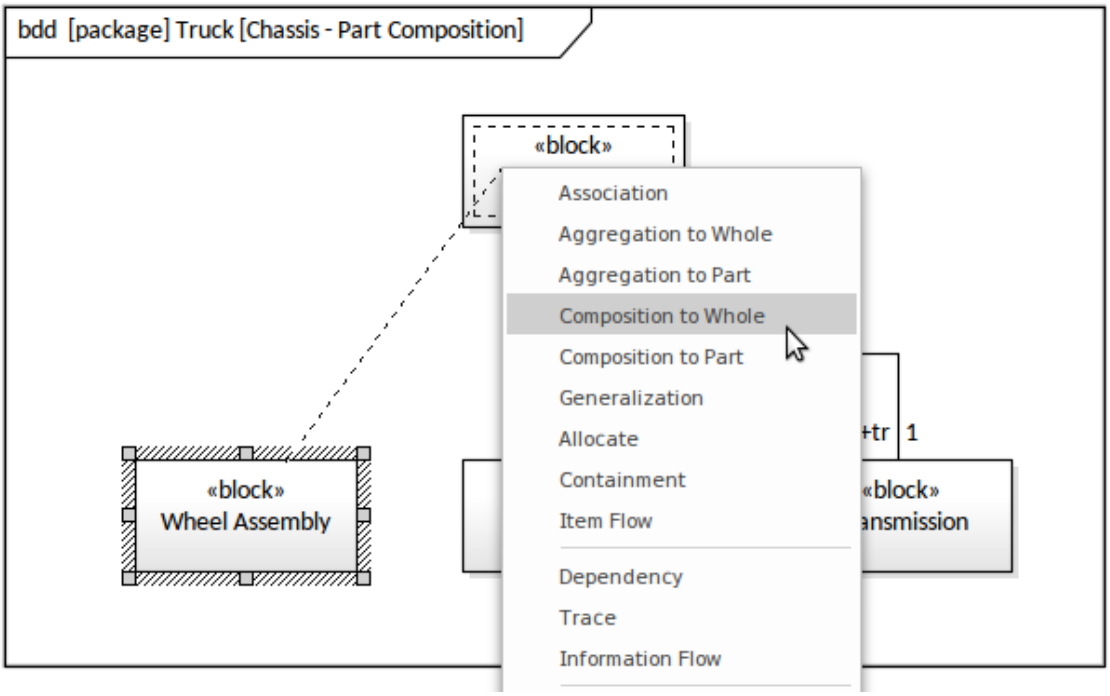
由零件组成的块

部件是A块的结构特征，并形成了一个块与其属性之间最强的关系之一。重要的是要理解一个块的实例可能有一个部件的多个实例。例如，一辆卡车可能有多个车轮组件，并且 - 根据卡车的尺寸和类型 - 这可能低至 2 个或高达 10 个。这些可能的配置可以在块及其零件的定义中指定，正式称为多重性 - 较低的数字称为下限，较高的

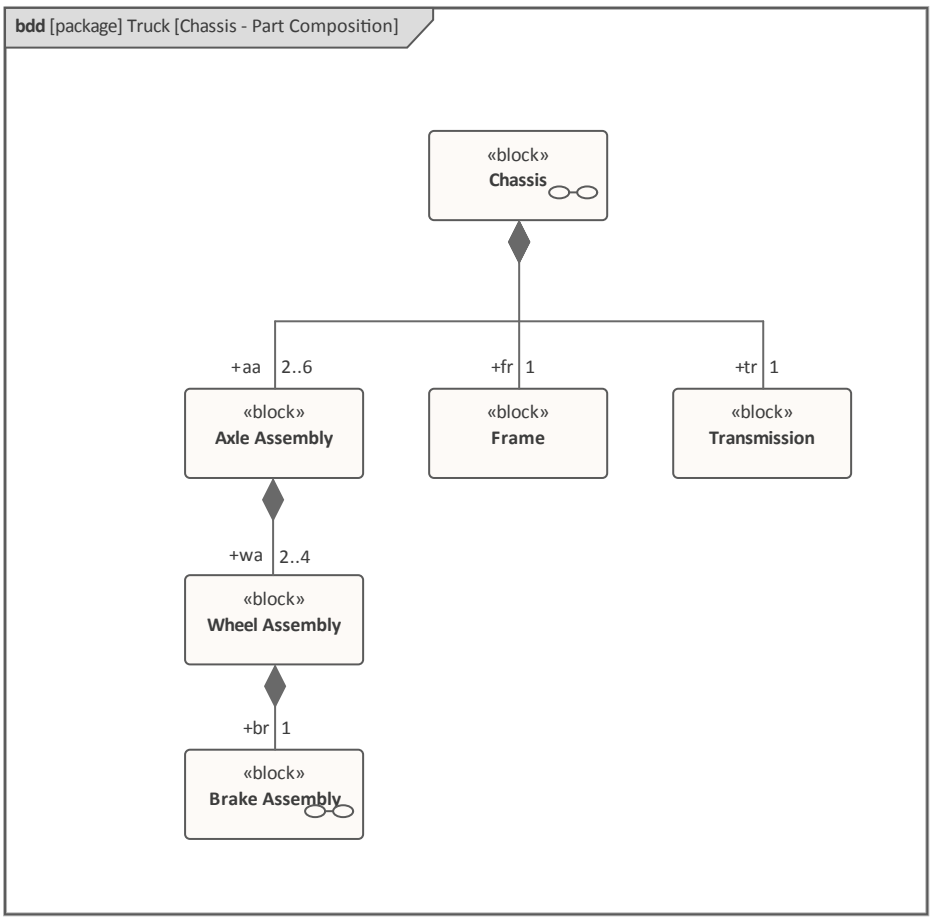
数字称为上限。部件A由另一个块键入，因此在示例中，部件的类型将是另一个块，称为 轮组装”，它本身通常包括一个轴和两个轮子组件。因此，每个部件都将在块中定义一个名称、一个类型和一个多重性。该工具允许以多种方式创建部件组合关系，但也许最直接的方法是将机箱块（整体）和组装块（部件）拖到图表上，然后使用 Quick链接器从部件（组件）拖动到整体（底盘组装。



从源object拖动到目标将显示一个可能的连接器菜单，工程师将选择组合到整个连接器。结果将是该行的*Chassis*端的菱形标记的关系，表示它是整体，而元素的组装是部件。

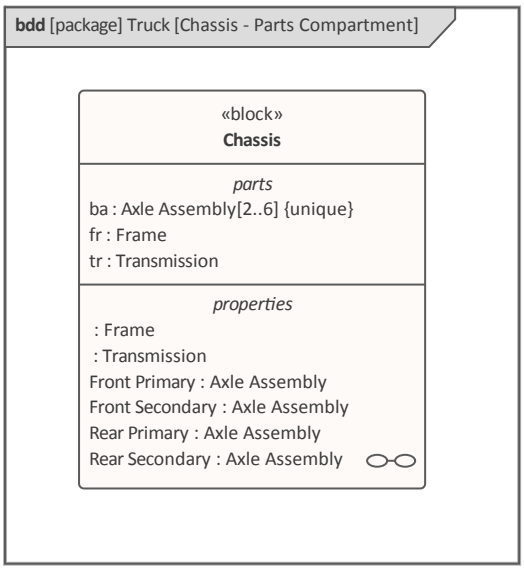


连接器属性将允许您设置源角色和多重性，如前所述，它们指定机箱的每个实例的名称和可能的零件数量。

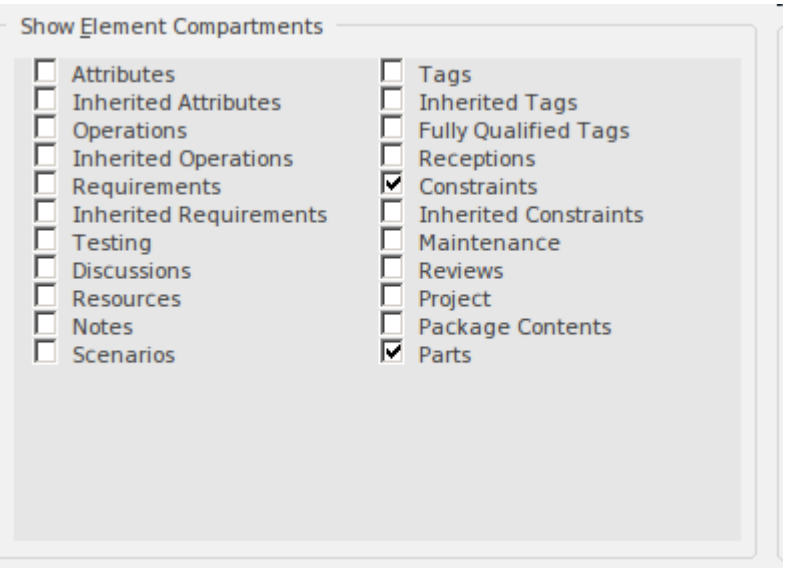


在图中，建模者使用部件连接明确定义了部件，部件连接可从部件块定义关联中获得。

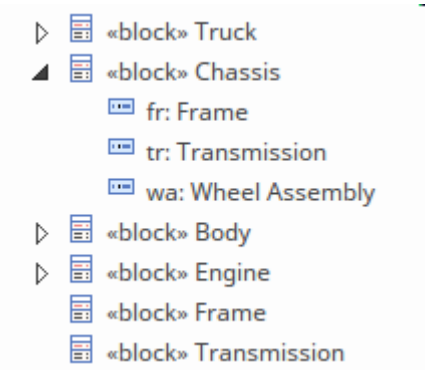
在此图中，**建模者使用** Owning块的部件隔间来显示 Chassis块拥有的部件。



默认情况下会显示部件隔间，但可以使用图表属性在图表级别控制其可见性，或者使用元素元素 "上下文可见性"选项在单个元素级别控制其可见性。在图表级别设置可见性将导致图表中的所有元素都符合指定的可见性 - **按**指定显示或不显示 - 而在元素级别设置它只会影响所选元素。



存储库元素将被更新，无论它们是否在图表或浏览器或任何其他窗口中进行编辑。在示例中，工程师通过从工具箱中拖动一个部件关联来创建图表中的部件；作为对此Enterprise Architect的响应，创建了三个新部件，它们位于浏览器中的机箱节点下，如此屏幕截图所示。



部件关联关系是最强的关联类型 - 从最弱到最强的强度连续体是：

1. 参考关联
2. 共享关联
3. 部件关联

我们将在本指南的后面部分探讨其他关系。

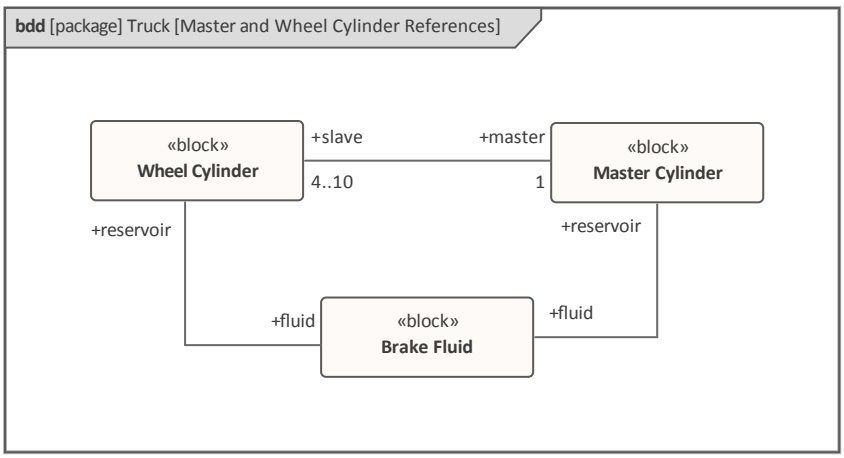
对其它块的引用

如前所述，部件关联是 SysML 中最强的关系类型，并暗示了整体的责任感：

- 它对其组成部分的生命周期负责
- A零件只能以一个区块参与零件合成

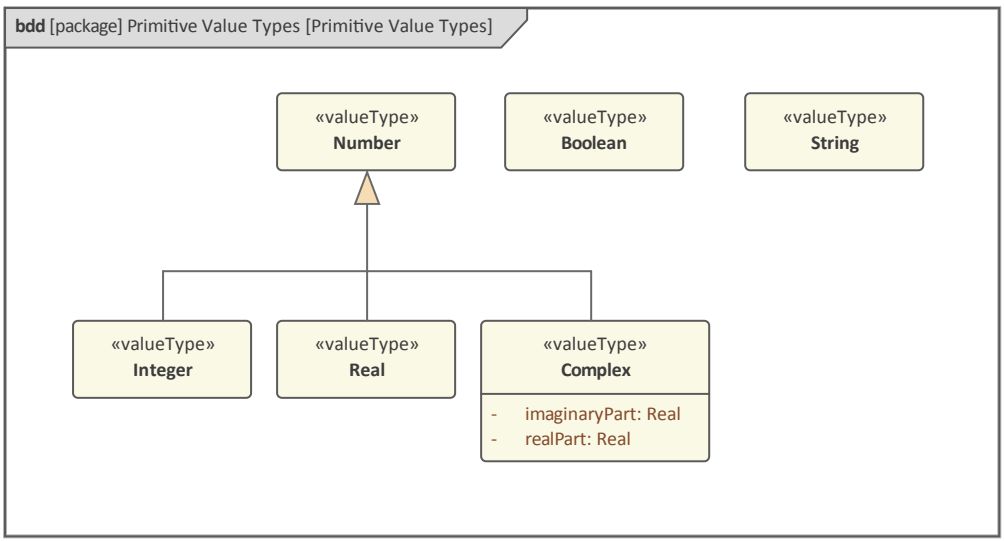
第二个条件意味着一个部件组合的整个末端的多重性始终是1 .. 1 ，可以缩写为1 。

然而，还有另一种关系，即参考关联（或简称为参考），可用于指定块之间的关系，与组合或一个块是另一个块的一部分的概念无关。这提供了一种非常有用的机制，用于在属于不同部分层次结构的块之间或在任何两个相互关联的块之间创建关系。例如，主缸和轮缸都与制动液有关系，制动液用于填充它们的油箱。轮缸反过来可能与定期检查气缸是否有泄漏会影响制动系统效率的机械师有关系。`

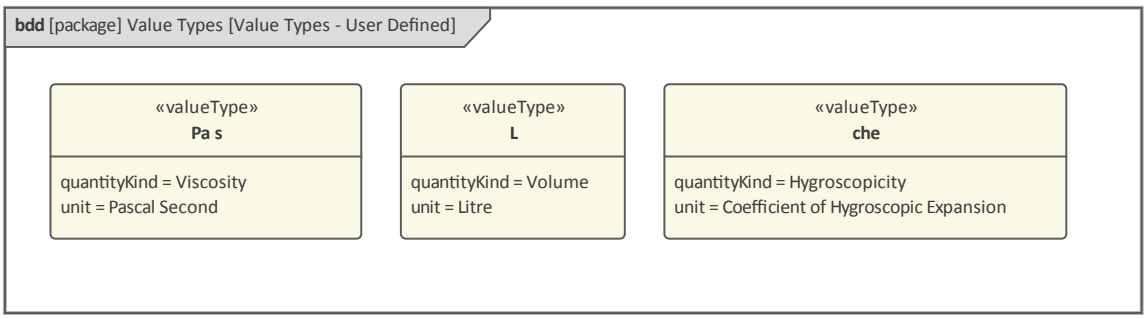


用于描述数量的值

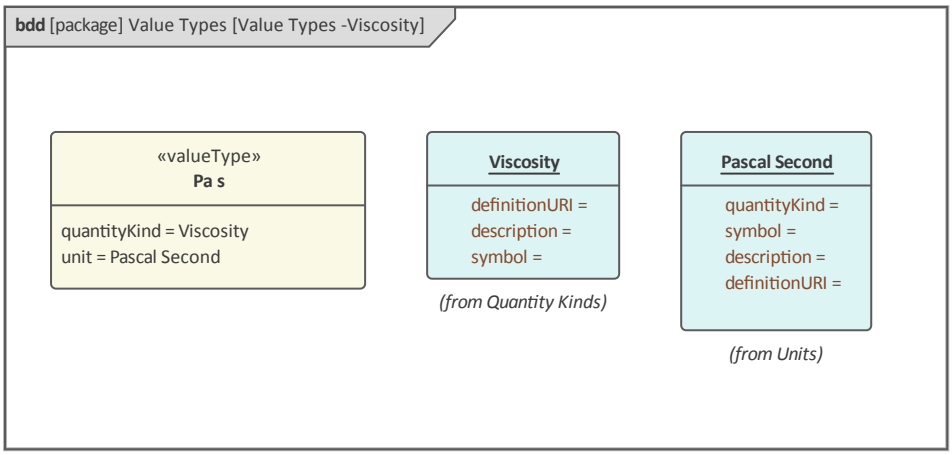
块可以具有具有某种可量化值的属性；例如，一个发动机有一个动力输出，一个水库有一个容积，一辆汽车有一个颜色，一辆铁路车厢有许多转向架。定义的类型可以是Number、整数、Complex、Boolean或字符串，如图所示。



工程师、团队或实践社区还可以定义任意数量的简单或结构化的值类型。这些可以基于任意数量的单位系统，例如国际单位系统(SI)。设计制动系统的汽车工程师可能会发现自己使用了许多标准 SI 值类型和许多派生类型，以及未定义为该标准一部分的其他值。此图说明了如何使用 SysML块定义工具箱中可用的 Value元素类型来定义这些值。



值类型有两个定义的属性——数量种类和单位。这些也可以在Enterprise Architect中建模，并严格应用价值类型。工程师会知道该类型基于数量（尺寸）和定义的单位。此图显示了这些元素的（粘度）值类型。



建模行为特征

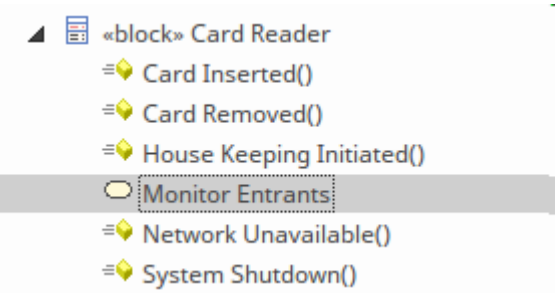
当系统运行时，已被定义为架构和详细设计一部分的块的实例被实例化。此时，如果一个块具有分类器行为定义，则此行为通常会开始并继续运行，直到该块被销毁。因此，在我们的停车场系统示例中，当系统启动时，读卡器将开始运行，其主要行为将生效。除此之外，一个块（即使本质上是结构性的）具有将被要求进行工作的行为特征。总之，在块的上下文定义了两个基本的行为定义，即：

- 分类器行为-当一个块被实例化时启动的本机行为
- 特征行为-这些是操作和信号接收（及其相关信号）

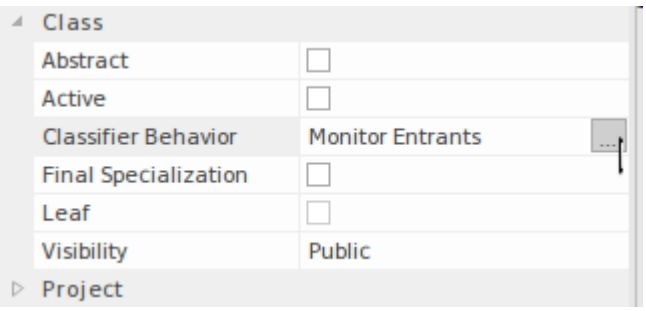
我们将在本指南的下一部分中介绍这些不同的行为，但重要的是要了解它们将协同工作，由系统交互协调，这将确保操作按序列调用并且信号被接收和执行通过信号接收。

块分类器A行为

块具有工作A潜力，但它本身是一个有点潜在的实体，需要通过某种类型的对其操作的调用或通过接收信号、状态变化或其他行为触发来命令其采取行动。 A块有它的原生或分类器行为的概念，正如它的正式名称。此图显示了浏览器窗口中的一个块，该块具有一个嵌套活动，该活动将被定义为该块的分类器行为。

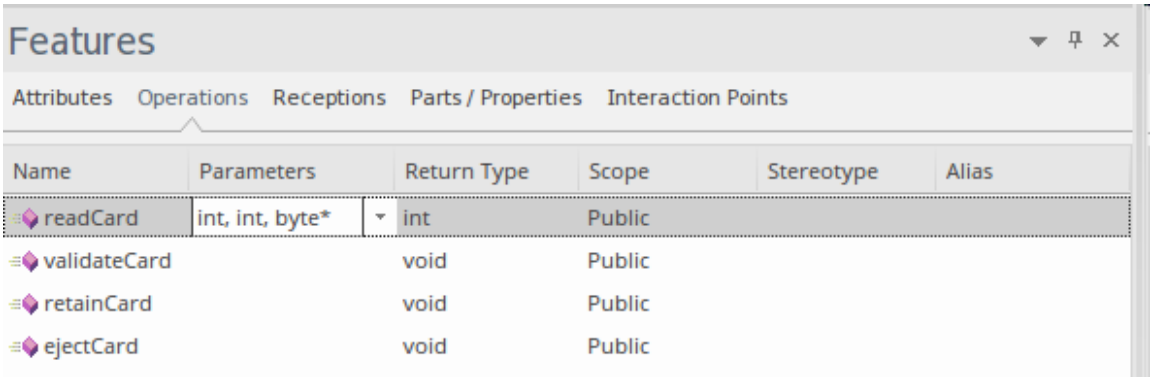


要为分类器行为选择此行为，请打开属性窗口并更改分类器行为属性，方法是选择 [...] 图标并找到相应的行为(活动)，如图所示。



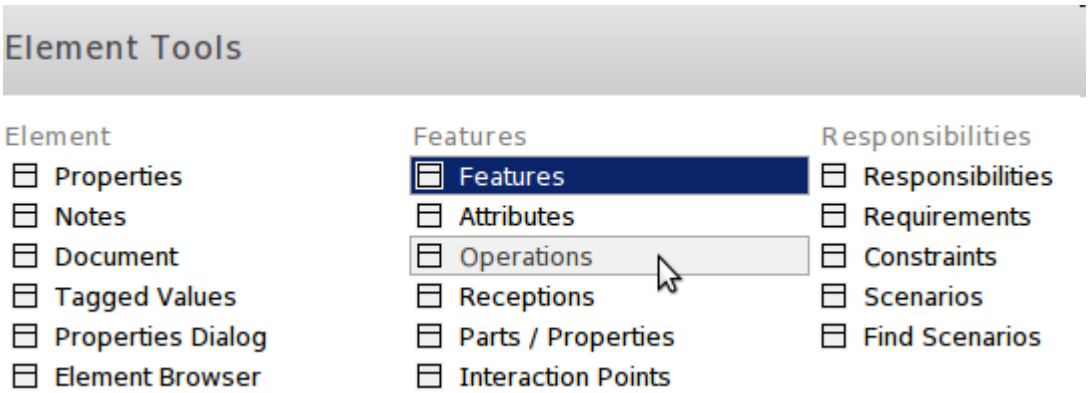
操作行为特征

块可以将操作本质上定义为块的 肌肉”，因为它是完成系统所需的大部分工作的操作。在Enterprise Architect中，工程师可以从用户界面中的多个点访问操作，但所有这些点都将打开特征窗口，其中列出了”操作”选项卡上的操作，如下所示：

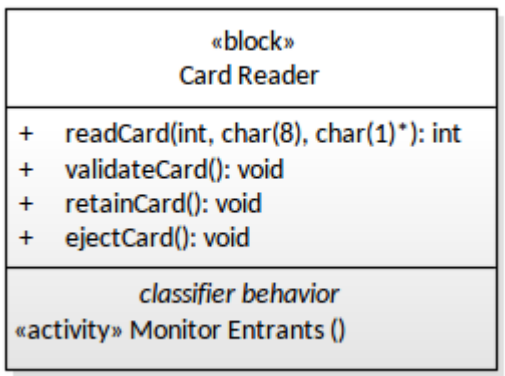


特征窗口可用作对结构和行为特征的总结，包括该块拥有的所有部件和交互点。创建操作的最简单方法是在图表或浏览器窗口中选择块，然后单击功能区项：

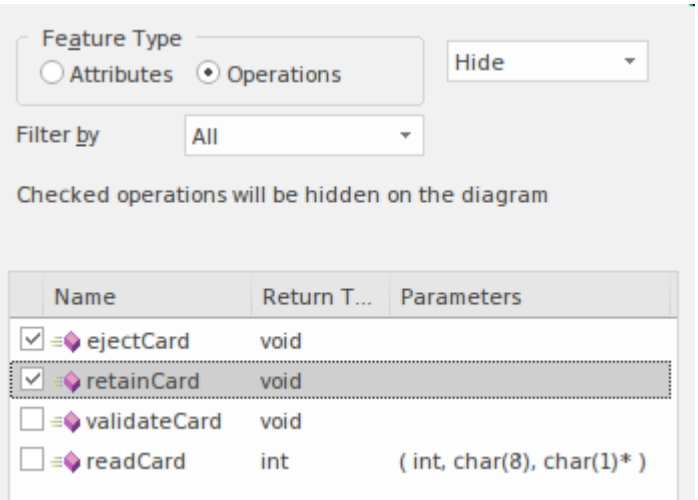
设计>元素>编辑器>特征>操作



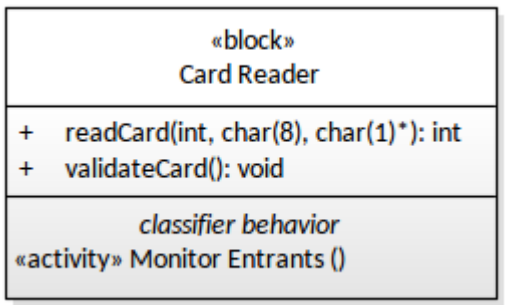
只需选择 操作”选项卡并在窗口的一行中添加名称和其他详细信息即可创建操作。可以创建任意数量的操作，并且每个操作可以定义任意数量的参数，这些参数指定操作的输入和输出。当我们描述活动参数和行动销之间的关系时，它们的重要性将在本节后面讨论。操作也可以显示在图表中，可以单独显示，也可以与其他特征一起显示，每种类型都显示在单独的隔间中父元素。



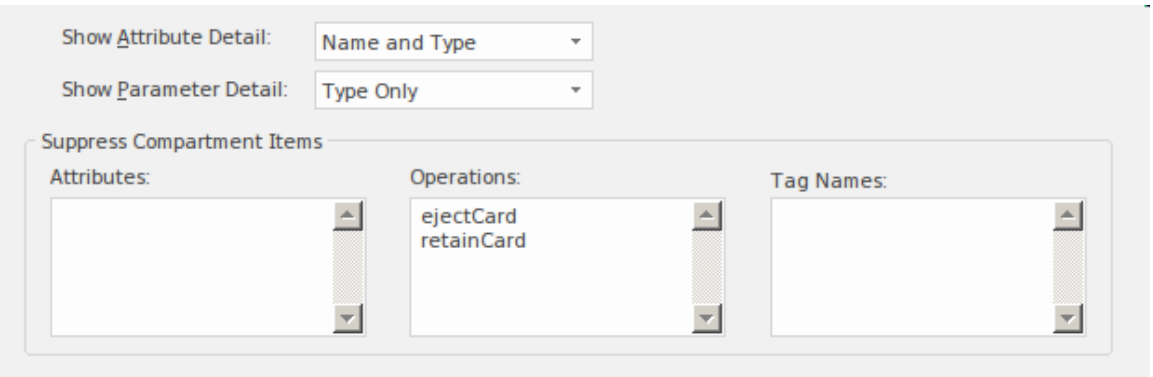
有很多选项可以控制操作的显示方式，包括不显示整个隔间或通过禁止显示其他操作来仅显示特定操作的能力。



这将导致选定的操作被隐藏在图表上，这是一种非常有用的演示设备，因为它可以帮助工程师创建专注于块的特定方面的图表，抑制或隐藏不相关和分散注意力的内容。此图表片段显示了抑制操作的结果：



对于元素级别的属性也可以这样做，并且可以使用类似的函数来抑制图表级别的特定操作、属性和标记值。当一个特定的操作出现在多个块上并且他们想要为图中的每个元素抑制它时，工程师可能会使用图表级函数。



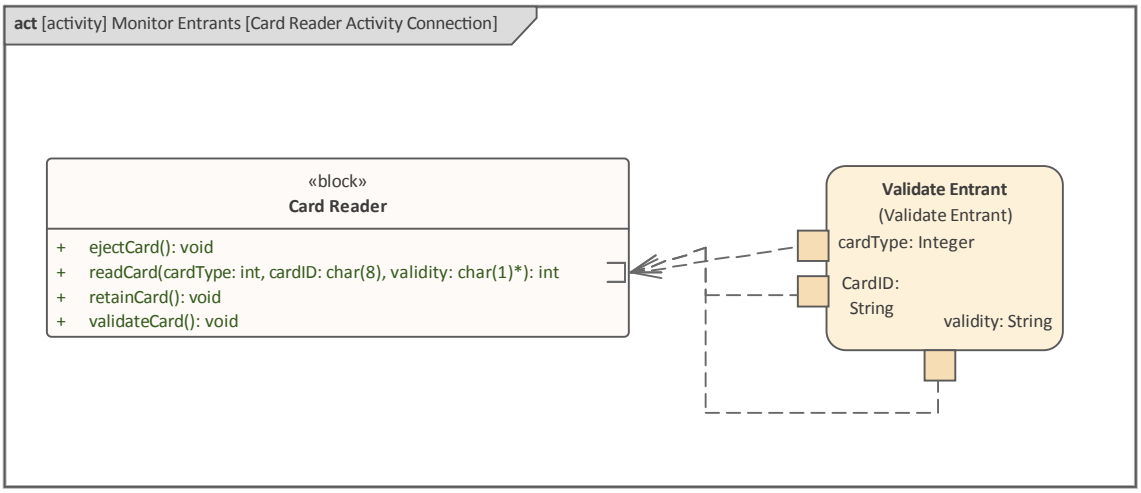
操作可以以同步或异步两种模式调用，并且可以根据编排系统行为的行为类型以多种不同方式启动，**包括**：

- A调用操作行动（调用一个活动）
- 信息作为A**系列的交互**（序列图）
- 状态机A

这意味着该操作可以在一系列 SysML 图表中可视化，并且在不同的上下文中会以不同的方式出现。例如，**在消**息在块实例或其他分类器之间发送的序列图中，操作将作为块传入消息之一的注释出现，以表明操作将作为消息的结果进行初始化。 Enterprise Architect允许工程师直接从该图表访问块的操作列表，并且还允许直接从该图表创建操作。

Usage Type	Diagram Type	Diagram
Classifier	Sequence	Card Reader and Control Unit
Classifier	Sequence	Interactions
Classifier	Sequence	Interactions Basic
Link	SysML Block Definition	Verify Entrant
Link	SysML Block Definition	Boom gate Dependencies
Link	Activity	Card Reader and Controller

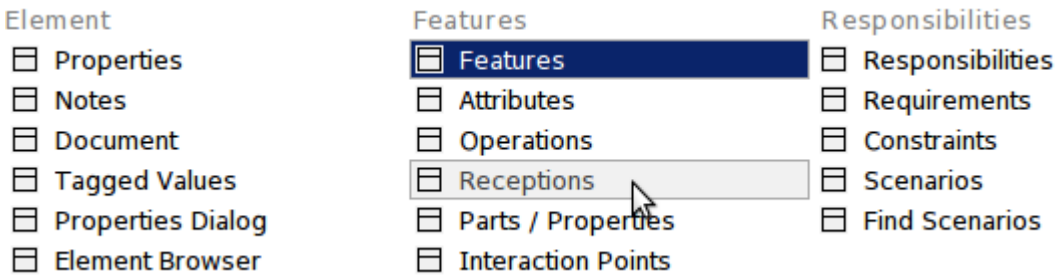
在调用操作行动的情况下，元素的销必须按类型和名称与被调用操作的参数对齐； Enterprise Architect使用"特征链接"功能帮助您在图表上可视化此映射。

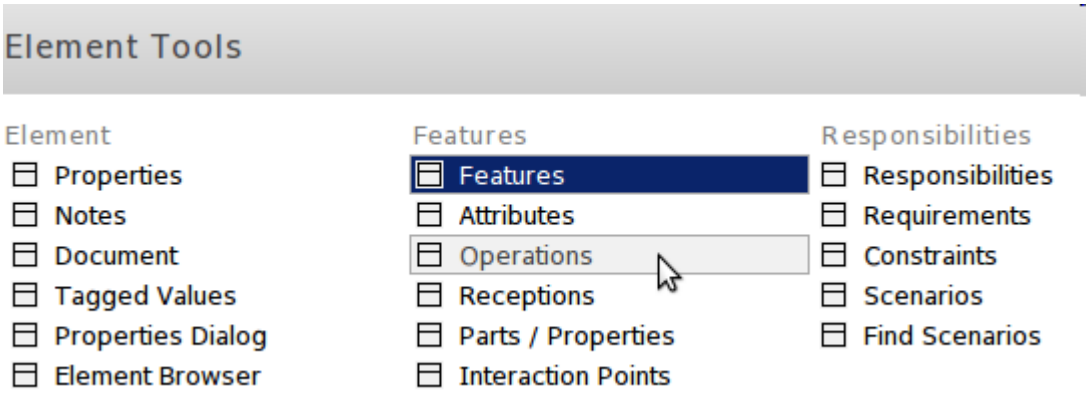


信号接收作为行为特征

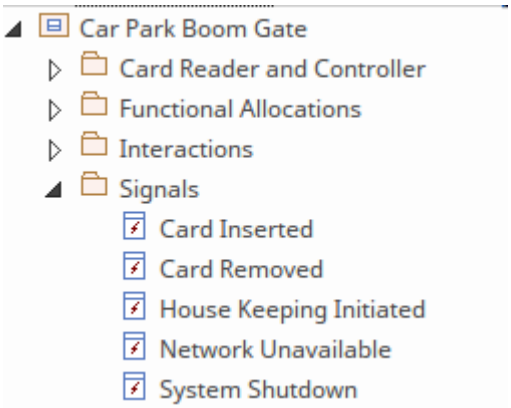
信号接收是块的另一个行为特征，但是，与操作相比，信号接收只能被称为异步。信号接收也与操作不同，因为操作重新调用具体标识要调用的操作，而接收信号的实例被认为是对引用该信号的接收object的任何接收的请求，或者它的任何直接或间接概括。通过这种方式，调用元素和接收之间存在一定程度的间接性。接收具有A接收所引用的信号的属性相对应的参数，并且这些被认为是接收的 输入"参数。

创建接收的最简单方法是单击图表或浏览器窗口中的块，然后选择功能区项 设计>元素>编辑器>信号接收"。

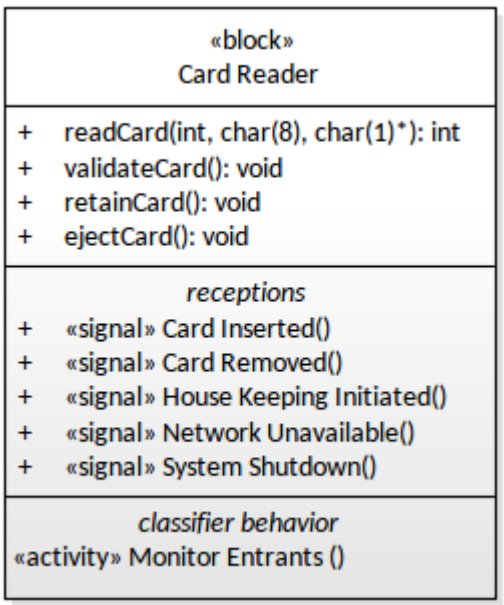




要创建一个新的 Reception，您必须首先创建适当的信号来关联 Reception。创建接收时，系统将提示您在浏览器窗口中找到适当的信号，如下所示：



与操作类似，信号接收可以显示在图表上一个块中的专门隔间中。可以定制显示并抑制所有信号接收或配置显示哪个特定信号接收。在这个屏幕截图中，工程师决定让所有信号接收可见，但是每个图表和图表中的每个块都可以进行不同的配置。

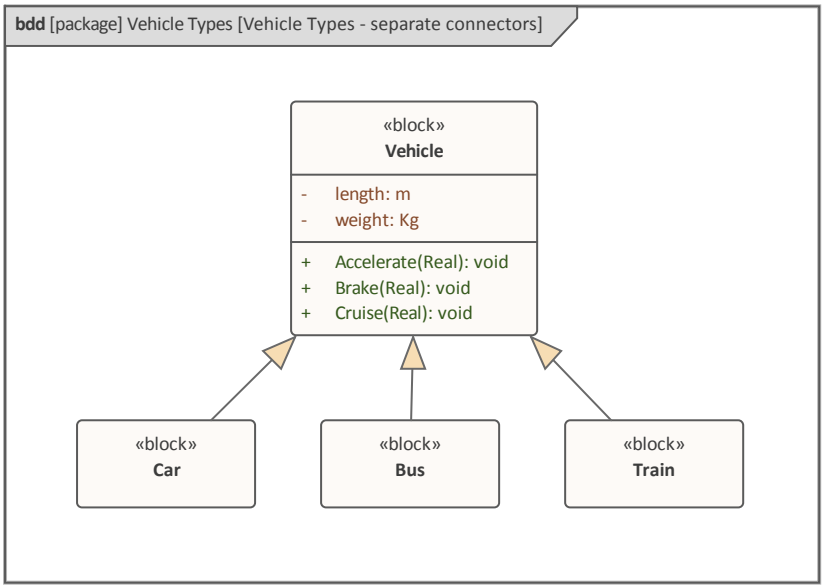


其它块关系

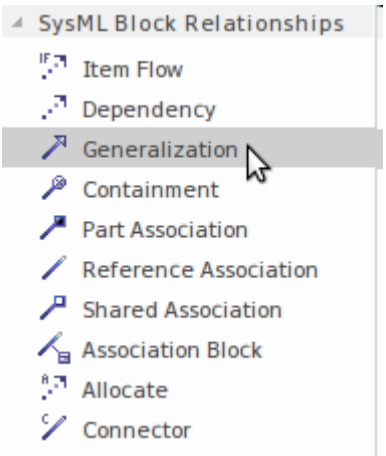
块是 SysML 中的关键结构元素，可以参与各种关系，其中一些已经在指南的前面部分中讨论过，而我们正在讨论关联。定义块时可以使用许多其他关系。

概括关系

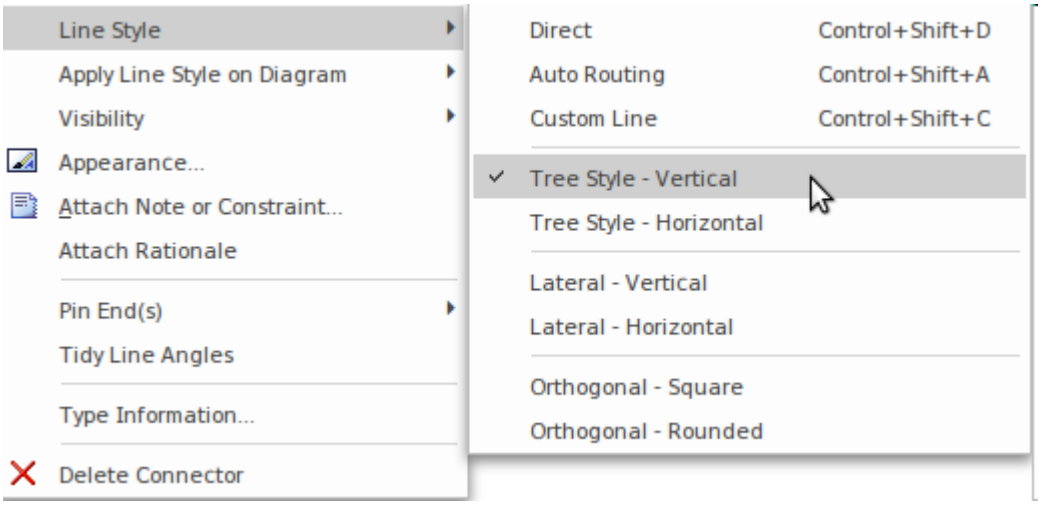
在前面提到的部件关联的关联关系类型中，还有另一种关系最概括的类型，但我们也非常强大，并且本质上用于模型块（和其他分类器）属于同一个家庭。 分类器”这个词来自我们的自然语言，例如汉语和泰语，它们具有对具有相似特征的名词进行分类或分组的抽象方式；例如，带和路是细长的东西，而浆果和球是round的东西。概括也是如此，使用深度关系和结构来对事物进行分类也可以是任意的关系。在许多方面，工程师更自然地理解反向关系并说某事是某事的专门版本。



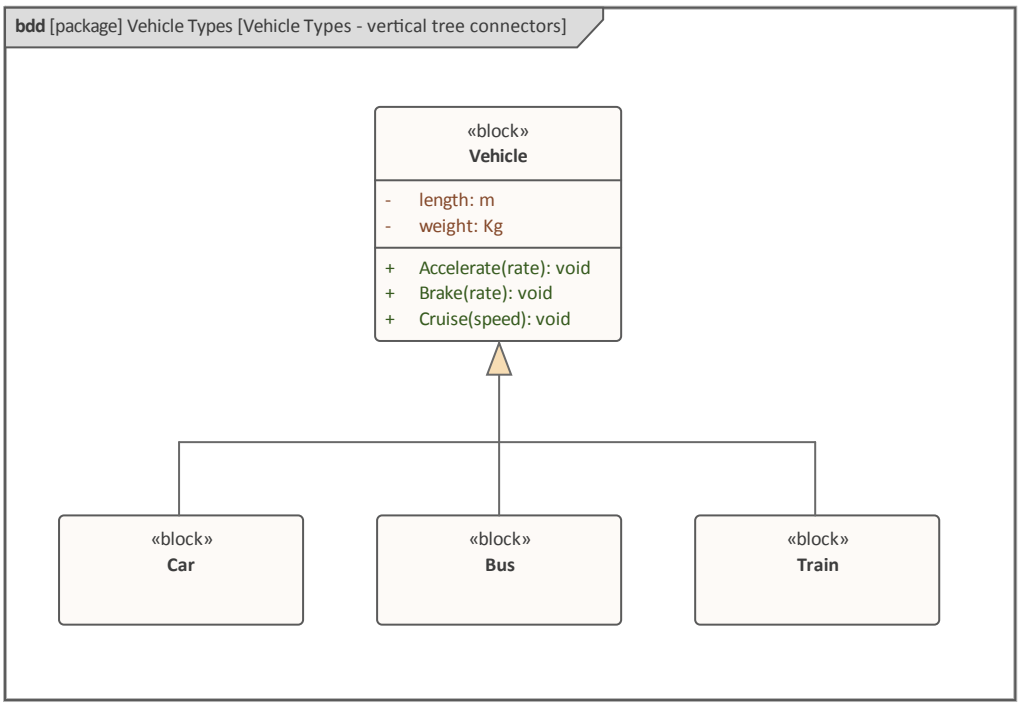
Enterprise Architect允许工程师为块、值类型、信号、接口、活动等创建这些分类层次结构。图表通常包含A系列。



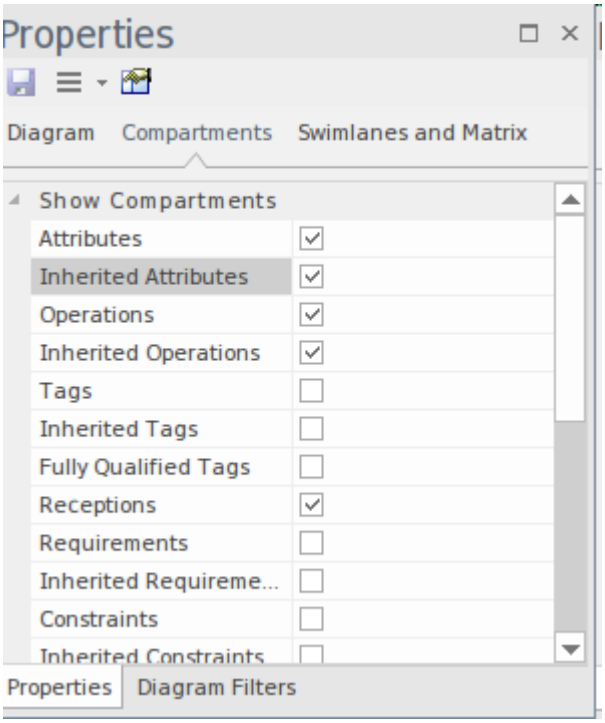
先选择概括中的 关系”工具箱，然后从更专业的元素拖放到更通用的元素，即可绘制关系。或者，这可以使用快速链接器来完成。



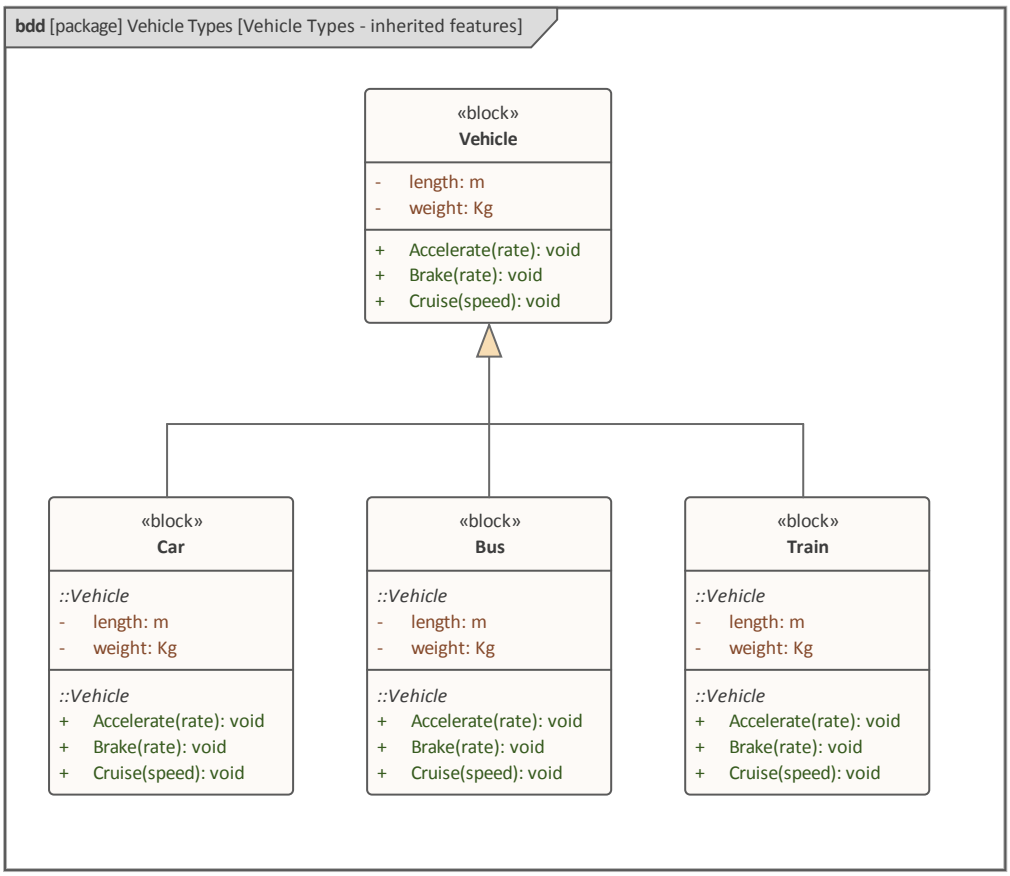
当一个块参与泛化层次结构并具有许多特化时，从该块发出的连接器可能会变得不整齐。 Enterprise Architect提供了一种将线条样式更改为多种样式中的任何一种的机制，但最有用的样式可能是垂直方向的树样式，它将关系的头部组合在一起并允许它们的尾部平行对齐.



有用的语言机制之一是从概括元素继承结构和行为特征的通用元素。到目前为止，在示例图中，工程师选择不显示这些继承的特征，但可以使用元素属性表的隔间部分将它们设置为显示。



结果将是专用块将显示从父块继承的属性和操作。这些将按父块的名称分组显示。这种机制在软件工程中被广泛使用，但对系统工程师也很有用，其中专业块由于是 家庭成员"而自动继承其父级的特征。就像在人类家庭中，一个专门的块（孩子）可以覆盖从父母那里继承的结构或行为特征。

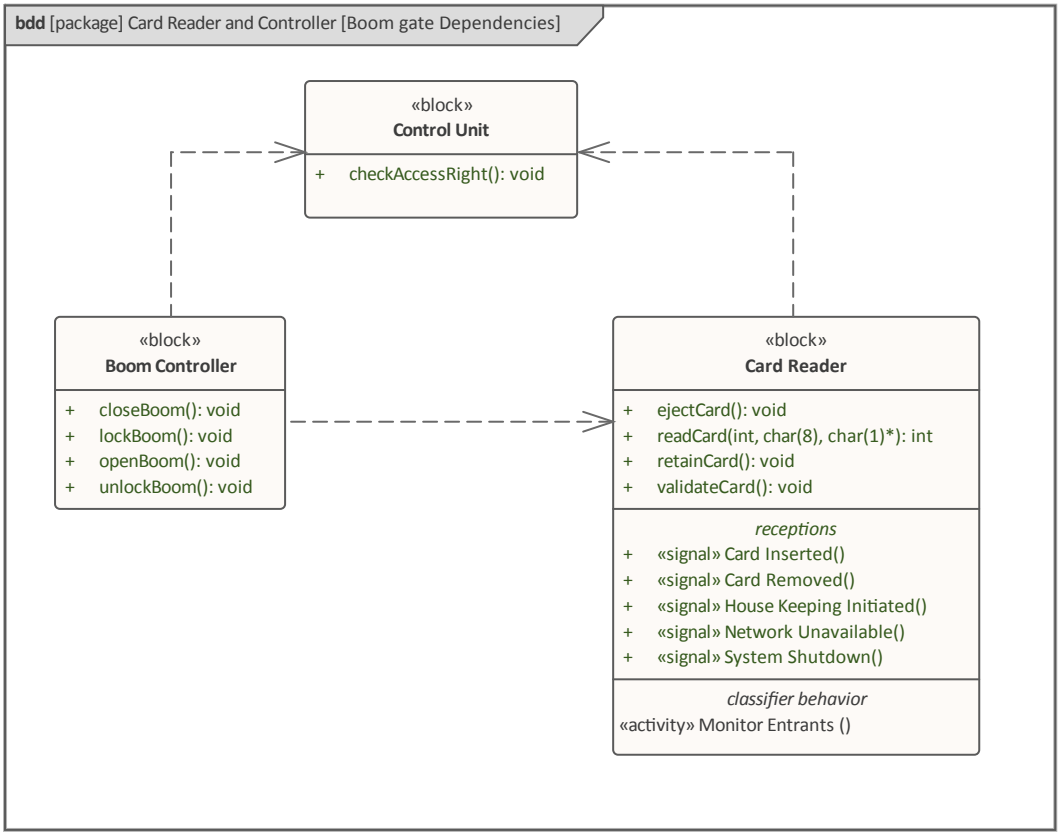


块基于某些标准，可以使用这个概括来建模，这是一种用于定义家庭成员资格的机制。

依赖

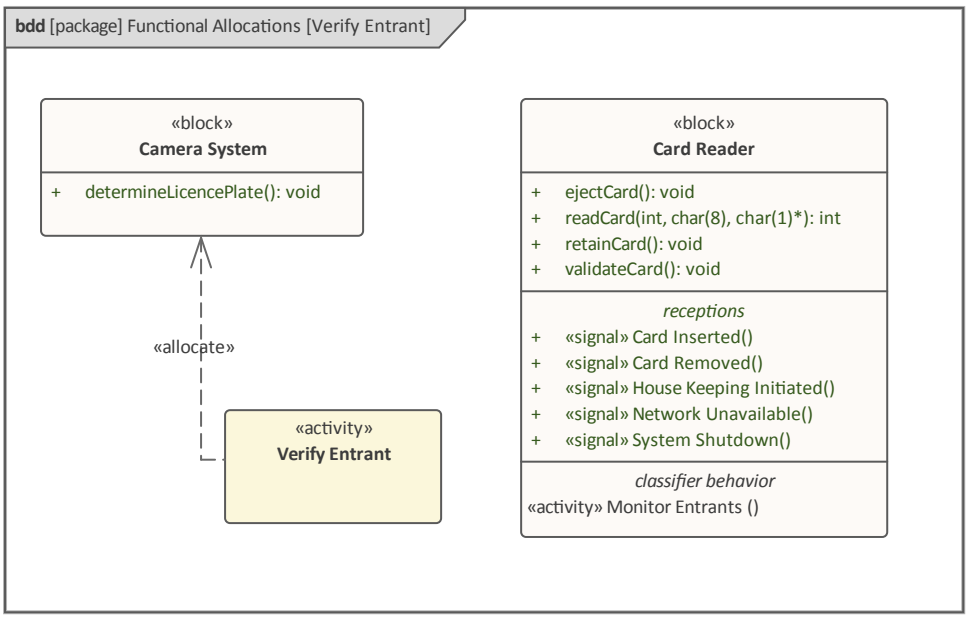
Dependency 是一种有用但语义较弱的关系。它是工程师关系工具包的 “棋子”，通常在建模过程的早期使用，当时系统元素之间的关系细节尚未分析或根本不知道。它模拟了这样一个事实，即关系尾部的元素（客户）以某种方式依赖于关系箭头端的元素（供应商）。可以原谅新手建模者以相反的方向绘制这种关系，因为轶事材料通常被认为是从供应商到客户的方向传递。一旦理解了关系的语义，并意识到关系没有说明流向，就不会犯错误。

Enterprise Architect支持多种依赖类型。可以通过在工具箱的 “SysML块关系”页面中选择 “依赖”图标然后单击客户端（尾端）元素并将光标拖动到供应商（箭头端）来元素连接器.也可以使用所选图表元素右上角的快速链接器箭头创建连接器。一旦创建了关系，就可以从连接器的属性窗口中选择一个构造型，以使依赖关系更加具体。此屏幕截图显示了所有可用的构造型，其中一些用于块以外的不同类型的元素之间；例如，包和需求。



在块和活动之间分配

分配关系可以在多种情况下使用，但它对于表达两个最典型的行为和结构元素之间的基本关系特别有用，即活动和块。这类似于我们的自然语言，如果没有名词来执行动词所描述的动作，动词是没有意义的。这种类型的分配称为功能分配，工程师通过找到可以执行活动描述的行为的块来弥合系统的这两个方面之间的鸿沟。



在此图中，工程师创建了两个功能分配关系，用于描述活动验证参与者中指定的工作将如何执行。一个关系以用于捕获车辆牌照的摄像头系统为目标，以确定特定车辆是否已获准进入。其他关系以读卡器块为目标，用于确定卡所有者与停车站有关系。

建模交互

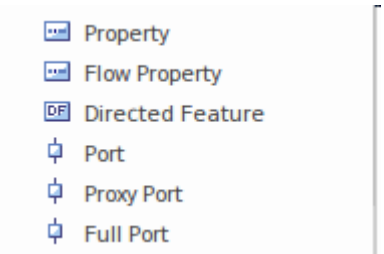
块 - 以及由块键入的部件 - 驻留在环境中，并将与该环境及其包含的其他元素进行交互。一般，交互提供了一种**交互机制**，称为提供点，作为入口和出口的边界位置，用于与自己的元素进行通信。是元素端口的一种类型，用于块的点。暴露它的行为——要么是它天生拥有的，要么是由它的部分提供的。端口由安装在块或部件边界**上的小矩形**（通常为正方形）**表示**。 SysML 目前支持两种类型的端口，旨在最终取代 Flow端口和标准端口的早期概念：

- **代理端口**-充当中继以公开由拥有块提供的行为特征，并由描述这些服务的接口键入
- **端口**充当部件并由一个块端口键入，这意味着它本身可以包含部分

Enterprise Architect完全支持这两种类型的端口，并且向后兼容早期的标准和流端口（它们仍然可以使用，**但在**标准的后续版本中将被弃用）。

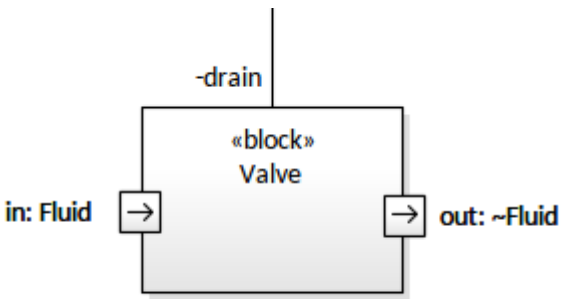
一旦创建了块定义图并在图上放置了块，端口可以由以下任一方式创建：

- 将适当的端口从工具箱到块上 - 此图显示了列出端口的块定义工具箱



- 从块的上下文菜单中选择 新子元素”选项，然后选择适当的端口类型，如图所示：

无论您选择哪种方法，端口都会自动放置在块的边界上，并且可以移动到所需位置并命名。注记可以从 设计 ”功能区的 元素”添加端口。 .此屏幕图像是显示端口元素内的两个端口方向指示器的端口图的一部分。端口分别被命名为 in”和“out”，并输入了 fluid”，表示到达端口的项目类型。

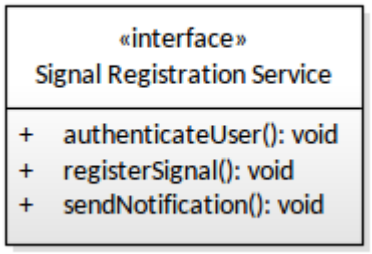


端口可以包含接口，也可以嵌套端口； Enterprise Architect提供了一个有用的特征，允许工程师自定义端口大小，有效地将小正方形更改为小矩形。

接口和端口

An接口是封装一组由块提供的服务的有用方法，提供了一种将这些服务公开给客户端的简单方法。接口具有与

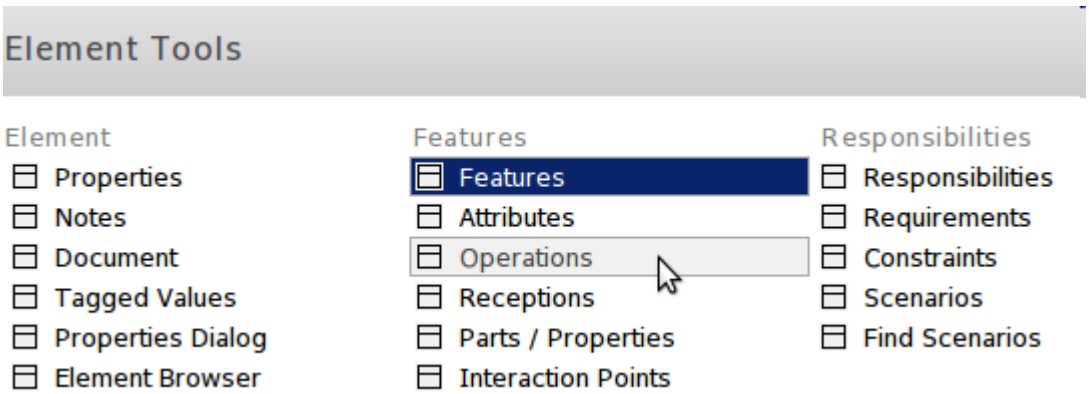
块相同的外观，可以定义操作和信号接收，但没有属性（属性）。



接口从工具箱页面拖动 接口"图标即可创建接口。行为特征可以通过从接口的上下文菜单中选择适当的特征来添加。

创建界面操作的另一种简单方法是在图表或浏览器窗口中选择块，然后单击功能区项：

设计>元素>编辑器>特征>操作



只需选择 操作"选项卡并在窗口的一行中添加名称和其他详细信息即可创建操作。可以创建任意数量的操作，并且每个操作可以定义任意数量的参数，这些参数指定操作的输入和输出。信号接收- 其他行为特征- 可以使用 信号接收"选项卡以类似的方 添加。这些项目中的任何一个都可以使用 <Ctrl>+ 向上和向下箭头键盘键重新排序。

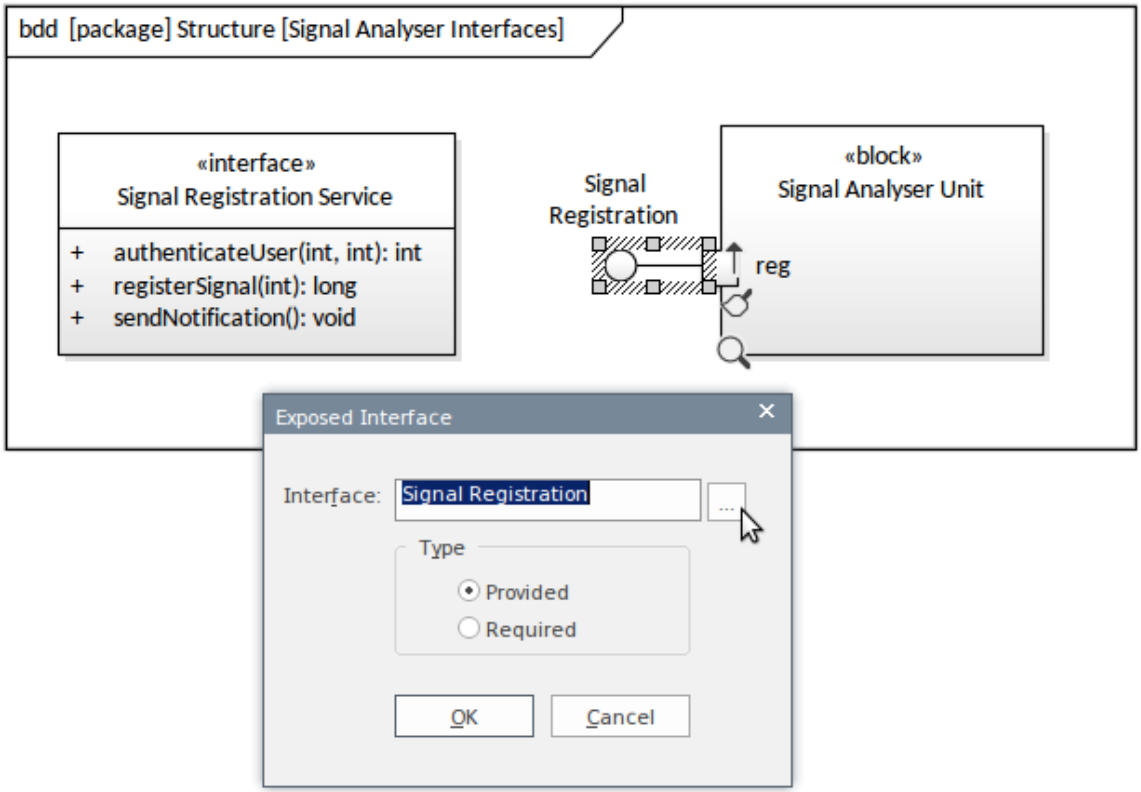
接口可以添加到端口，这是Unified Modeling Language中的一种常见机制，并提供了一种发布在端口可用的服务的方式。接口有两种基本类型：

- 提供- 可供使用
- 需要- 需要使用

在图表中一个块的接口上选择一个端口，可以将一个端口添加为任何其他结构元素- 从端口的上下文菜单项：

- 新子元素|假如
- 新子元素|需要

以这种方式添加的接口必须由接口正确类型（元素原型为 <<interface>>）。这可以通过首先选择接口中的接口，然后从接口的属性上下文菜单中选择窗口来实现。然后，您可以命名接口并使用 [...] 图标导航或搜索接口元素。此图演示了信号注册接口的步骤。

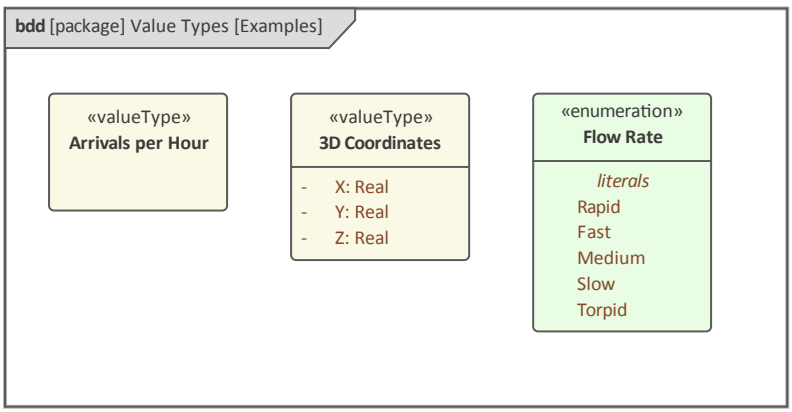


使用值类型建模数量

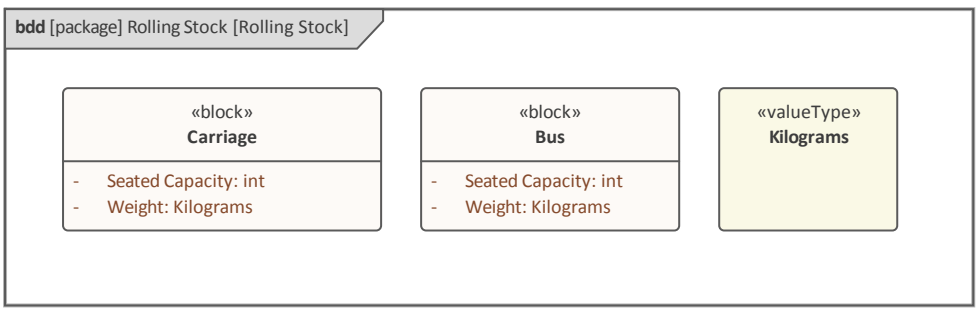
块可以具有称为值属性A简单属性，它定义表示标量或矢量的块的属性。值用于表达有关块的信息，并为要在块实例中输入的实际值提供一个槽。例如，坦克可以属性直径、高度和体积的属性。值类型允许工程师创建量化属性的通用方法；例如：

- 离心机A最大速度以每分钟转数 (rpm) 为单位
- 火车车厢A重量以千克 (kg) 为单位
- 罐A容积以升 (l) 为单位
- 光源具有以源(cd) 为单位指定A发光强度
- 透析机A血流速度以每分钟毫升数 (ml/min) 为单位

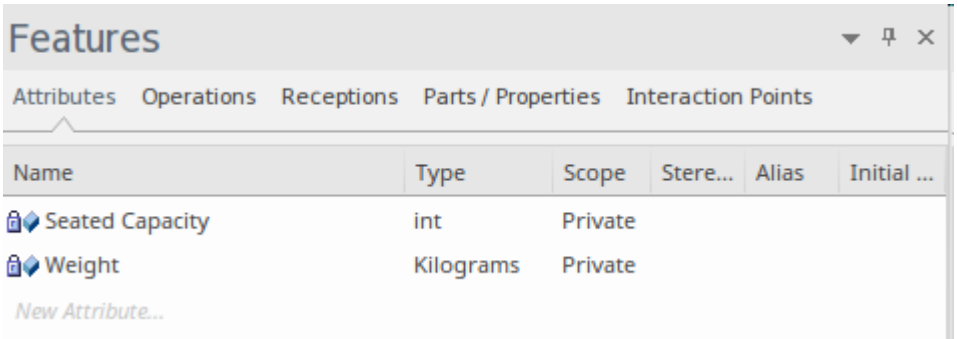
该图显示了许多不同的值类型，它们可以在Enterprise Architect中定义，然后应用于块中定义的任意数量的属性。



值类型的目的是允许工程师、团队或行业定义标准类型，这些标准类型可以重复使用来表征为任意数量的块定义的值属性。例如， 公斤”的值类型可以应用 指定火车重量或公共汽车重量或两者之一的座位容量的值属性。



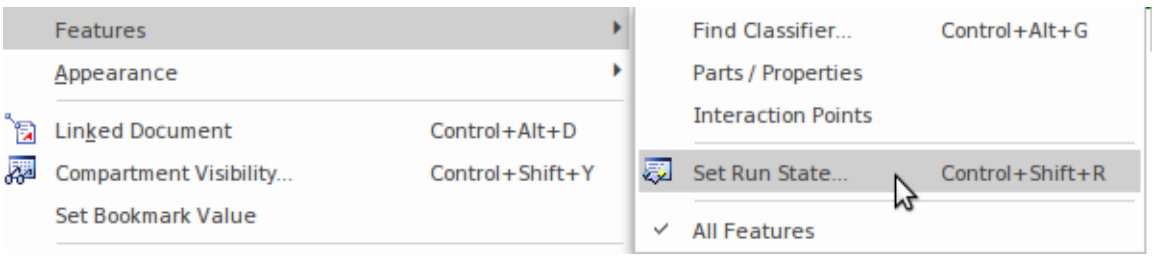
这些值类型，顾名思义，都有一个描述属性数量的属性；它们可以使用特征窗口的 属性”选项卡进行定义，如此处所示。



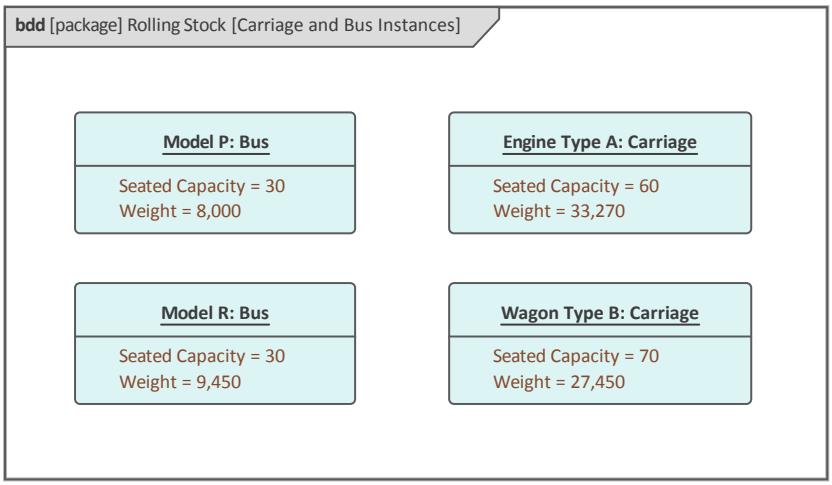
例如，代表火车和公共汽车的两个块可以具有定义的 重量"属性，该属性由值类型 千克"键入。如果 整数容量"这样的简单类型在 类型"中有一个属性可以直接在 类型"下拉列表中选择，但如果 类型"是基于值的，则可以使用 选择"选择从下拉列表中键入..' 选项。

具有在Enterprise Architect中定义的属性（值属性）的块实例可以具有为该属性指定的实际值。例如，Bus 和 Train 的每个特定模型可以定义不同的权重。诸如 其它Capacity"之类的其他属性可以具有定义的原始类型 还具有"的整数，并且可以设置这些对于 马车"的特定实例。

Enterprise Architect允许工程师通过使用块实例的 特征"子菜单中的 设置运行状态"选项来设置每个已定义属性的值。



属性值（插槽）可以显示在图表上，允许工程师创建引人注目的块实例示例或目录，如下所示。



使用属性和部件来模型块用途

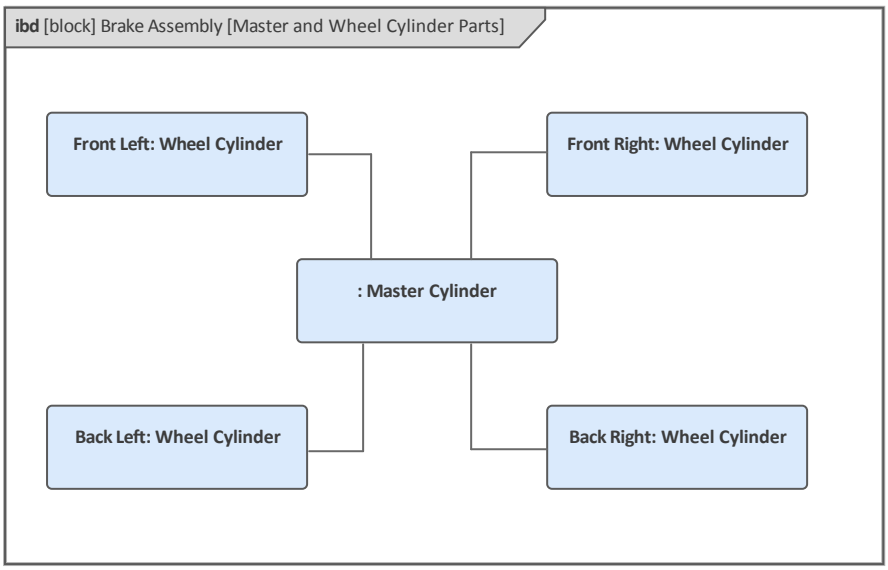
块是分类器，描述了一组元素的特征，这些元素代表了块在时间中的使用上下文。当块具有定义的属性（值属性）时，这些属性在块的实例中被赋予特定值。实际上，每个块实例都有一个身份，并且通常会分配不同的值来定义块的状态。 Enterprise Architect允许使用 生下文"菜单中的 设置运行状态"选项来指定这些值。

内部块图通常显示一个块的部分在使用上下文中是如何连接在一起的。 Enterprise Architect允许将块从浏览器窗口拖到图表上并作为部件属性拖放。这些实际上是部件并表示块分类器的实例。工程师有机会在上下文的时间点中命名这些。例如，此图表示已建模的组装，在图框架中以这种格式表示：

组装[block] Brake组件[Master and Wheel Cylinder Parts]

- *ibd* - 表示它是一个内部块图
- *block* - 表示元素类型
- 刹车组装-是元素的名字
- *Master* 和 *Wheel Cylinder*部件- 是图表的名称

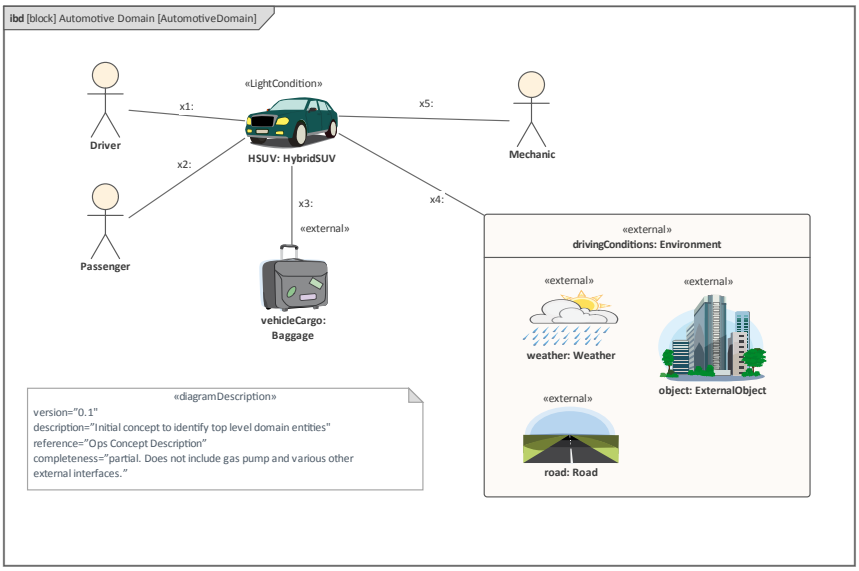
工程师已命名每个轮缸部件（左前、右前、左后、右后），因为这些部件需要根据它们在车辆中的位置进行识别，但决定不再将主缸命名为资格是必需的。



内部块图表

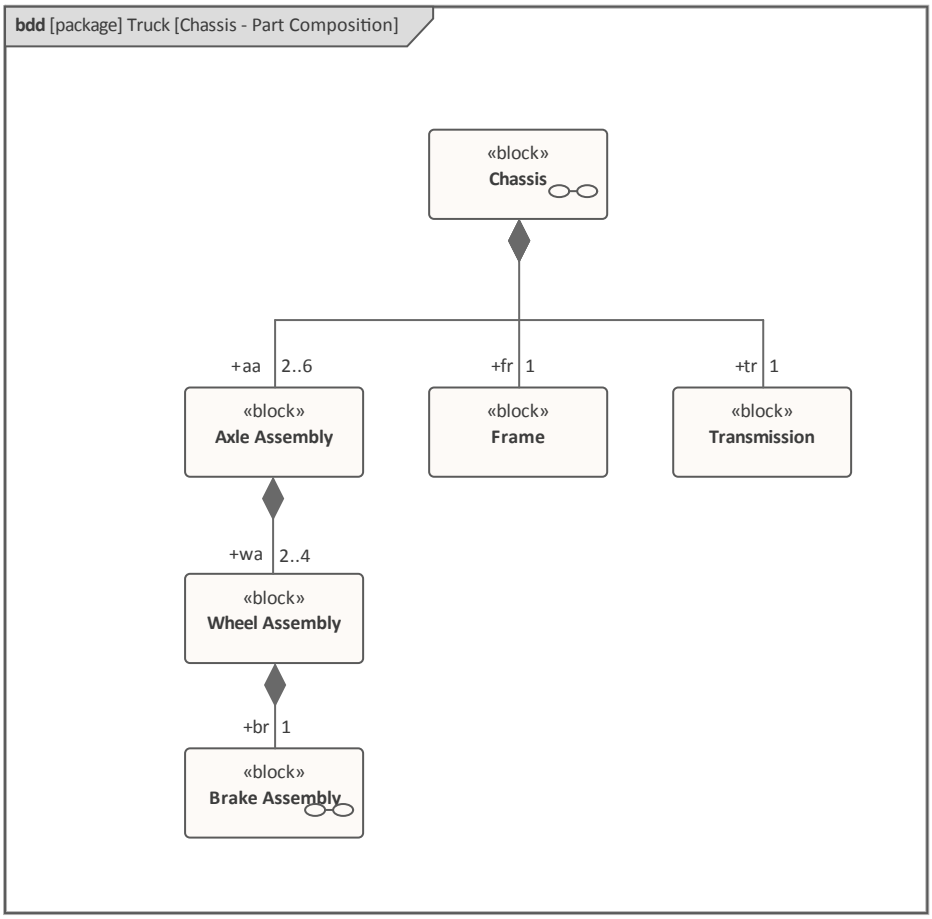
内部块图提供了一种可视化块内部结构的方法，包括其属性和部件以及这些部件相互关联的方式。图表不需要显示给定块组成的所有部分，工程师通常会创建一个专注于系统或子系统特定方面的图表。

内部块图的框架代表拥有块，因此将这样命名，并且出现在图上的元素将是组成拥有块的块的实例的部件。这个内部块图显示了给定时间的车辆上下文；它使用许多图像来代替传统的 SysML 语言符号，以此来增加吸引力并使图表对非工程观众更具吸引力。有关更多信息，请参阅Internal Block Diagrams帮助主题。

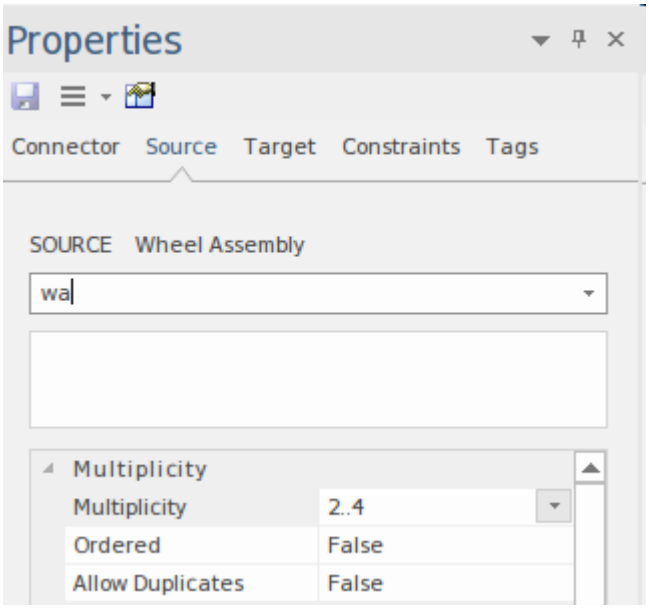


建模和连接部件

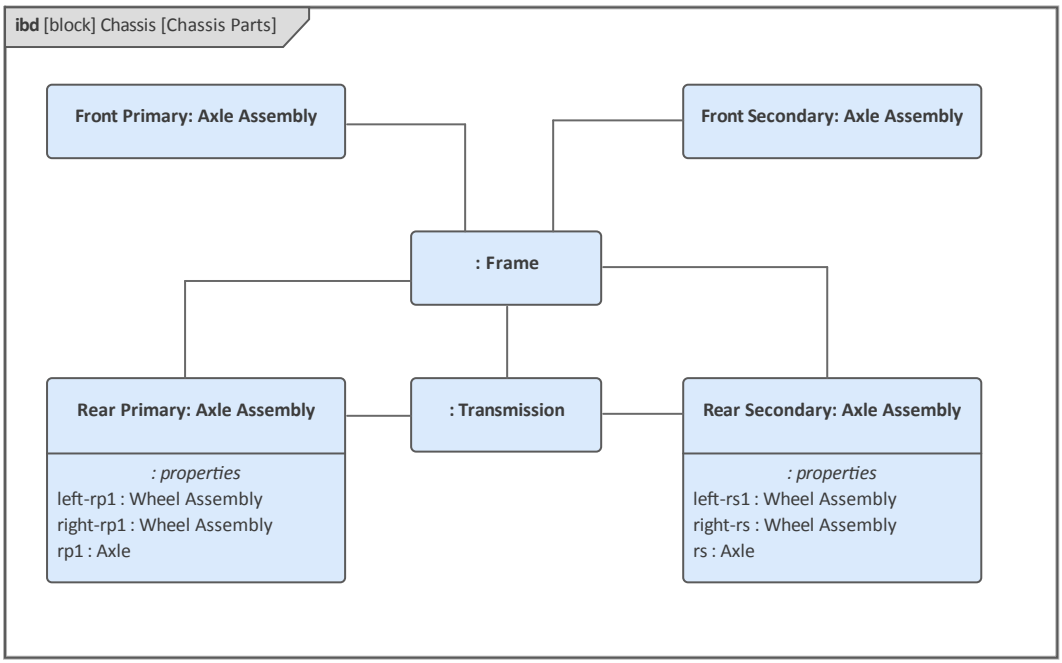
当一个块由许多其他块组成时，通常情况下，当系统实例化时，包含的块将执行拥有块所需的大部分工作。内部块图提供了一种语言机制来可视化部件如何交互，显示块在上下文上的结构并提供由拥有块指定的整体行为。在之前的主题中，我们查看了卡车底盘的块定义图，描述了基于 Blocks 构成底盘的 Blocks。



该图在关联的部分末端包括多重性，显示有多少特定部分可以包含在拥有块的单个实例中。这些数字表示以上限和下限表示的基数，可以在连接器的属性窗口中定义。此屏幕截图显示了属性窗口的一部分，用于定义关联端的多重性和其他属性，所有这些都为关联添加了丰富的语义。

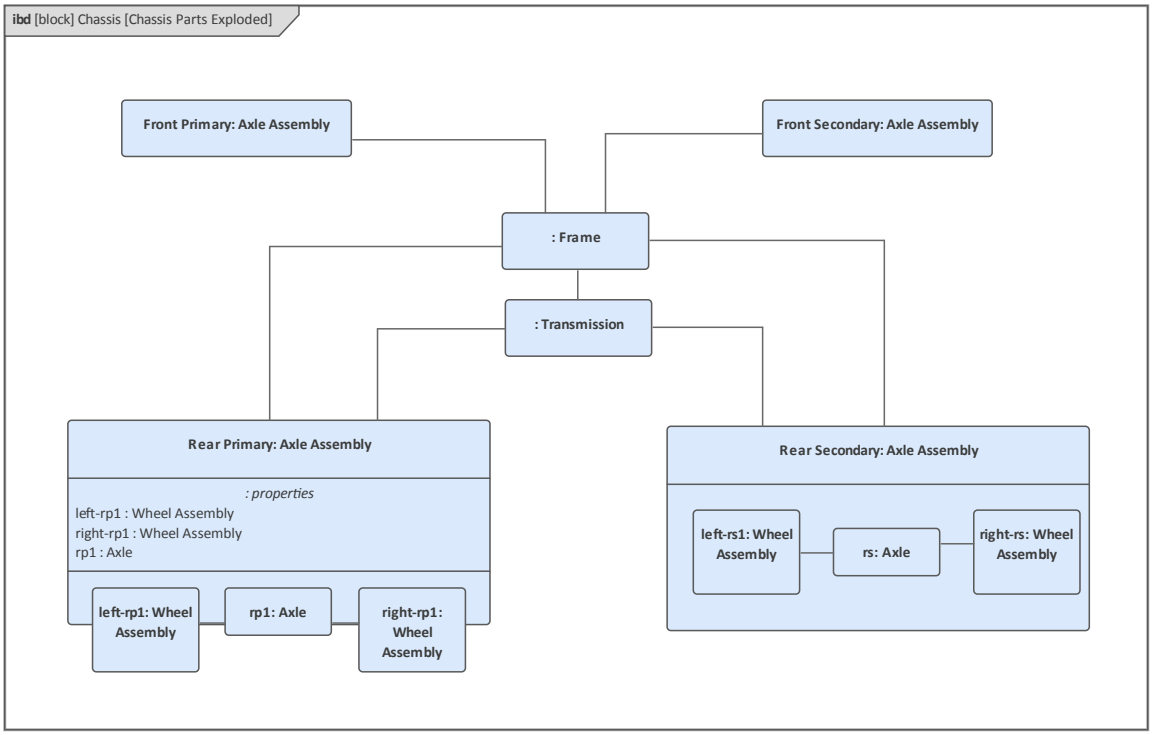


机箱内部块图显示了在实际时间中构成机箱的上下文。车轴的数量是为卡车的特定实例定义的。



Enterprise Architect允许工程师创建包含嵌套到任何级别的部件的图表，这有助于展示块的结构以及部件在现实世界中的连接上下文。

该图显示了嵌套在两个级别上的部件，但任何级别的级别都是可能的，并且可以在图上创建。这种类型的表达可以导致非常大的图表， Enterprise Architect支持最大 A0 的纸张大小，允许创建和打印大型图表。



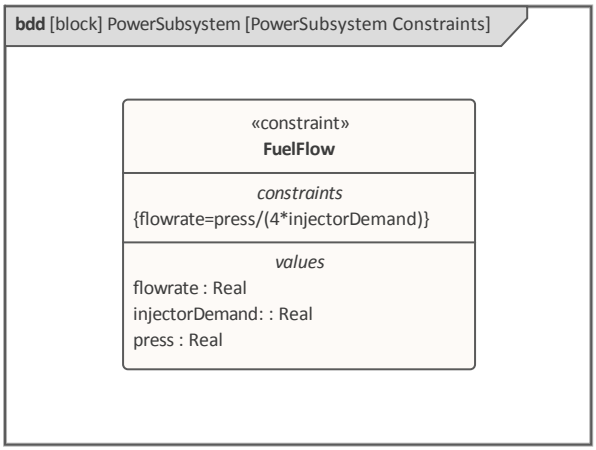
在图中，后主轴组件和次轴组件已显示在细节中，其中每个轴由左右车轮组件组成，它们本身可以显示为包含制动组件的组装结构，其中turn 可以显示轮缸总成。

如果零件已添加到图表中，但建模者出于某种原因需要更改类型块的块分类器，这可以从 零件”上下文菜单中完成，该菜单为属性类型提供选项改变了。这在创建了复杂图表的情况下很方便，并且更容易将视觉元素留在原地并更新它所基于的块。

建模参数方程

工程师负责寻找问题和机会的解决方案，并使用模型来可视化正在考虑的系统和系统需要运行的世界上下文的简化。在Enterprise Architect中创建的系统工程模型提供了一个有价值的工具用于分析、设计、架构、测试和可视化。这包括能够预测系统在给定时间的行为方式，以上下文利益相关者协商和权衡分析的形式平衡竞争需求和设计考虑因素。参数图是一种高级工具，可以帮助工程师在模型中解决这些问题，并预先表示系统可能的行为方式。

在较早的主题中，我们学习了如何使用块定义图对方程进行建模，部件关联关系阐明方程的变量（参数）。这提供了重用的基本机制。作为我们如何将约束块用于模型的复习，请参阅此图，该图使用约束块（这是一个刻板印象的块）来模型车辆的燃料。燃油流量基于以燃油压力（压力）和燃油需求为变量（参数）的方程式。



任何方程或方程组都可以使用约束来建模。

此约束可能会在许多不同的上下文中重复使用。在参数图上，我们可以看到它是如何使用的。有关详细信息，请参阅Parametric Diagrams帮助主题。

介绍参数图表

参数图提供了A在特定上下文以约束属性的形式可视化方程及其参数的方法。这些属性中的每一个都表示通常在块定义图上定义的 ConstraintBlock 的用法。

参数图的框架代表拥有的ConstraintBlock，所以它会这样命名，并且出现在图上的元素将是约束属性，它们是组成拥有块的ConstraintBlocks的实例，从而显示了组成约束。

创建参数图表

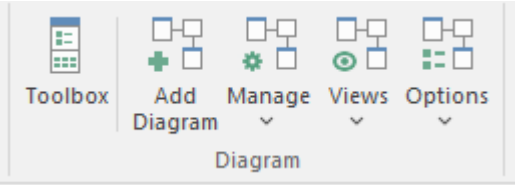
可以A以下任何选项从用户接口中的多个位置创建参数图：

- 设计ribbon - 在图表面板上添加图表图标
- 浏览器窗口工具栏 -新的图表图标
- 浏览器窗口上下文菜单 -图表图形

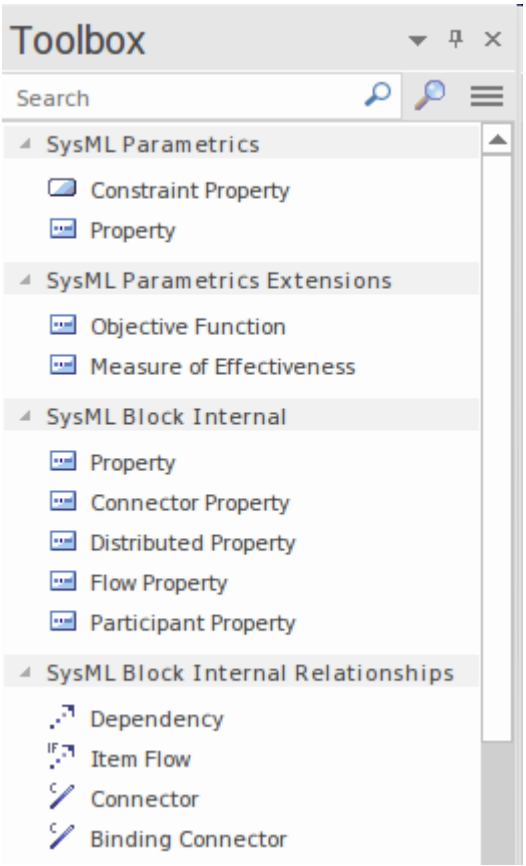
我们将使用 设计"功能区选项来创建参数图。首先，在浏览器窗口中选择要放置图表的包。

与大多数其他 SysML 图相比，参数图通常插入在其拥有的约束块下。在浏览器窗口中选择位置后，选择：

设计>图表> 添加图表



选择此选项会打开 新建图表"对话框，允许您为图表命名；该名称最初将默认为拥有该图表的约束块的名称。选择 SysML蓝图并选择 SysML 版本后，将显示图表列表，允许您选择参数图。单击确定按钮在浏览器窗口中选择的位置创建一个新的参数图。图表视图将打开，允许您开始添加描述方程和参数的元素和连接器。 Enterprise Architect还将显示工具箱的 参数"页面，其中包含图表规范定义的元素和关系，以适用于构造此图表类型。如果需要，除了默认显示的 公共"元素和 公共关系"页面并允许添加图表注记、图例和其他常见元素之外，还可以打开任意数量的其他工具箱页面。



与参数图一起使用的最重要的元素和连接器是：

元素

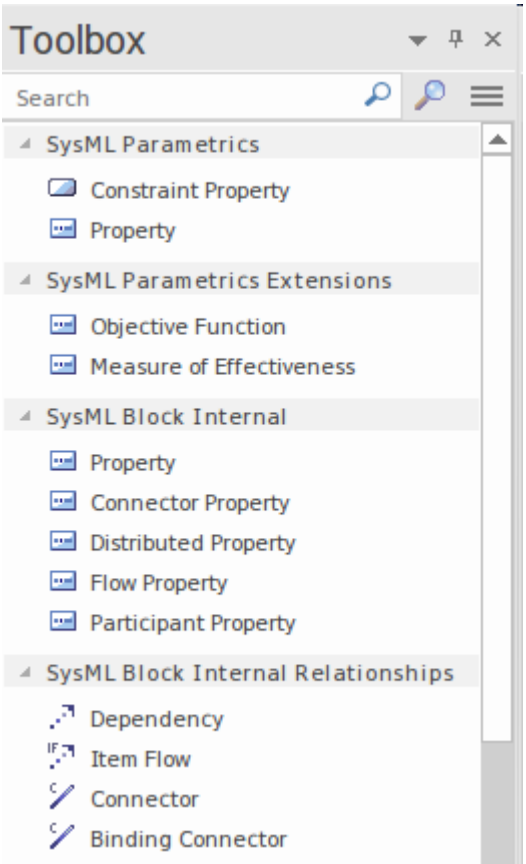
- 约束属性
- 属性

元素扩展

- 目标函数
- 有效性测量

连接器

- 依赖
- 项目流
- 连接器
- 捆绑连接器

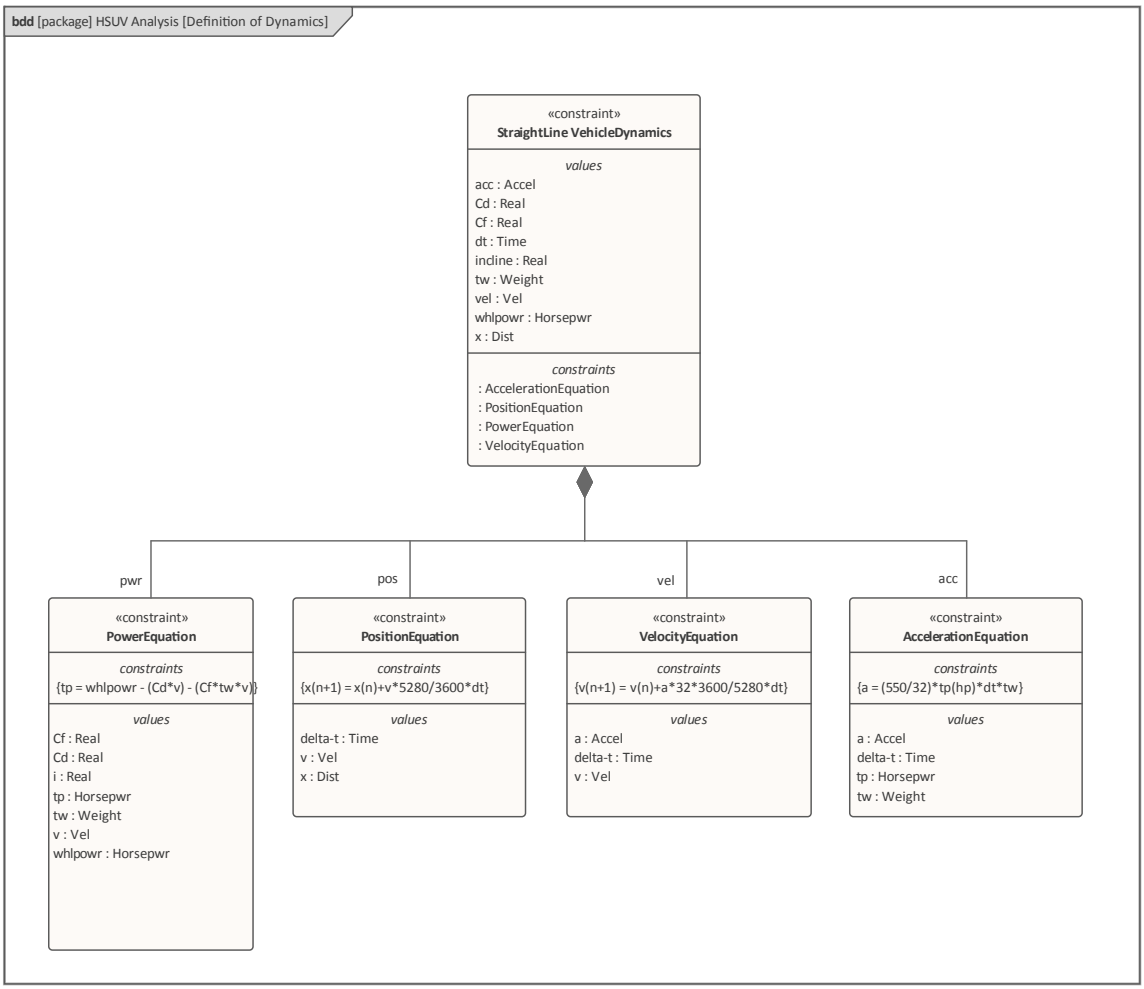


可以通过将元素从工具箱页面拖放到图表画布上来将元素添加到图表中。

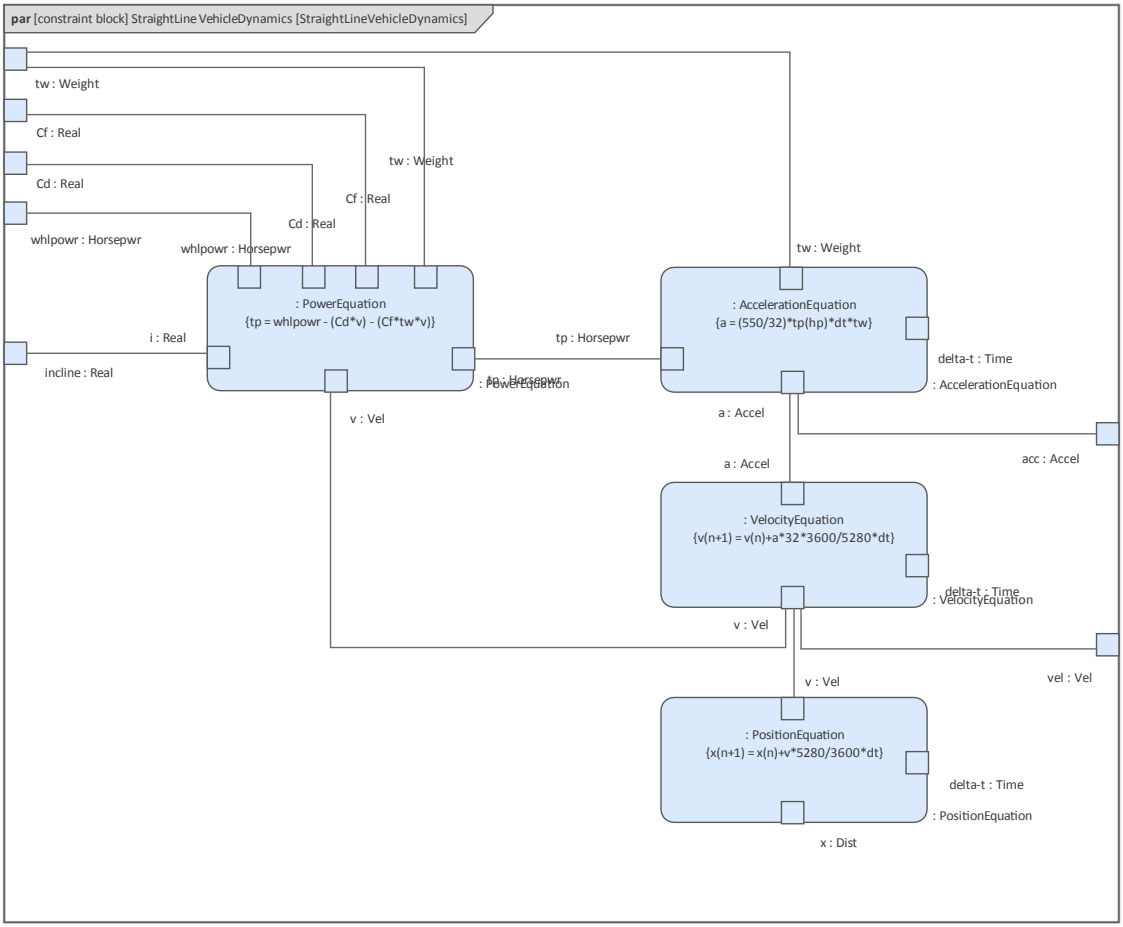
使用元件连接的部件关联

工程问题和系统通常需要详细分析以确定特定提议的解决方案将如何执行。分析可以涉及任何数量的方程，这些方程通常相互关联以确定特定值。 Enterprise Architect允许工程师使用块定义图构建方程组，然后在多个参数图中使用这些方程来描述建议的解决方案。

此块定义图描述了车辆的直线动力学参数 - HSUV - 并包括在图表上表示为约束的许多方程，它们是一种类型的块。

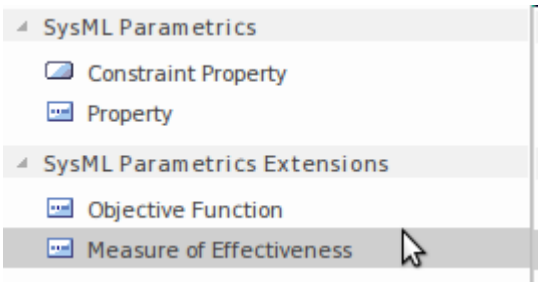


下一个图表，一个参数图表，显示了 ConstraintBlocks 如何在特定上下文使用，在图表上表示为 ConstraintProperties。我们可以想象总功率参数是如何计算的，功率方程和加速度方程上的等效参数之间存在联系。可以看到位置方程和速度方程之间的联系，最终连接回加速度方程。


par [constraint block] StraightLine VehicleDynamics [StraightLineVehicleDynamics]
tw : Weight
Cf : Real
Cd : Real
whlpowr : Horsepwr
tw : Weight
Cf : Real
Cd : Real
whlpowr : Horsepwr
: PowerEquation
{tp = whlpowr - (Cd*v) - (Cf*tw*v)}
i : Real
incline : Real
tp : Horsepwr
: PowerEquation
v : Vel
tw : Weight
: AccelerationEquation
{a = (550/32)*tp(hp)*dt*tw}
delta-t : Time
: AccelerationEquation
a : Accel
acc : Accel
a : Accel
: VelocityEquation
{v(n+1) = v(n)+a*32*3600/5280*dt}
delta-t : Time
: VelocityEquation
v : Vel
v : Vel
vel : Vel
: PositionEquation
{x(n+1) = x(n)+v*5280/3600*dt}
delta-t : Time
: PositionEquation
x : Dist

使用参数的有效性度量

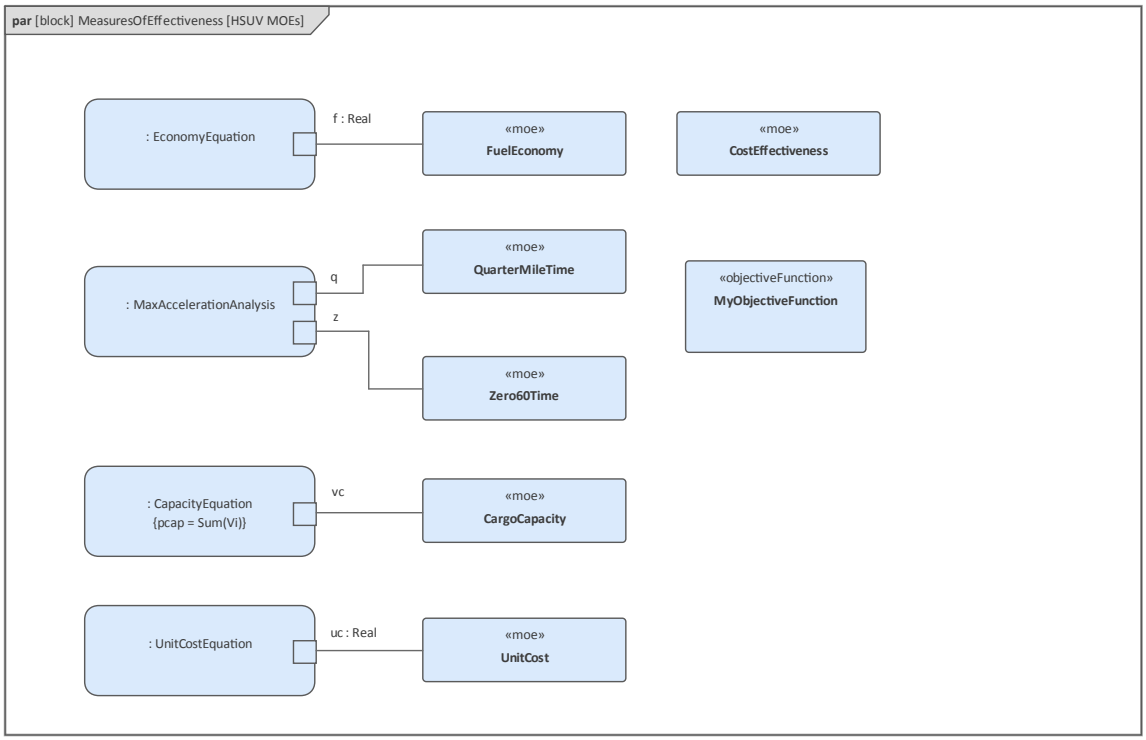
有效性测量 (MOE) 是一种重要的工程机制，用于量化任务目标或特定期望结果的实现情况。它们可以使用参数图进行建模，而有效性元素的度量可从工具箱的 参数"页面获得，从中可以将其图表图表上并与表示为约束属性的方程的参数相关。



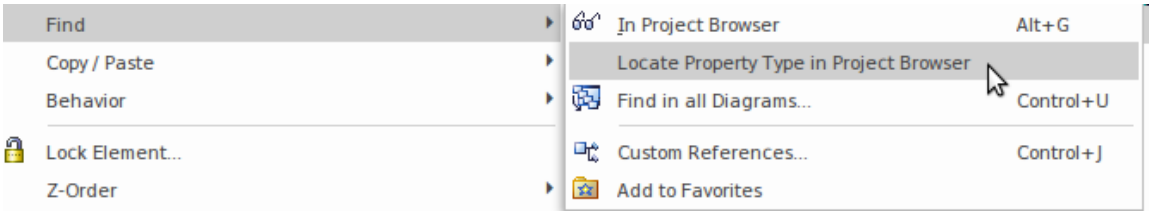
有效性度量可用于评估任意数量的设计备选方案，并允许系统地比较和评估这些设计。此图显示了混合动力 SUV (选择) 的总体成本效益，基于多个 MOE，包括：

- FuelEconomy -燃油经济性的表达
- QuarterMile -时间*0.25* 英里所用的时间
- Zero60Time -时间到 *60/mph*所需的时间
- CargoCapacity -货物空间的体积
- UnitCost -车辆成本

这些中的每一个都有自己的参数模型，能够确定有效值并对整个方程做出贡献，这是一系列加权和{CE = Sum (W i*P i)}。 Enterprise Architect允许定义任意数量的备选方案，并且工程师可以重用 MOE 元素和用于定义相关方程的 ConstraintBlocks。



Enterprise Architect有一个有用的搜索特征，它允许在浏览器窗口中定位属性所基于的类型。此函数对于查找在参数和内部块图中找到的块或 ConstraintProperty 的拥有块或约束特别有用。



可以使用 Focus 窗口的 模型视图"选项卡设置幻灯片放映。有关更多信息，请参阅Diagram Slide Show帮助主题。

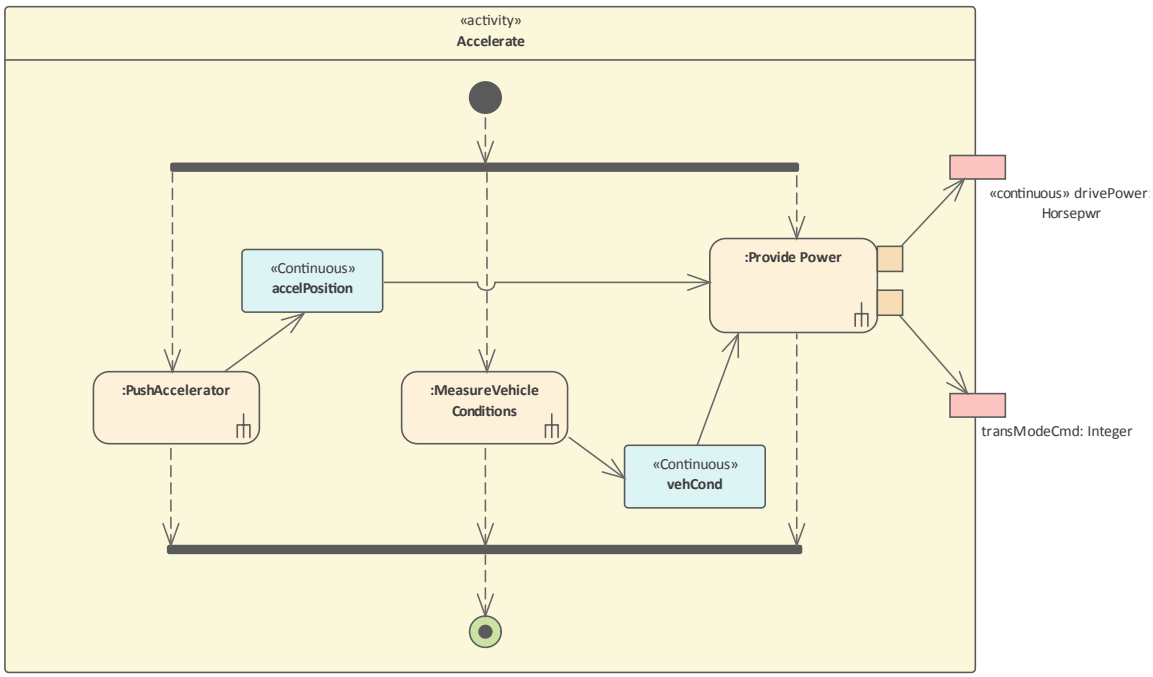
协调行为与活动

正如在前面的主题中所讨论的，系统建模语言（SysML）有两个基本方面，类似于人类用来交流的自然语言中的两个重要语法类别，即名词和动词。在 SysML 中，这些是结构和行为构造；结构结构类似于自然语言中的名词，行为结构类似于动词。

当我们讨论包和块时，我们在前面的主题中提到了语言的结构方面。我们现在将现在转向主要的行为图，即活动图。还有许多其他的行为图，实际上行为在结构图中以操作的形式可见，并且在直接分配给块的行为中也是可见的。

虽然 SysML 的新手，第一次查看活动图，可能会想起流程图，但他们很快就会知道活动图的语法和语义远远超出了流程图。活动图正式基于称为*Petri Nets*的数学分支，它使用标记系统来指示动作的序列以及流经系统的项目。流动的项目可以是信息项目、物理项目甚至是控制信号。我们将引用这个令牌系统作为阐明活动图工作的一种方式。

该图描述了车辆的加速度，显示了活动图中常见的许多元素。您将在后续主题中看到，这是一个非常富有表现力的图表，如果精心制作，可以严谨地传达大量信息。



事实上，活动图的语法是任何 SysML 图中最丰富的语法之一，当您添加Enterprise Architect包含的用于处理这些图的丰富而有效的机制和工具时，建模者就有机会表达它们本身使这些成为系统表示中最通用但也最具挑战性的部分之一。

SysML活动图基于同名UML图，但在两个方面增加了额外的语义：

- 连续流，允许限制实体在活动中沿边缘流动的速率，以及确保最新信息可用于行动的机制
- 概率，引入到活动中以包括一个值可用于参数集的边缘或输出的可能性

虽然该图可以说是基于我们自然语言中的动词序列化机制（动词串与名词连接在一起），但如前所述，它的正式起源于称为*Petri* 网和令牌流的数学分支。建模者必须理解语言的标记流方面，并且能够学习可视化这些流经物件流的不可见项目，这些项目被滞留在缓冲区中，并受其他语言机制控制，这些语言机制指导项目如何从行动中流动。如果没有这种理解，就很难解释一个活动图，包括如何控制序列的行动、如何消耗输入以及如何创建输出。

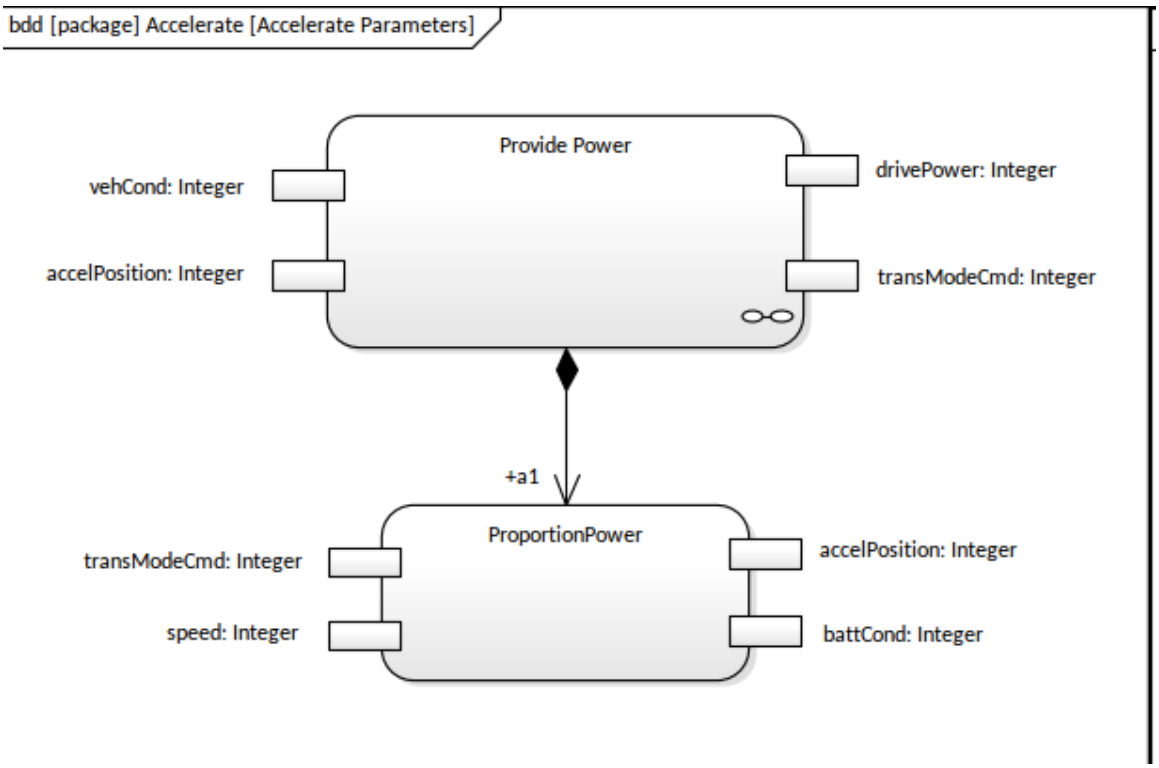
活动图与它们的近亲（如流程图或进程流程图）之间的显着区别在于能够在这些行为元素和结构元素之间建立关系。

系统工程学科的A基本方面是功能与形式函数的能力，而且还能够在它们之间创建映射，从而暴露与架构和设计的这两个组成部分相关的接缝。大规模、复杂系统工程问题的经验证据证明，这种方法可以带来深远的好处。

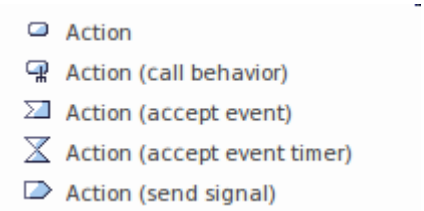
Enterprise Architect提供了一个丰富的工具箱来处理这些关系，不仅包括将系统行为以活动和行动的形式分配给块的能力，而且还能够将这些元素与块拥有的行为特征相关联，例如操作。

行动基本行为建造积木

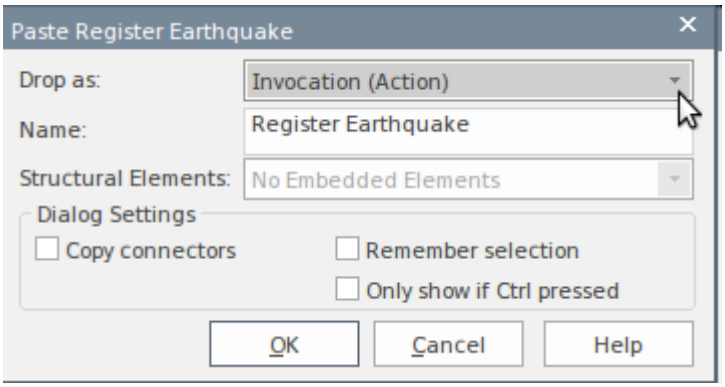
行动是连接在一起以描述活动、子系统、系统或其部分之一的行为的行为原子。实际上，一个活动由一组行动组成，这些行动共同将输入到活动中的项目（令牌）转换为活动输出的项目（令牌）。一个行动中的第一个动作将从拥有的活动的一个输入参数节点接收其输入，而该序列中的最后一个序列行动输出放置到该活动的一个输出参数节点上。行动本身具有称为销的输入和输出设备——行动将在其输入销上接收代币，执行其工作并将生成的代币放置在其输出销上。



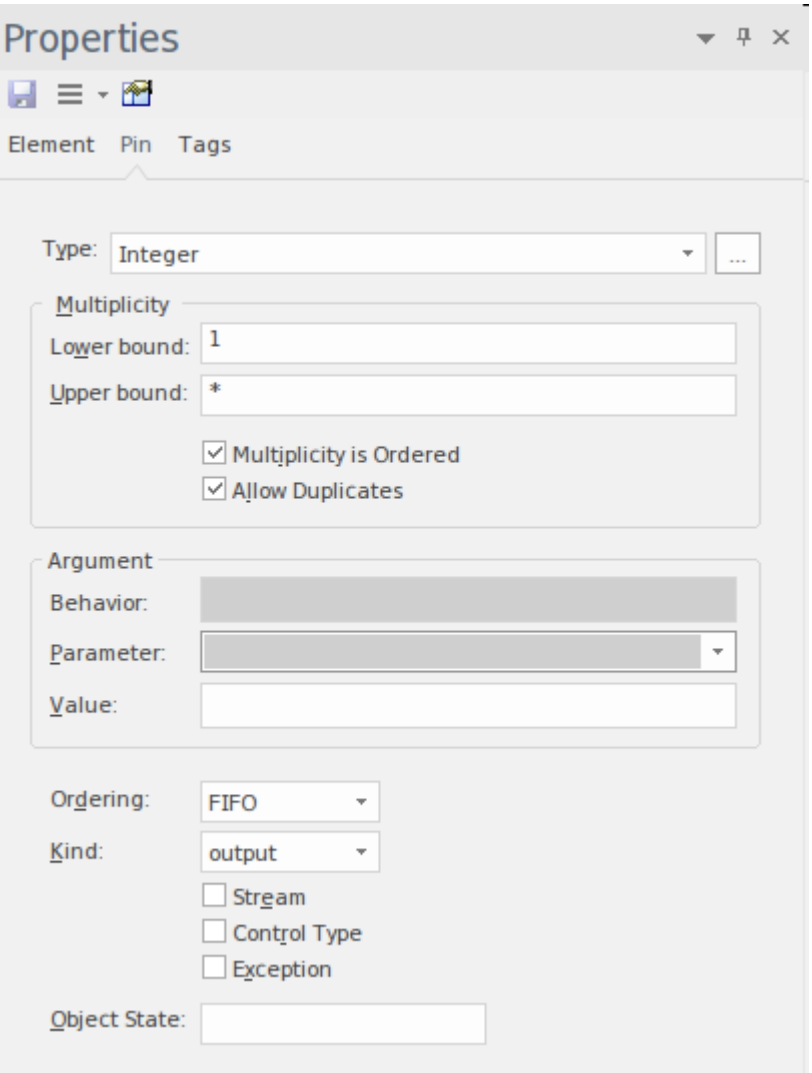
Enterprise Architect对建模行动及其输入和输出具有丰富的支持，并且可以在使用这些基本的行为构建块时使用用户界面的各个部分。工具箱中提供了许多不同类型的行动。



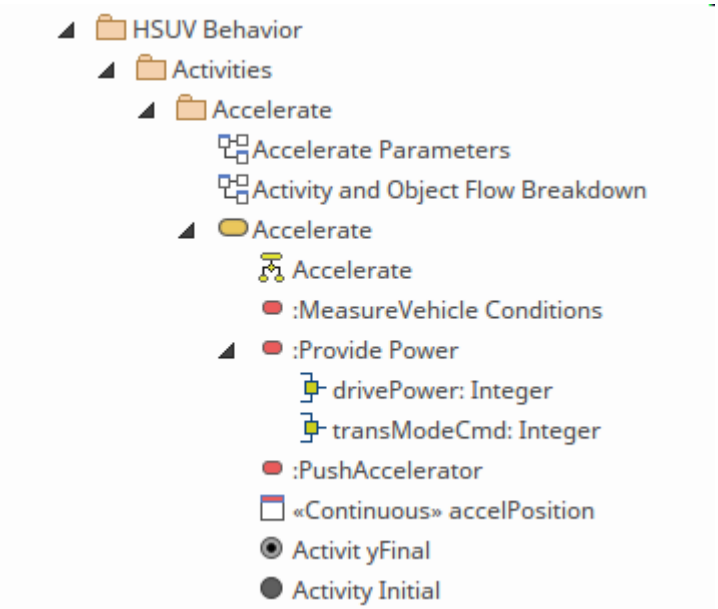
工程师可以直接从工具箱添加行动，但更常见的是从已在层次结构中定义的现有活动创建行动，如主题创建活动层次结构中所述。要做到这一点，一个活动将不是从工具箱而是从浏览器窗口中拖放到一个打开的活动图表中作为调用 - 这具有基于该活动创建一个行动并将其放置在图表中的效果。



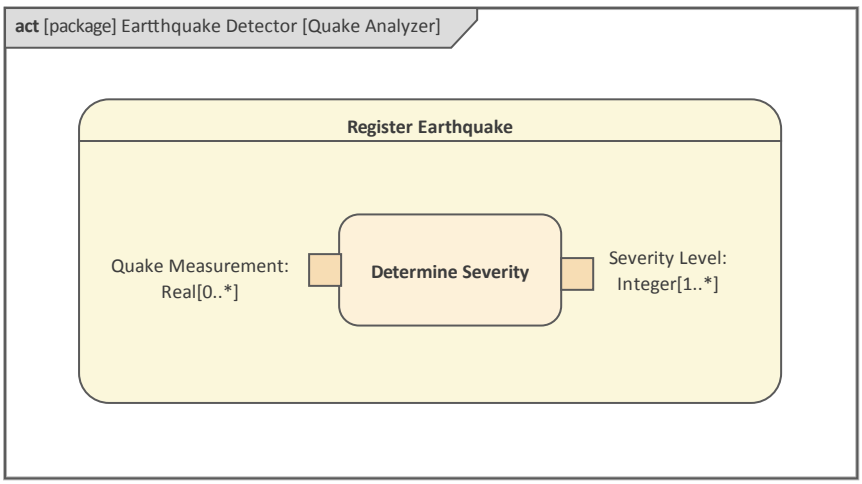
集成的属性窗口使行动和他们的销售变得销。工程师可以指定行动及其属性的范围，包括销名称、销类型、多重性、方向等等。属性窗口可以停靠或浮动，甚至可以拖动到不同的监视器上；当在浏览器窗口或图表中选择元素时，可以查看、创建或更改属性。



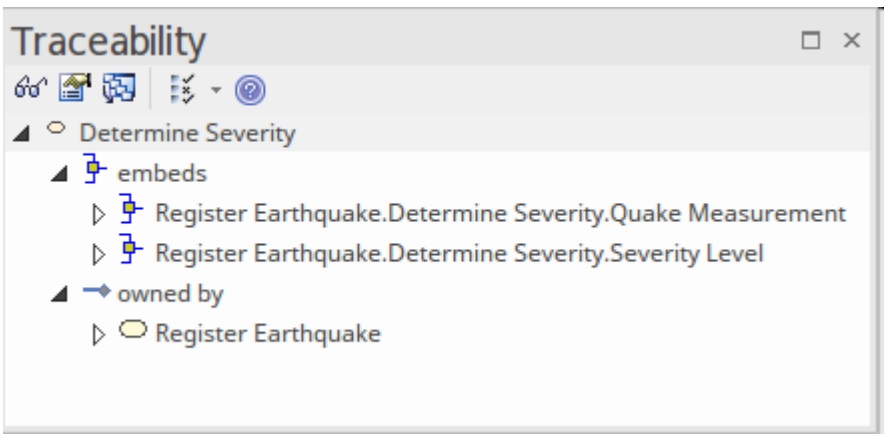
创建图表时，添加的元素会自动插入浏览器窗口，让工程师可以查看活动、行动、销和其他元素的结构视图。即使图表未打开，也可以在浏览器窗口中选择元素并直接在属性窗口中进行编辑。



A一些属性将显示在销标签的图表中，包括名称、类型和多重性。多重性指定了下限和上限。对于给定的操作执行，下限指定引脚可以使用或创建的令牌的允许数量，上限指定在该引脚上使用或创建的最大令牌数。



还有很多windows可以在行动性窗口中发挥作用，**包括可可追溯性窗口**，它显示了无论它们位于存储库中的什么**位置**，它们是如何相关的；它还显示了它们的结构特征，例如销和参数。



介绍活动图表

活动图是可用于显示序列块或其他结构元素行为的行动顺序的图。行动使用控制流进行排序，并且可以包含输入和输出销，它们充当从一个行动流向另一个（或从控件或缓冲区节点）流动的项目的缓冲区。行动所进行的工作要么消耗要么生产这些物品。根据所描述的系统和活动，这些项目可以是材料、能源或信息。

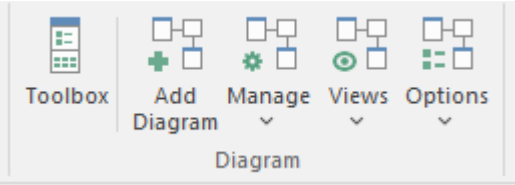
创建活动图表

可以从用户接口中的多个位置创建活动图，**例如**：

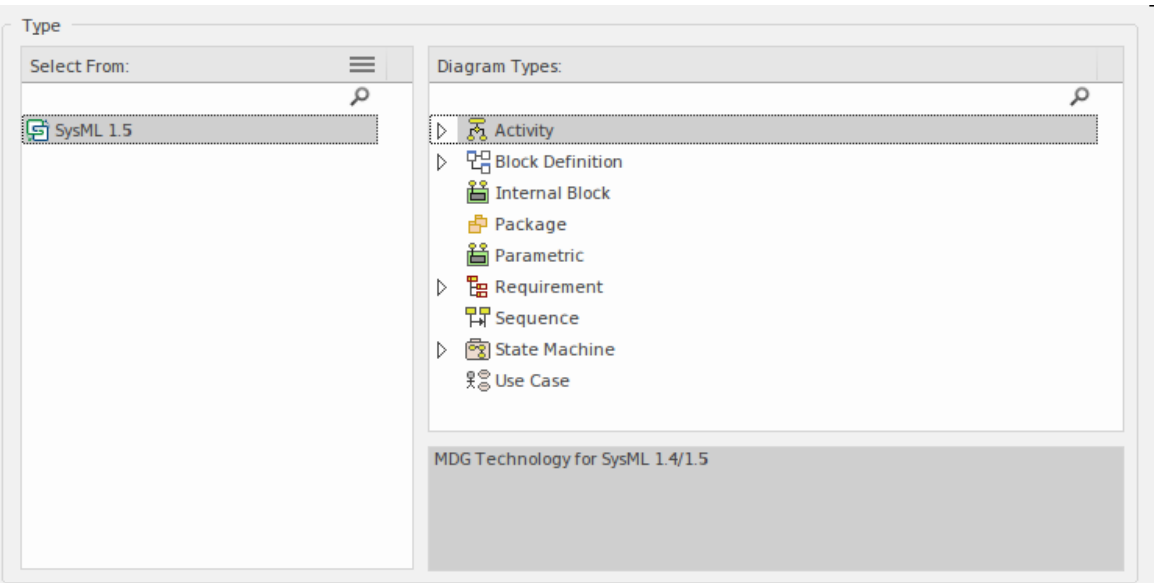
- 设计功能区- 图表" 面板上的 添加" 图表图标
- 浏览器工具栏 -新图表打开
- 浏览器上下文菜单-图表图形

我们将使用设计功能区创建一个活动图。首先，在浏览器中选择要创建活动图的位置。与所有图表一样，这可**以是包或元素**，但通常将活动图表插入包中。在浏览器中选择包位置后，选择功能区选项：

设计>图表> 添加图表



选择此选项将打开*New*图表对话框，允许您更改图表的名称（默认为包含图表的包或元素的名称，如在浏览器中选择的那样）。选择 SysML 透视图并选择 SysML **版本后**，将显示一个图表列表，您可以从中选择活动图表。当您单击确定按钮时，将在浏览器中选择的位置创建一个新的活动图。将打开图表画布，允许您开始添加元素和连接器，以描述系统将为其用户提供的价值。 Enterprise Architect还将显示工具箱的 活动"页面，**其中包含** SysML 规范定义的元素和关系，以适用于构建活动图。除了始终可用的公共元素和公共关系工具箱页面外，还可以根据需要打开任意数量的其他工具箱页面。



与活动图一起使用的最重要的元素和连接器是：

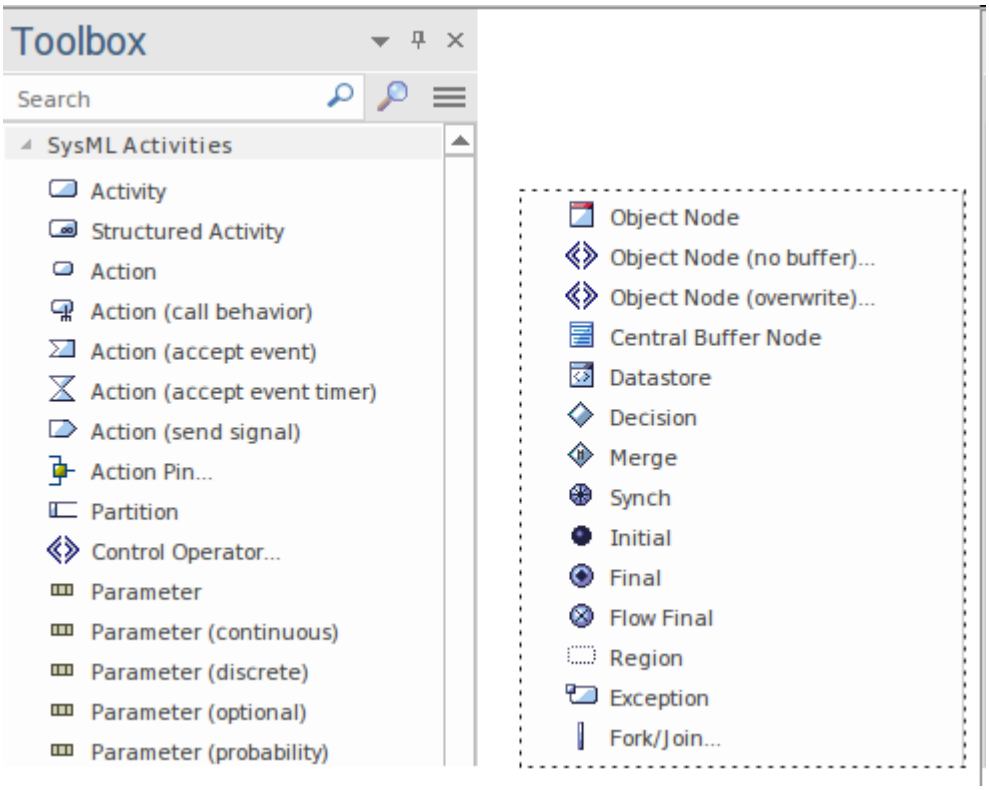
元素

- 活动
- 行动
- 行动销
- 分区
- 参数
- 最初的
- 终点
- 决策
- 分叉和汇合
- 数据存储

连接器

- 控件
- 物件流
- 中断流

在这些图表上可以使用许多其他元素和连接器，这对于更高级的建模很重要；随着建模者变得更有经验，或者正在描述或设计系统行为的更复杂部分，可能需要其中一些。其中包括活动参数、Merge、Central Buffer节点、区域、分叉与汇合、决策and Merge。



可以通过将元素从工具箱到图表画布上来将元素添加到图表中。以初始和一个或多个终点活动开始元素被认为是一种很好的做法，这些元素被命名以描述其可能完成的方式和可能的多种方式。将这些元素的名称留空或给它们一个陈旧的名称，例如 开始"或 结束"，将无助于让读者清楚地了解正在建模的系统或系统的一部分，并且可能导致误解图的。当这些节点被添加并适当地放置在图表中时，可以将行动和物件节点添加到图表中。可以使用控件流关系连接行动，定义序列将执行的行动。

一旦创建了基本图表，并且随着对领域和系统行为的进一步积累，可以使用其他元素和关系来构建或修饰图表，包括：

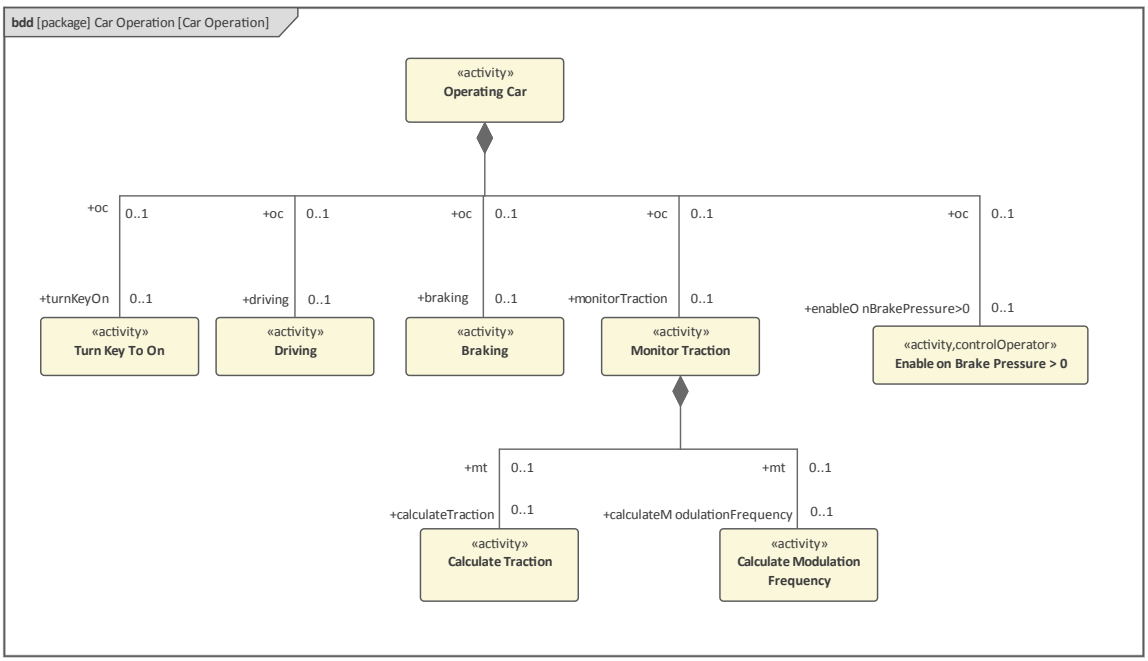
- 控件流控制结构：物件和汇合、分叉和决策节点

- 活动输入和输出结构：活动参数（流式和非流式）
- 分组行动集：可中断区域和边缘
- 储存存储结构：数据存储和中央缓冲区

如前所述，活动图有一组丰富的语言设备，鼓励工程师使用这些设备使系统描述更丰富，但需要注意确保目标受众能够理解这些语言机制.

创建活动层次结构

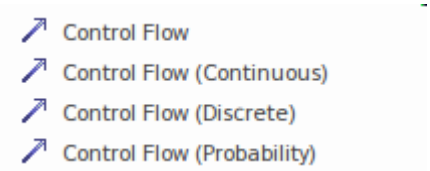
系统建模语言和Enterprise Architect的新手可能会惊讶地发现，活动图上使用的不是活动，而是行动。作为行动分类的活动通常在块定义图上可视化。这似乎有点违反直觉，但是当您了解行动是系统行为的基本原子时，它就更有意义了。活动是分类器，因此可以像块元素一样参与范围广泛的结构关系，这就是为什么可以在活动之间使用标记为组合的关联等关系。



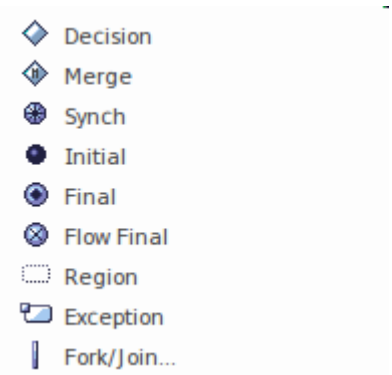
在该图中，使用了分解结构，其中使用组合关系将活动分解为多个更细化的活动。使用时间上下文菜单中可用的灵活线条样式之一，将许多关系组合在一起以使图表更具吸引力。

使用控件流指定行动序列

行动在活动的上下文范围内，行动执行的顺序行动是通过使用称为控件流的特殊连接器来控制的。这些连接器是在行动之间绘制的有向线，基本上充当控制令牌的管道 - 允许令牌沿箭头方向从一个行动流向下一个行动。一个行动可以开始它的工作，直到所有传入的控件流都收到一个令牌；一旦他们执行了，并且表示将放置在行动控件流中的标记，这意味着它将移动到下一个动作。控件关系可从 活动序列行动"工具箱图表

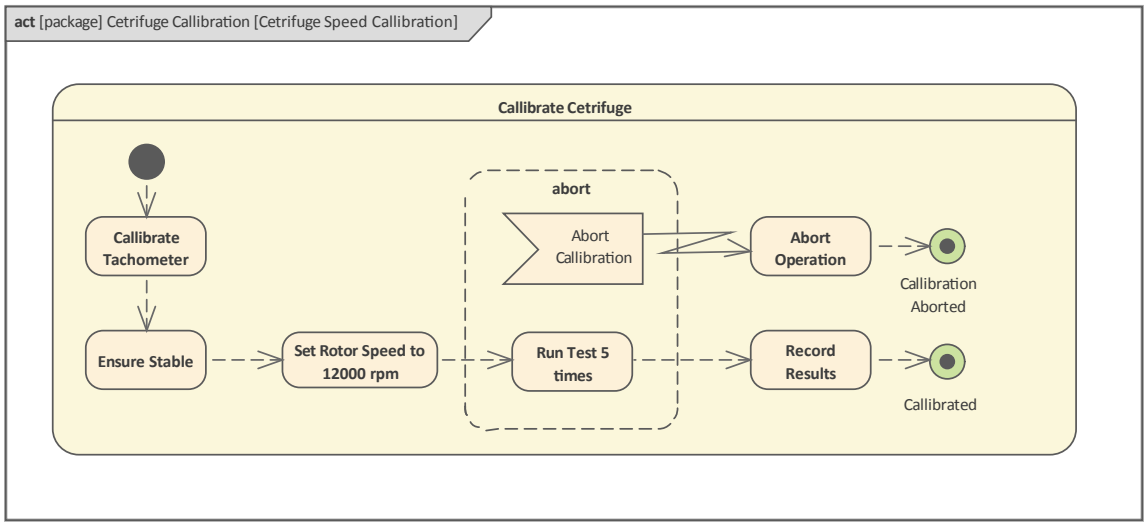


还有一些控件可以和控件一起使用控件使用分叉、汇合、Decision 和 Merges 来编排Flows 的工作方式有：初始、终点和流终点三个专门的节点，分别作为流的起点和终点。终点（形式上是活动终点）是用来表示当一个节点到达时，一个令牌到达，而活动流终点将消耗整个令牌，对封闭的活动没有影响。



中断正常流程

在活动的执行过程中有许多情况，建模者可能想要指定一种方法来停止活动的一部分中的行为。例如，在现实世界的场景中，用户可能会在使用机器功能（例如校准函数）时部分完成，然后决定他们想要结束校准过程的特定部分。这种情况可能由界面上的取消按钮提供。 SysML 允许使用可中断区域和中断边缘对这种情况进行建模。该符号允许绘制可中断区域以包含许多元素，例如行动和其他节点。通常，当发生不寻常的事情时，会在活动中触发事件并由 Accept信号行动接收；这个元素没有传入的控制流和一个传出的中断边缘，它的目标是行动在区域。

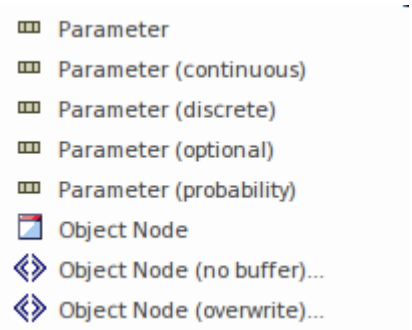


在此图中，工程师对用于执行离心机速度校准的过程进行了建模。离心机校准过程可能因各种原因而中断；例如，如果离心机变得不稳定或操作员被叫去执行其他职责。接受事件行动用于表明该活动具有一种机制来侦听该活动的指定区域内所需的中断。然后，特殊的中断流连接器针对区域外的行动，用于关闭离心机；最后这流向校准活动被终止。

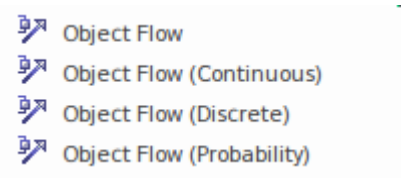
使用物件流指定项目流

活动及其组成的行动通常通过处理到达输入节点的项目来完成工作，并且当工作完成时，将结果项目放置到输出节点上。如前所述，SysML 中的活动状态建模基于称为 Petri 网的数学分支，它与状态事件系统有关。到达输入结构的项目必须以有序和系统的方式通过活动图及其包含的行动。通道由物件流创建，这些流充当将令牌从一个节点传送到另一个节点的管道。代币代表了许多不同类型的 事物”，包括信息、结构或物理项目，如固体、液体和气体。因此，项目通过活动的方式有两个重要部分 - 充当令牌起点和目的地的节点，以及传输项目的连接器（导管）。

Enterprise Architect完全支持对这些流程进行建模，当创建或打开图表进行编辑时，工具箱包含如下所示的物件节点：

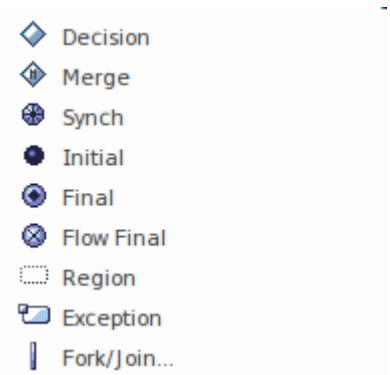


它还包含一个部分，列出可用于连接节点的物件流关系，为令牌从一个节点流向另一个节点创建管道。



协调令牌流

在对复杂系统进行建模时，通常需要为令牌流创建更精细的路径（管道），例如分叉和连接路径以允许将令牌发送到多个object节点，以便可以同时完成工作，或者允许令牌根据某些条件沿着特定路径路由。这些控件节点控制流程，并在工具图表箱的页面上图表工具箱。



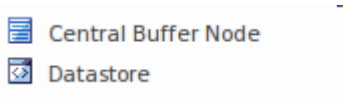
Enterprise Architect允许操作连接器以创建所需的任何路径。这可以通过使用连接器的上下文菜单中的线条样式来完成；其中最灵活的是自定义线样式，但还有其他几种非常有用的样式。建模者还可以A连接器末端固定到

源或目标元素的特定部分。

储存用于运输中的代币

在活动执行期间，有时需要将令牌存储的时间比活动参数和行动销可能的时间更长，它们只是充当临时存储设备。 A常见的情况是，当许多行动需要访问令牌流时 - 令牌可以存储在中央缓冲区中，并可供需要它们的节点使用。中央缓冲区接受其传入流上的所有令牌，然后将令牌提供给下游节点；一旦被接受，令牌就会从缓冲区中删除。

可以通过将工具箱中的 "中央缓冲区"图标拖到打开的活动图表上来创建中央缓冲区；然后可以使用物件流将其连接到其他object节点。

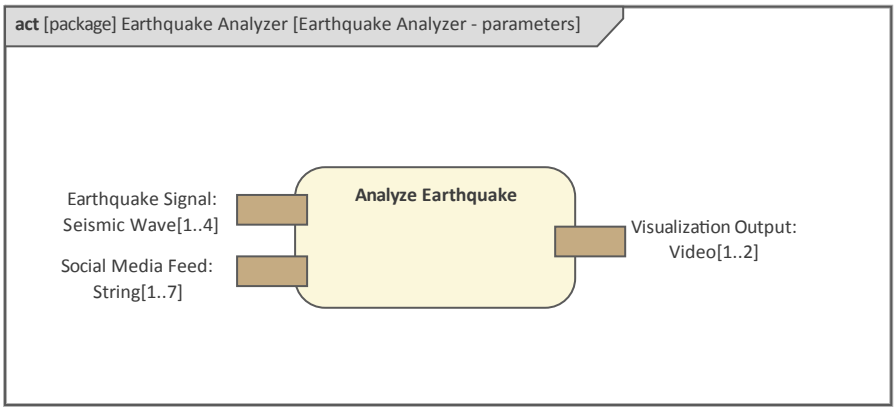


因此，在活动执行期间，中央缓冲区可以充满令牌或清空，具体取决于令牌的消耗。另一种类型的节点是数据存储，它是中央缓冲区的一种特殊化，其中，当下游操作使用令牌时，会制作一个副本并将其存储回缓冲区中。这具有数据存储具有永久存储外观的效果 - 但仅限于活动执行的生命周期。

可以通过将工具箱中的 "数据存储"图标拖到打开的活动图表上来创建数据存储；然后可以使用物件流将其连接到其他object节点。

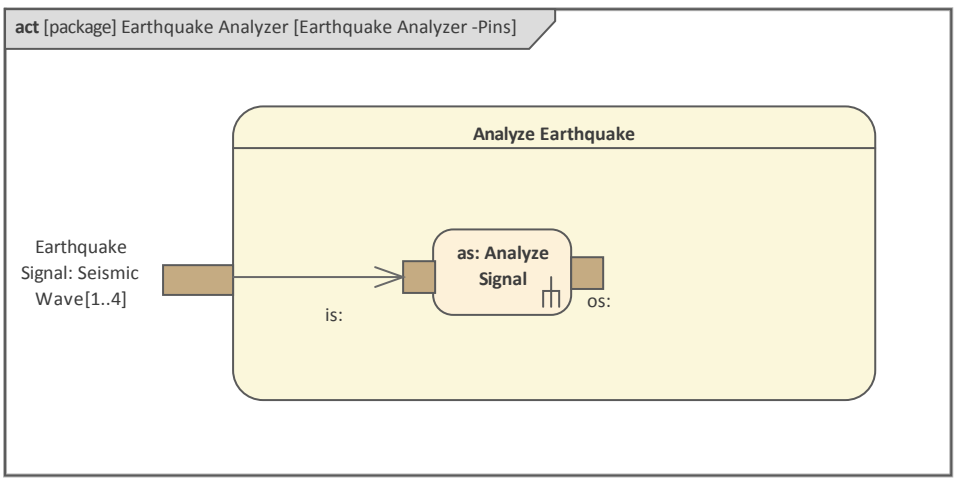
使用参数和销对输入和输出建模

活动及其组成行动是系统的劳动力；块和部件等结构元素定义了系统的结构或解剖结构，而活动定义了生理学。当一个活动正在执行时，我们看到结构元素被调用以完成某种类型的系统行为。系统所做的大部分工作以及定义工作内容的行为都取决于执行活动为产生输出而消耗的系统输入。



系统之间的输入和输出差异很大，可能包括控制信号、材料、光、流体、能量、数字和信息等。输入和输出称为参数，可以有类型并且可以有多重性。键入确保活动指定它所期望的 事物"的种类（类型）。 **因此，如果一**个输入参数定义了一种液体类型，或者更具体地说是一种液体污染物，那么如果它接收到一个气体或一个整数值作为该参数的输入，则该活动将是不正确的。类型可以是从简单整数到复合结构的定义集合中的任何一种。输入和输出可以由一个块输入，这样你就有一个定义明确的结构元素——例如，在超市结账时通过自我扫描系统的杂货项目。可以为参数定义一系列其他属性，包括流式传输或非流式传输、多重性和方向。当参数中存在连续流动时使用流式传输，**例如流体**，或通信或信息信号，例如音频或视频流。多重性定义了输入参数消耗或输出参数产生的令牌数量的上限和下限。而方向定义参数是接收输入 (in) 还是产生输出 (out) 或两者的组合 (inout)。

当活动作为调用放置在活动图上时，它们由行动表示，并且活动拥有的任何参数都将被建模为这些行动销进入物件流上的销接收令牌和拥有行动执行其工作并将任何指定数量的令牌放置在输出销上。销可以有一个简单的类型，比如整数，也可以有一个复杂的结构，比如一个块，甚至是一个视频流。多重性指定了一个下限和上限，这些下限和上限定义了允许到达和离开任何给定销售的销的最小和最大数量。这个未完成的图表显示了一个具有输入和输出的行动销，以及令牌从 Owning活动的输入参数沿物件流的传输。



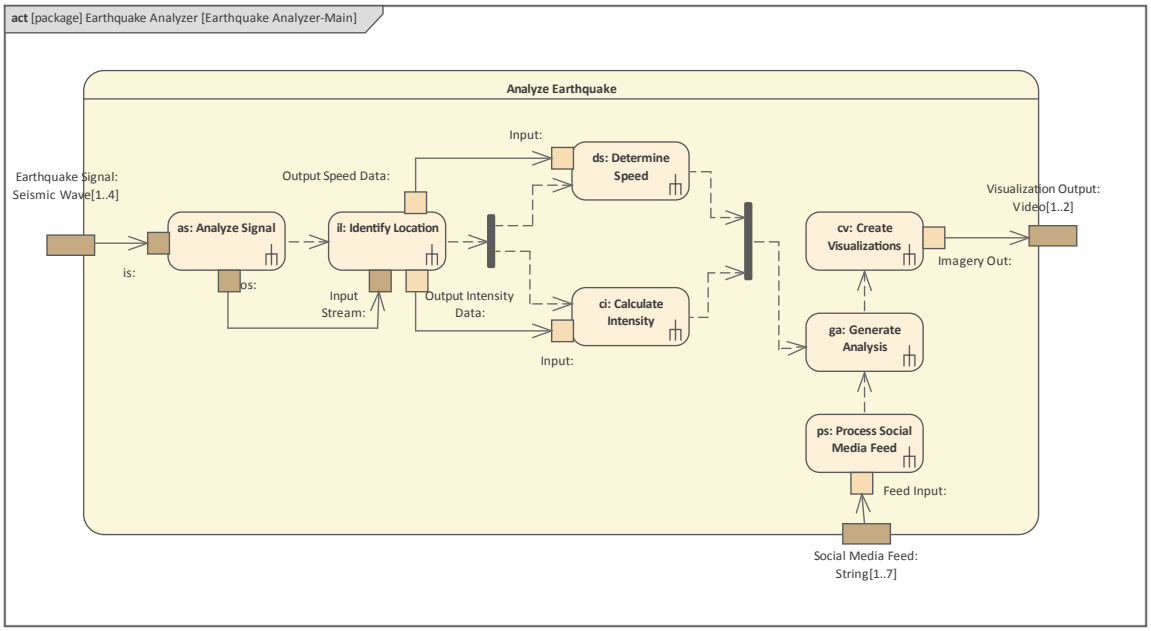
这些参数销为元素交互点，即它们与其所处环境发生作用的位置；可以使用多功能特征窗口选择它们以包含在图表中。

Features

Attributes Operations Receptions Parts / Properties Interaction Points

Name	Element	Type	Stereotype	Visible	Owner
Social Media Feed	ActivityParameter			True	Analyze Earthquake
Earthquake Signal	ActivityParameter			True	Analyze Earthquake
Visualization Output	ActivityParameter			True	Analyze Earthquake

Enterprise Architect允许您创建一个图表，将拥有活动显示为图表中包含的其他活动的容器作为行动。在此图中，在拥有的活动上定义的活动参数表示为活动销上的行动，这些活动已被包括为活动的调用。该图显示了一个具有两个输入参数和一个输出参数的活动。令牌形式的输入可以在到达销时通过图表进行跟踪。一旦行动完成其工作，代币就会被放置在输出销上。控件显示了所附行动的顺序。注意，a分叉和汇合用来表示两个行动可以并行进行。另请注意，许多销已被定义为流，在图表上以销的销售表示。



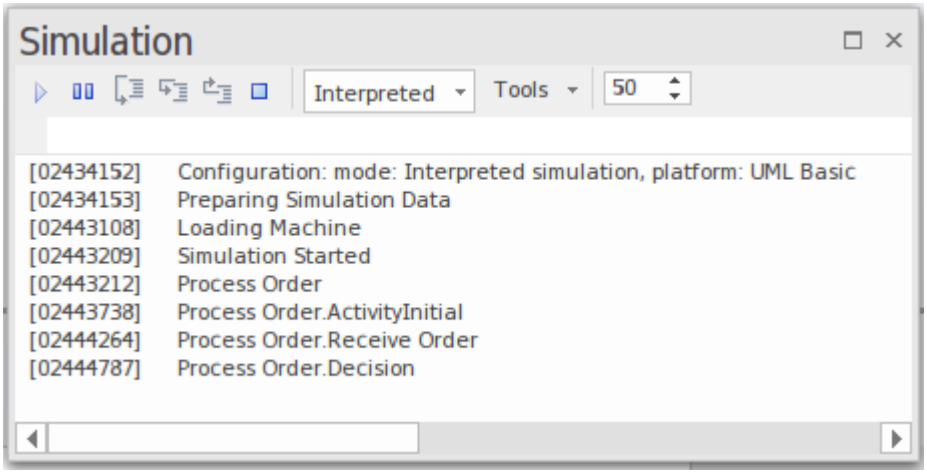
使用模拟可视化活动

您的模型中的任何 SysML活动图都可以使用内置的动态模型模拟器进行模拟。这提供了一种引人注目的图表可视化方式，并且对于与用户和工程社区中的其他人一起运行演示或演练很有用。

使用模型模拟器，您可以模拟包含行为的概念模型设计的执行。启动仿真时，对当前模型包进行分析，并生成动态仿真过程来执行模型。由于模拟器直接分析和使用UML结构，因此不需要生成中间代码或编译模拟 可执行文件”。这导致了一个非常快速和动态的模拟环境，在该环境中可以非常快速地进行更改和测试。

仿真窗

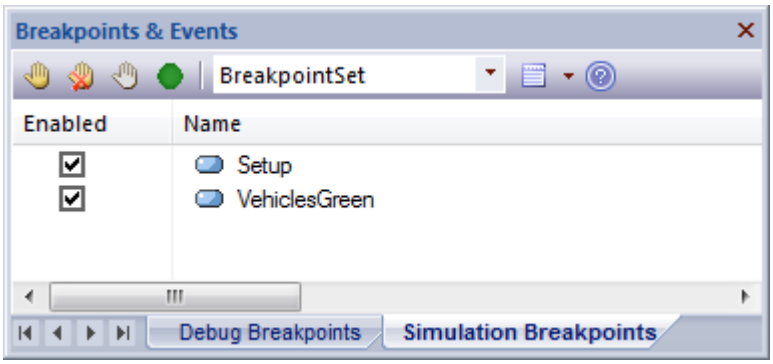
仿真窗口提供启动、停止和步进仿真的主界面。在执行期间，它显示与当前执行步骤和其他重要信息相关的输出。有关工具栏命令的更多信息，请参阅Run Model Simulation帮助主题。



注记工具栏正下方的文本输入框。这是控制台输入区域 - 在这里您可以输入简单的JavaScript命令，例如：*this.count = 4;*将名为 “count”的仿真变量动态更改为 4。通过这种方式，您可以在运行时动态影响仿真。

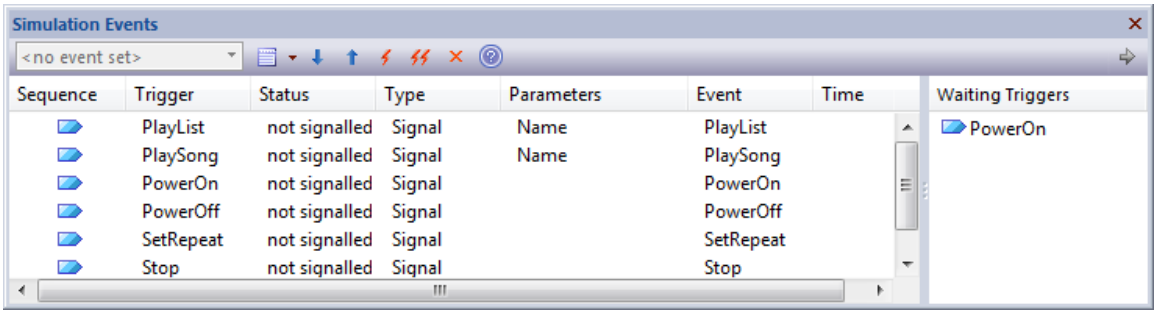
断点和事件窗口

仿真过程还使用了断点和标记窗口的 “仿真断点”选项卡（ “仿真>动态仿真仿真>断点”）。在这里，您可以在仿真中的特定元素和消息上设置执行断点。看Simulation Breakpoints 帮助 主题以获取更多详细信息。



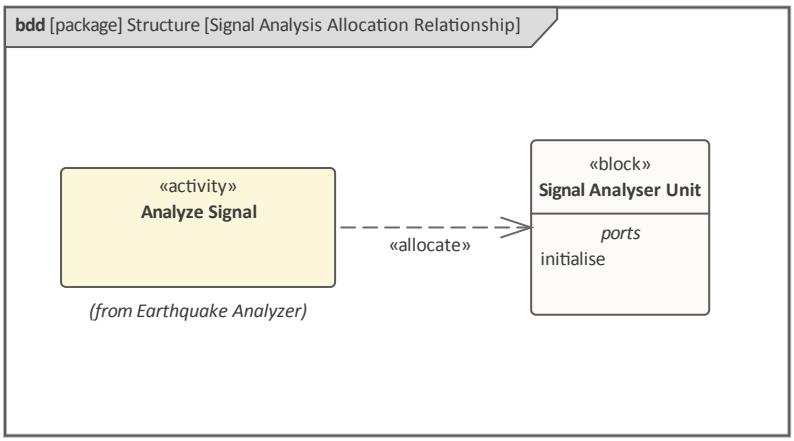
仿真事件窗口

仿真事件窗口（ “仿真>动态仿真>事件”）提供了管理和执行触发器的工具。触发器状态机用于控制过渡的执行。

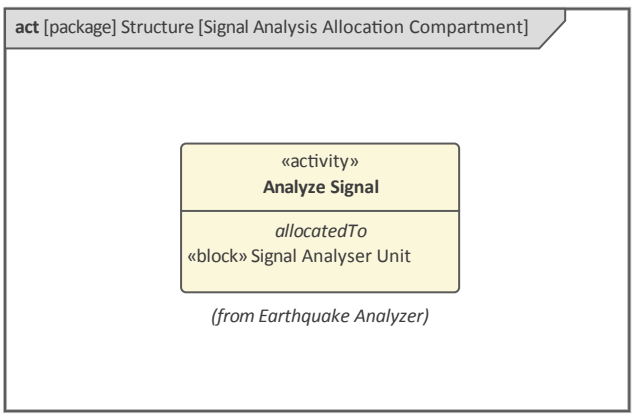

Simulation Events
<no event set>
Sequence
Trigger
Status
Type
Parameters
Event
Time
Waiting Triggers
PlayList not signalled Signal Name PlayList
PlaySong not signalled Signal Name PlaySong
PowerOn not signalled Signal PowerOn
PowerOff not signalled Signal PowerOff
SetRepeat not signalled Signal SetRepeat
Stop not signalled Signal Stop
PowerOn

分配和其他关系

系统工程的A基本方面或学科是将结构元素与行为元素联系起来。随着模型的发展，模型的这两个重要方面将相互交织，但是当许多工程师开始定义系统时，通常很难准确地说明行为和结构将如何相关。分配关系在这些情况下特别有用。它可以用作显示行为元素和结构元素之间关系的一种方式，这将为模型中描述的概念变得更加确定时将采用的更严格的建模提供信息。



Enterprise Architect还支持许多其他表示分配关系的方式，包括作为行为或结构元素中的隔间。

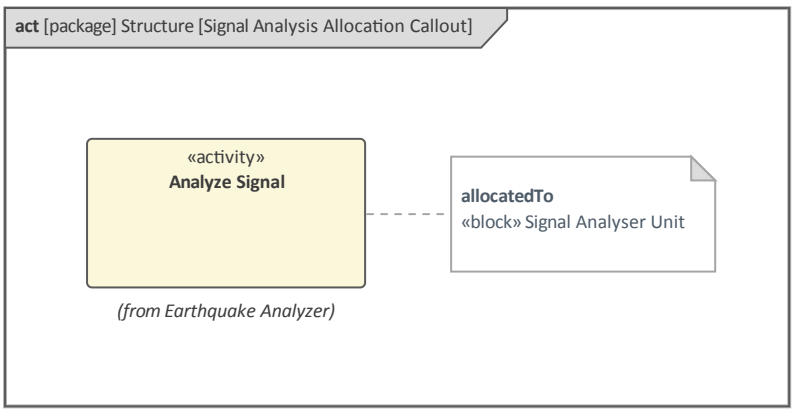


任何允许隔间的元素都可以配置为隐藏或显示任意数量的可用隔间。隔间列表特定于元素并且是动态的，这意味着只有当元素参与指定类型的一个或多个关系并且关系在包含图上不可见时，隔间才会可见。相同的选项可用于显示元素的一系列其他结构或语义方面，如此处所示。

也可以在标注符号中显示关系，其中注记连接到元素并显示关系的名称和相关元素的详细信息。此图显示了一个活动的符号，显示了它已分配到的块。为此，工程师必须：

1. 确保关系显示在图表中。
2. 选择关系并显示上下文菜单。
3. 选择 创建链接注记”选项。

此标注符号可用于任何类型的 SysML元素或关系，并且是为某些类型的受众显示关系的有用方式。

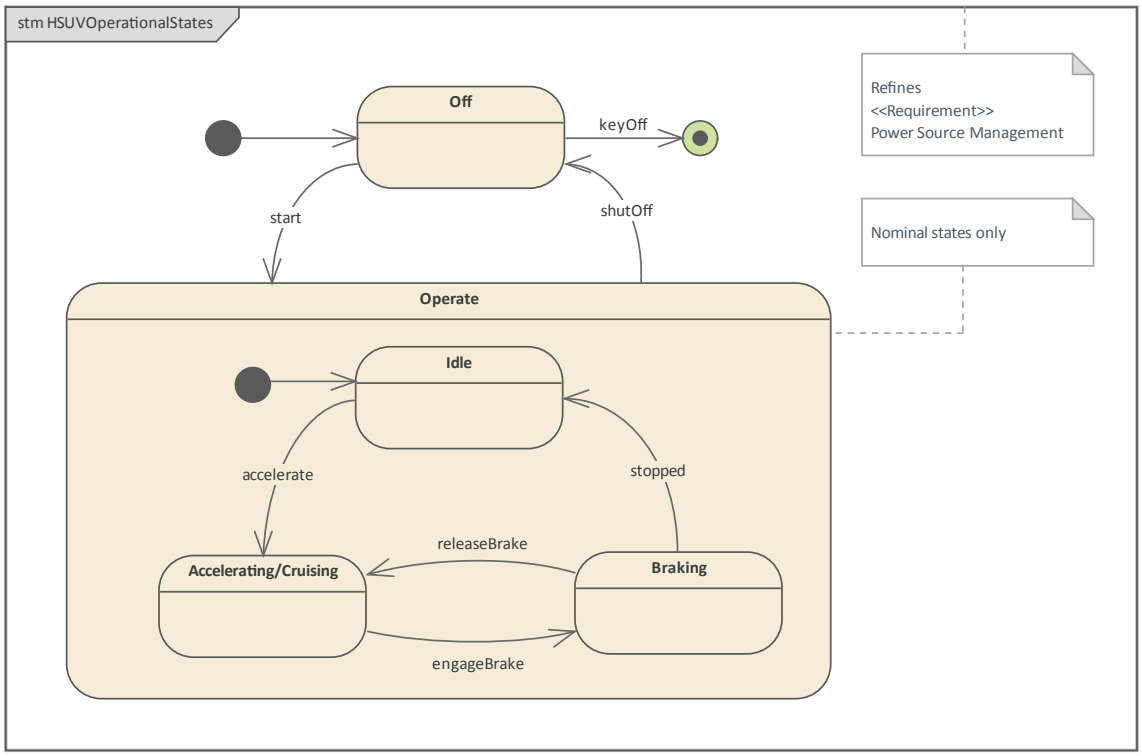


用状态机建模更改

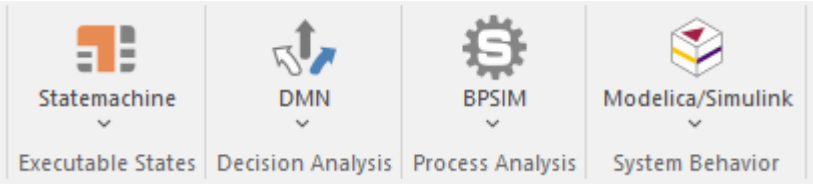
随着 事物"的变化或演变，我们的世界不断变化，从一种状态转移到另一种状态。水结冰，冰川变形和流动，冰融化，交通信号灯在绿色、琥珀色和红色之间循环，飞机起飞、爬升、巡航、下降和着陆。 SysML状态机用于描述结构如何以 Blocks 的形式在时间盒的生命周期中改变其状态。我们关心的不是块实例的结构，而是它的行为，这反过来会影响它的结构。我们不是对 事物"可能处于的每一个状态感兴趣，而是对重要状态感兴趣。因此，水分子的重要状态可能是固体、液体或气体，但我们通常对 67 摄氏度的液态水不感兴趣。如果我们正在查看object生命周期的电影卷轴，状态机将挑选出发生重要和相关变化的重要帧。

决定什么是相关的是建模工程师的特权和特权，同一个块可以有相同或不同工程师定义的任意数量的状态机。可以从乘客登机和下机的角度、维修计划的角度、升降机的角度或任何数量的其他角度对飞机的状态进行建模。

该状态机图描述了 SUV 机动车辆的运行状态。复合州，嵌套在其他州内使用哪个州。共有三个高级状态 - Off、Operate 和未命名的结束状态。运行状态有多个子状态，即怠速、加速/巡航和制动。连同转换一起，这描述了车辆在启动、加速、制动、停止以及最后点火关闭时的状态。



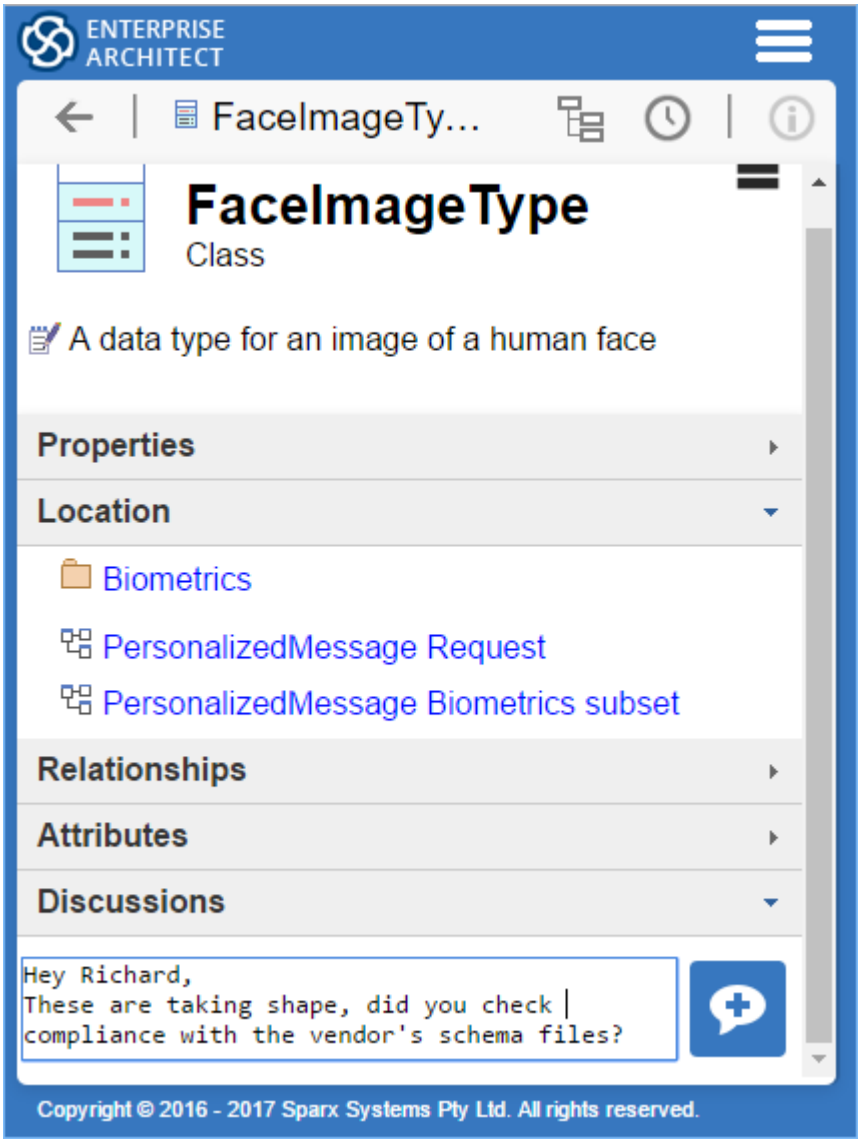
使用Enterprise Architect ，工程师可以创建状态机并定义从一个状态到另一个状态的转换，包括触发状态变化的事件和行动的动作。除了这些标准建模表示之外，该工具还具有一系列特征，可以帮助可视化和推理这种将结构和行为联系在一起的重要语言机制。我们将在本主题中看到的这些功能之一是可执行状态机功能，可从仿真区获得。



状态机可以在任何粒度级别定义，因为它们是块行为的表达。许多 SysML 新手对这一点感到困惑。因为块可以表示非常简单的东西——比如潜艇控制面板上的开关——或者像潜艇本身这样复杂的东西，所以状态机也可以同时表示开关和潜艇的状态。两个状态机模型可能具有相同的复杂性，即使被建模的事物本身在复杂性方面显然处于光谱的两端。

对于没有经验的建模者来说，状态机图可能看起来非常简单，但它们是描述和分析无法通过其他方式解决的复杂问题的高效工具。它采取了不同的思维方式和方法，问题的核心往往集中在块的层次、上下文和状态机的视

角的选择上，而不是图的细节。通常，最好的结果是由许多工程师一起工作以启发式方式实现的。这可以使用Enterprise Architect的协作特征来完成，允许分散在地理上的工程师通过桌面客户端的邮件、讨论、聊天和正式评论，或在智能手机、平板电脑或笔记本电脑上的浏览器中在模型中进行通信.



状态机起源于离散的事件驱动行为，使用基于 David Harel 的 StateCharts 形式主义的面向对象变体的有限状态机。

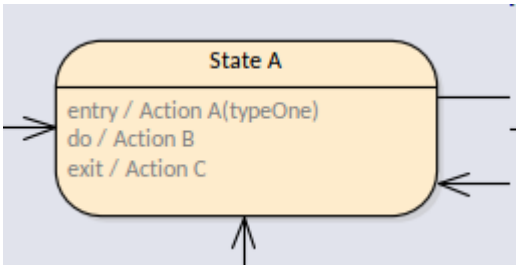
状态和行为

状态是在上下文A时间状态机内创建的，用于模型用于工程师定义的拥有块的重要条件。重要的是要记住，状态机是从特定的角度描述块的生命周期，并且必须从这个角度定义状态 - 不是所有状态，而是重要的状态，例如 On/Off、Open/Closed ，绿色/琥珀色/红色或冰/水/蒸汽。形式上，状态模型化了状态机行为执行中的情况，其中某些不变条件在特定持续时间内保持不变。

A块通常会在给定的状态下花费一些时间，这可能会持续纳秒或几天，具体取决于上下文；这个时间方面通常不会在模型中编码，但可以在模拟中设置。对于任何给定的状态，可以定义三种行为（称为行动）：

- *Entry* - 输入状态时触发
- *Do* - 在进入行为之后和退出行为之前触发
- *Exit* - 在状态退出之前触发

此图显示了这些状态如何在状态机图中表示。 Enterprise Architect可以在单个元素的级别有条件地显示这些和其他隔间，或者为图表上的所有元素集中显示。



需要注记的是，终点节点在形式上也是一种状态，但它不具有与圆角矩形图上表示的状态相同的行为语义。

状态分为三种基本类型，每一种对于建模不同类别的问题都很重要：

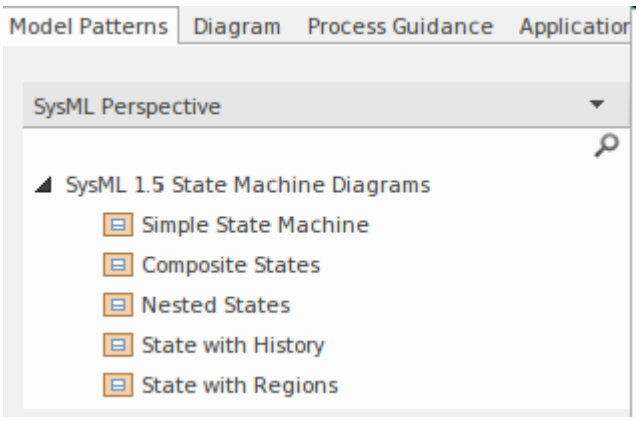
- 简单状态——不包含内部状态
- 复合状态- 包含至少一个拥有状态的区域
- 子机状态——表示嵌套在所属状态中的整个状态机

Enterprise Architect允许您对模型状态类型中的每一种进行建模，建模者可以在状态机图中高效地使用它们来表达现实世界的工程问题和解决方案。

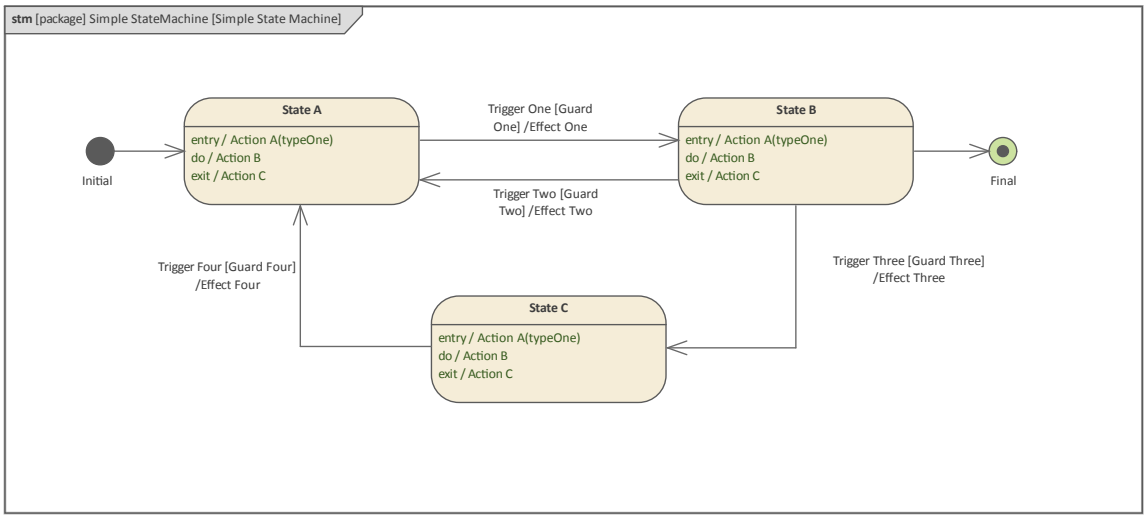
状态机图表

状态机图是 SysML 九种核心图类型之一；它用于在块更改状态时创建和可视化块的行为。图上的关键元素是状态、转换和伪状态。从特定的角度来看，状态代表块生命周期中的重要时间，转换代表从一个状态到另一个状态的移动，而伪状态，正如我们将看到的，充当影响转换工作方式的交通控制器。

Enterprise Architect帮助建模者创建任意数量的状态机图表，每个图表可以有任意数量的状态、转换和伪状态。这些图表元素和连接器中的每一个都可以依次添加其他信息，这些信息将用更多细节修饰图表。该应用程序有一个模式库生产力工具，对新手非常有用，同样欢迎有经验的建模者。此屏幕截图显示了可用于创建状态机图的模型模式列表。



该模式可用于创建多个不同的状态机；在这个例子中，我们创建了一个简单的（单个区域）图表，其中包含所有适当的细节添加到状态和转换。建模者可以A存储库中的适当位置创建此图表，然后编辑状态和转换以及图表以使其适合自己的建模上下文。从该模式创建的初始状态机图将如下所示：



这个方便的特征提示工程师可以完成详细信息，例如简单的触发器和守卫条件转移

，或状态上的进入和退出操作，他们可能没有意识到 - 结果是符合最佳工程实践的图表，为客户产生更好的结果。

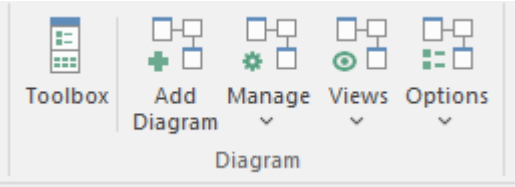
创建状态机图表

可以A以下命令从用户接口中的多个位置创建状态机图：

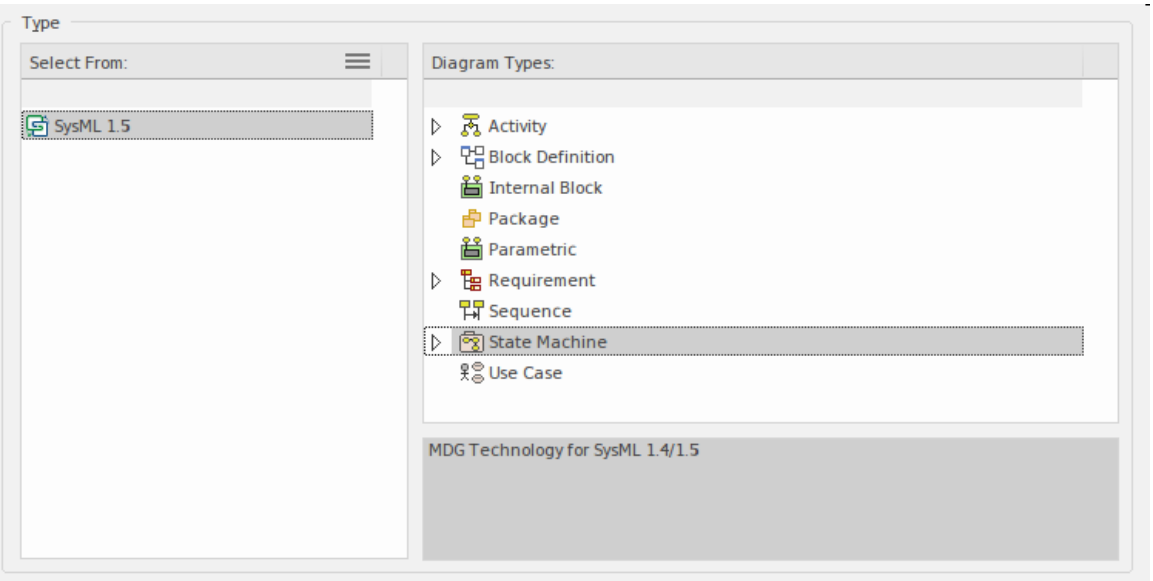
- 设计功能区 - 图表”面板上的 添加图表”图标
- 浏览器窗口工具栏 - “新图表”图标
- 浏览器窗口上下文菜单- '添加图表'选项

我们将使用设计功能区创建状态机图。首先，您在浏览器窗口中选择您希望状态机图表所在的位置。与所有图一样，这可以是包或元素，但通常将状态机图插入到元素（例如块）中以描述块生命周期中的重要阶段。在浏览器窗口中选择位置后，然后选择：

功能区：设计>图表> 添加图表



选择此选项将打开*New*图表对话框，允许您命名图表；该名称将默认为包含图表的包或元素的名称，但您可以更改它。选择 SysML 透视图并选择 SysML 版本后，将显示图表列表，您可以从中选择状态机图表。选择确定按钮后，将在浏览器中选择的位置创建一个新的状态机图。图表画布将打开，允许您开始添加描述主题生命周期中重要阶段的元素和连接器。 Enterprise Architect还将显示工具箱的状态机页面，其中包含图表规范定义的元素和关系，以适用于构建状态机图表。除了始终可用的公共（元素）和公共关系页面之外，还可以根据需要打开任意数量的其他工具箱页面。



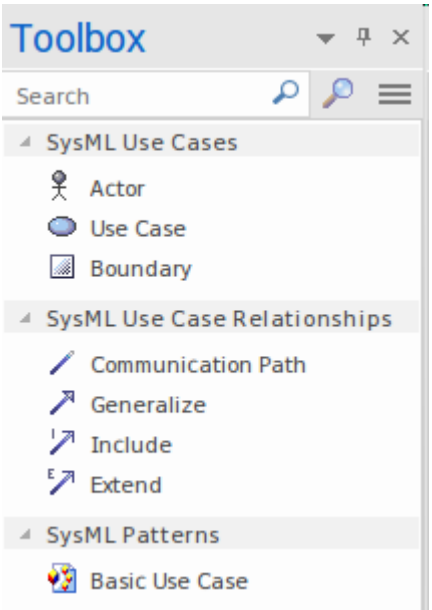
与状态机图一起使用的最重要的元素和连接器是：

元素

- -状态- 定义实体生命周期中的重要相
- 状态机——定义一组状态
- Initial - 定义区域的入口点
- 终点- 定义实体将拥有的最后状态
- 历史记录 - 在退出和重新进入状态时充当纪念品或书签

连接器

- 转移
  - 代表从一种状态到另一种状态的运动

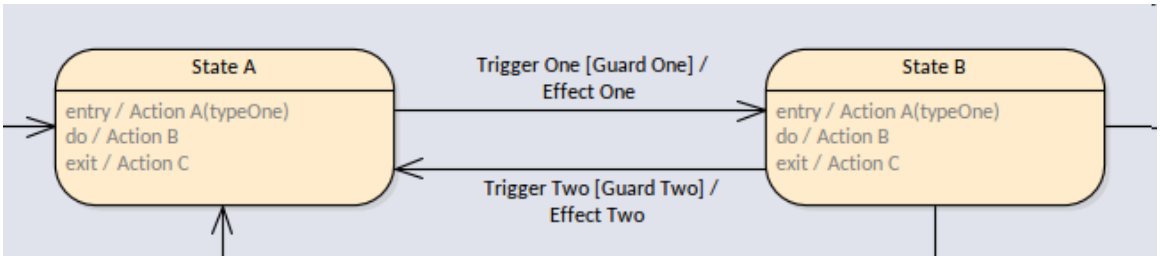


可以通过将元素从工具箱页面拖放到图表画布上来将元素添加到图表中。从初始状态开始和一个或多个终点状态元素开始被认为是一种很好的做法，它应该适当地命名以描述它状态机开始的方式和可能完成的多种方式。将名称留空或给它起一个陈词滥调的名称，例如 开始"或 结束"，将无助于让读者清楚地了解正在建模的系统或系统的一部分，并且可能导致对图表的误解.添加这些节点并在图中适当放置后，可以添加状态和转换，从而定义被建模实体生命周期中的重要阶段。

一旦创建了基本图表，并且随着领域知识和系统行为的进一步发展，就可以在转换中添加触发器、事件和守卫，在状态中添加进入、执行和退出行为。新手通常会认为这些图表是微不足道的，但它们可以揭示否则无法看到的深刻见解。

触发器和过渡

您在状态机图表上看到的大多数连接器将是转换；这些是将一个状态连接到另一个状态的线，表示拥有块（实例）可以更改的允许方式。它们更改的顺序和执行的行为将取决于块的条件和现实世界上下文。例如，在维护工程师纠正故障之前，交通灯可能会闪烁琥珀色，或者飞机可能会保持等待模式，直到控件塔在目的地机场提供着陆许可。该图显示了指向不同方向的两个转换，有效地创建了两个状态之间的循环的可能性。



转换总是源自一个状态并以另一个状态为目标，包括自转换的特殊情况，其中原点和目标是一个且相同的。图中的线条有一个标签，可以显示许多不同的选项：触发器、守卫条件和影响。我们将在细节中讨论这些选项，因为它们表达了关于转换的重要语义，包括转换是否将被执行。转换可以A三种情况下：

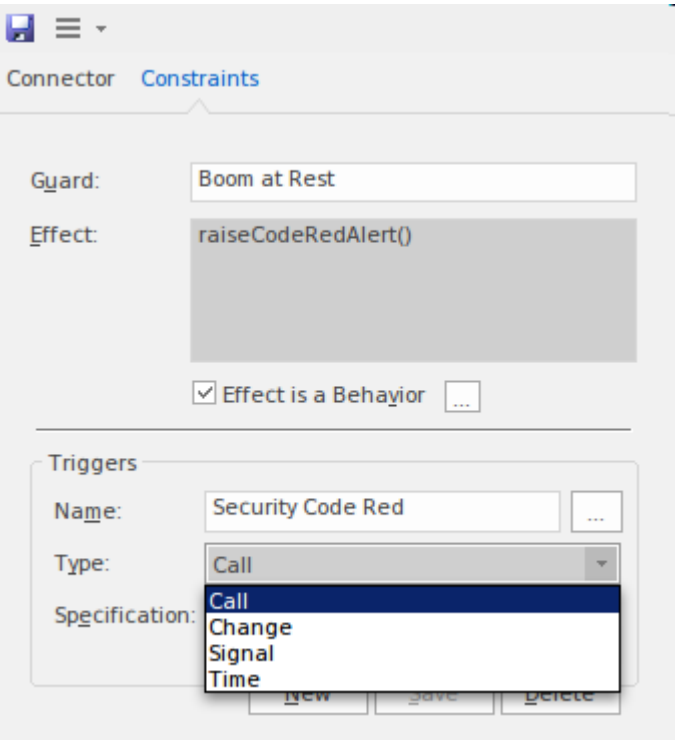
- 已达到- 原始状态（或顶点）处于活动状态并准备好完成其行为
- *Traversed* - 正在执行过渡（包括任何已定义的效果行为）
- *Completed* - 已达到目标状态，准备执行入口行为

这些术语对系统工程师和其他人在协作工作时很有用，讨论状态机的执行及其对拥有块行为的描述。

触发器

触发器，是转换的发起者，并被映射到据说触发事件发生的事件。正是这些触发器及其相关事件导致转换执行（触发）和拥有块从一个状态移动到另一个状态。当状态处于活动状态时，它实际上是在等待被事件触发，并且只要其进入动作完成——不管任何其他因素——它就准备好接收和响应事件。

此屏幕截图演示了如何在触发器Enterprise Architect中输入和查看简单的守卫条件和影响。



此图说明了简单的触发器守卫条件和影响在图表上的显示方式。这种情况下的影响已被定义为一种行为，并已

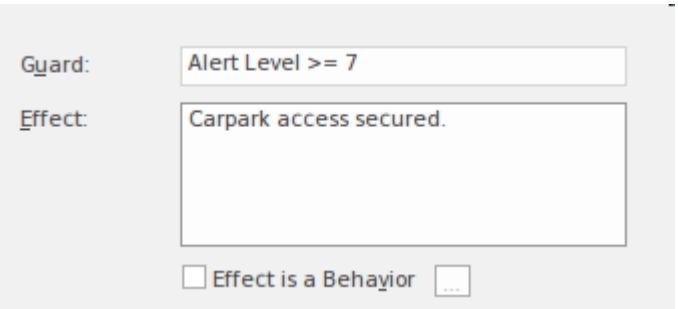
链接到在块上定义的操作。

Trigger \ State		Operating	Locked
		S0	S1
Security Code Red	E0	[Boom at Rest] / S1	
<None>	E1		

警卫

守卫是转换的“守门人”，只有当守卫的表达式评估为True时才会触发转换。如果表达式的计算结果为false ，则事件将被消耗，并且触发器导致块的状态不会发生可观察到的变化。

警卫A表达可以用简单的英语定义，但通常使用正式的约束语言（例如物件约束语言）以约束的形式编写。当使用模拟或可执行状态机时，条件以要生成的代码语言的语法表示，例如JavaScript或 C++。这也适用于效果。在这张图中，我们可以看到一个可以由人或机器评估的数学表达式。



效果

除了将拥有的块从一个状态转移到另一个状态之外，转移

的意义转移

关系是它可以在拥有块或任何其他块上执行可能是活动或操作的行为。此行为是对可能已在源状态上定义的 Exit 行为和在目标状态上定义的 Entry 行为的补充。这确保有一种机制可以在转移

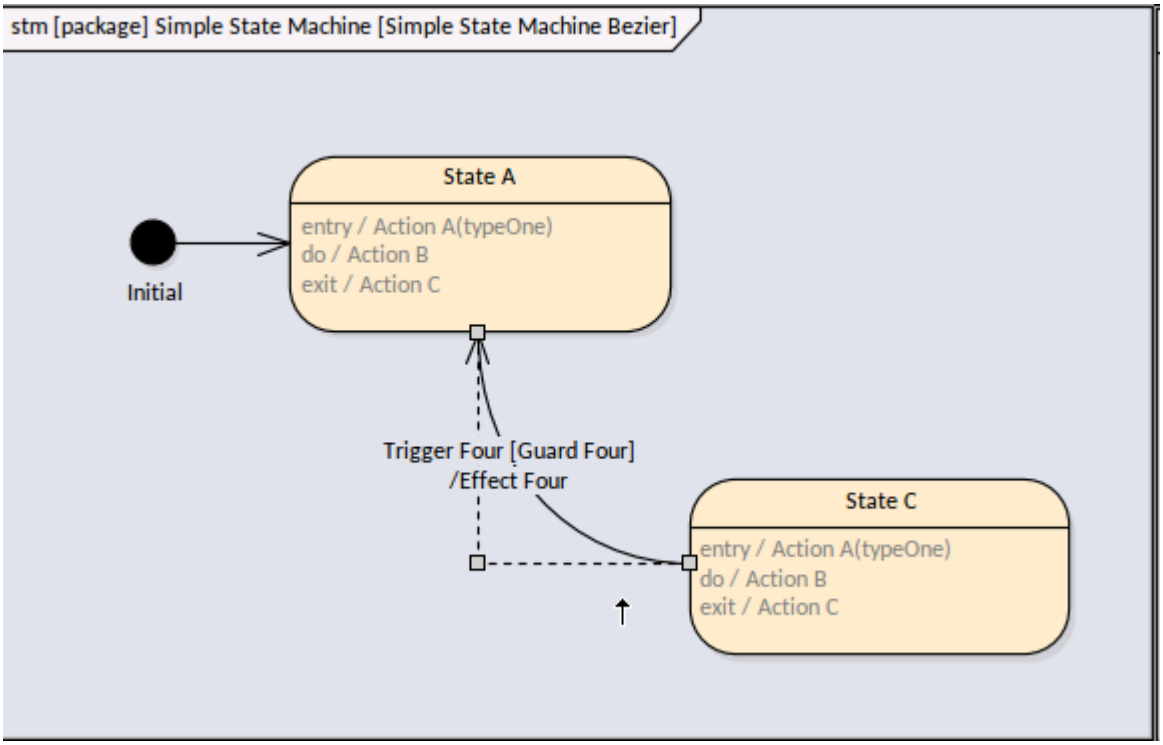
执行期间更改行为转移

.

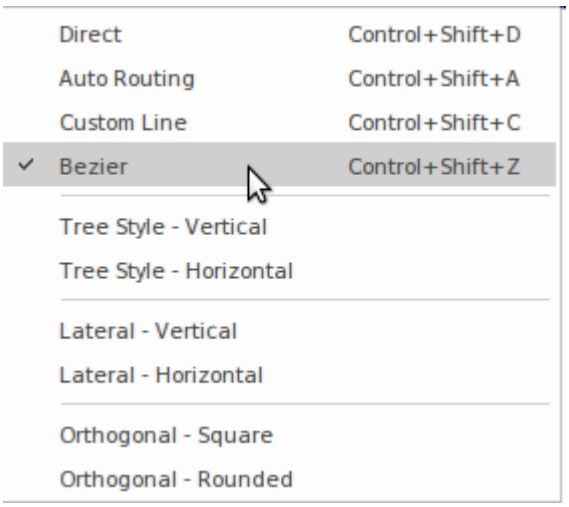
贝塞尔曲线

Enterprise Architect具有广泛的工具和功能用于处理图表，包括状态机图表，这些可用于帮助创建和可视化在状态机中编码的信息。使用转换时特别有用的是路由连接器的能力，这有助于使图表更具吸引力。此图中连接器的线条样式已设置为贝塞尔曲线，给出了转移

不那么僵硬的外观。



可以通过将构造点拖动到新位置来更改曲线的形状。可以使用多种线型中的任何一种，从而为建模者提供用于图表表示的选项工具包。可以选择此上下文菜单并为每个连接器单独设置线型。线条的颜色和粗细也可以从布局功能区中设置。



复合州和区域

状态建模必须是分层的，以处理工程系统的复杂性； SysML 提供了两种机制来在状态机中对这种层次结构进行建模：

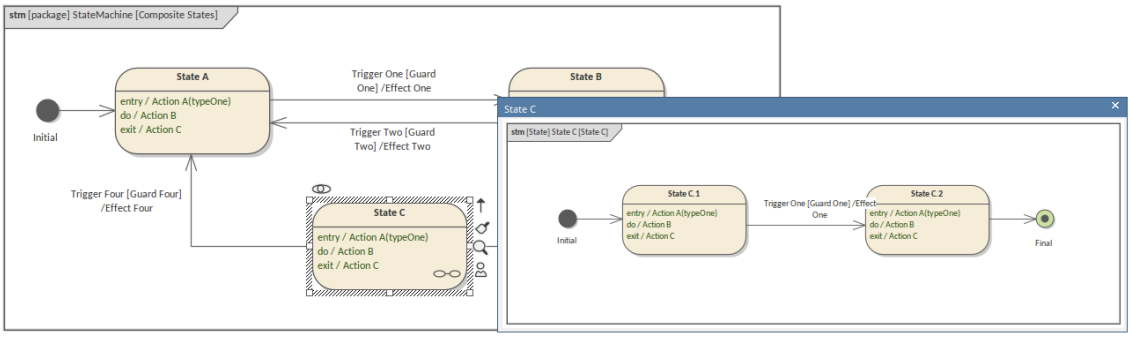
- **区域**-它们是状态机的独立部分
- **复合**- 包含其他状态的状态

系统及其包含的对象通常表现出并发行为，其中两件事可以同时发生；通常这些单独的行为会相互影响，以创建复杂的基于状态的行为。这些行为通常在状态机描述的块中表示，并且可能涉及具有不同生命周期的部分。这些情况可以使用区域进行建模；状态机可以包含任意数量的区域，每个区域都有自己的一组状态、伪状态和连接它们的转换。

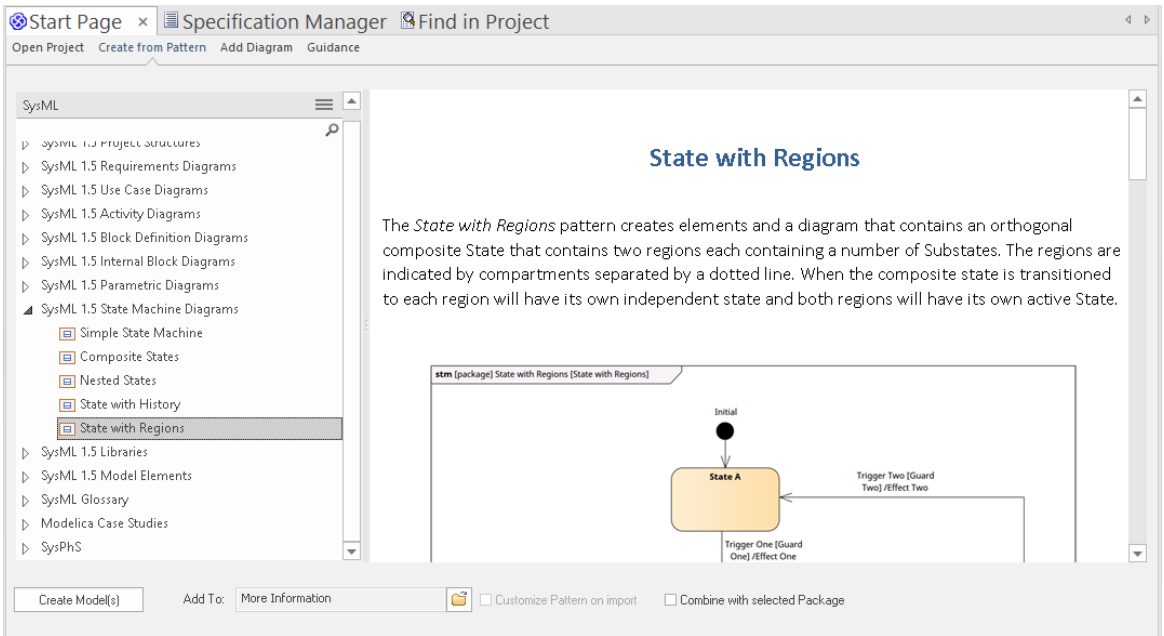
状态本身可能A是分层的，其中单个状态可以分解为多个子状态，这些子状态表示构成复合的一部分的状态。**例如**，机器人可能具有操作和维护；维护状态可以分解为充电电池、更新环境和更新软件模块等多个子状态。这些国家中的每一个都可以依次分解为若干其他国家。

Enterprise Architect为复杂状态行为建模提供了一个有用的起点，它提供了一系列模型模式，可用于对模型状态机的所有方面进行建模，包括建模复合状态和区域。

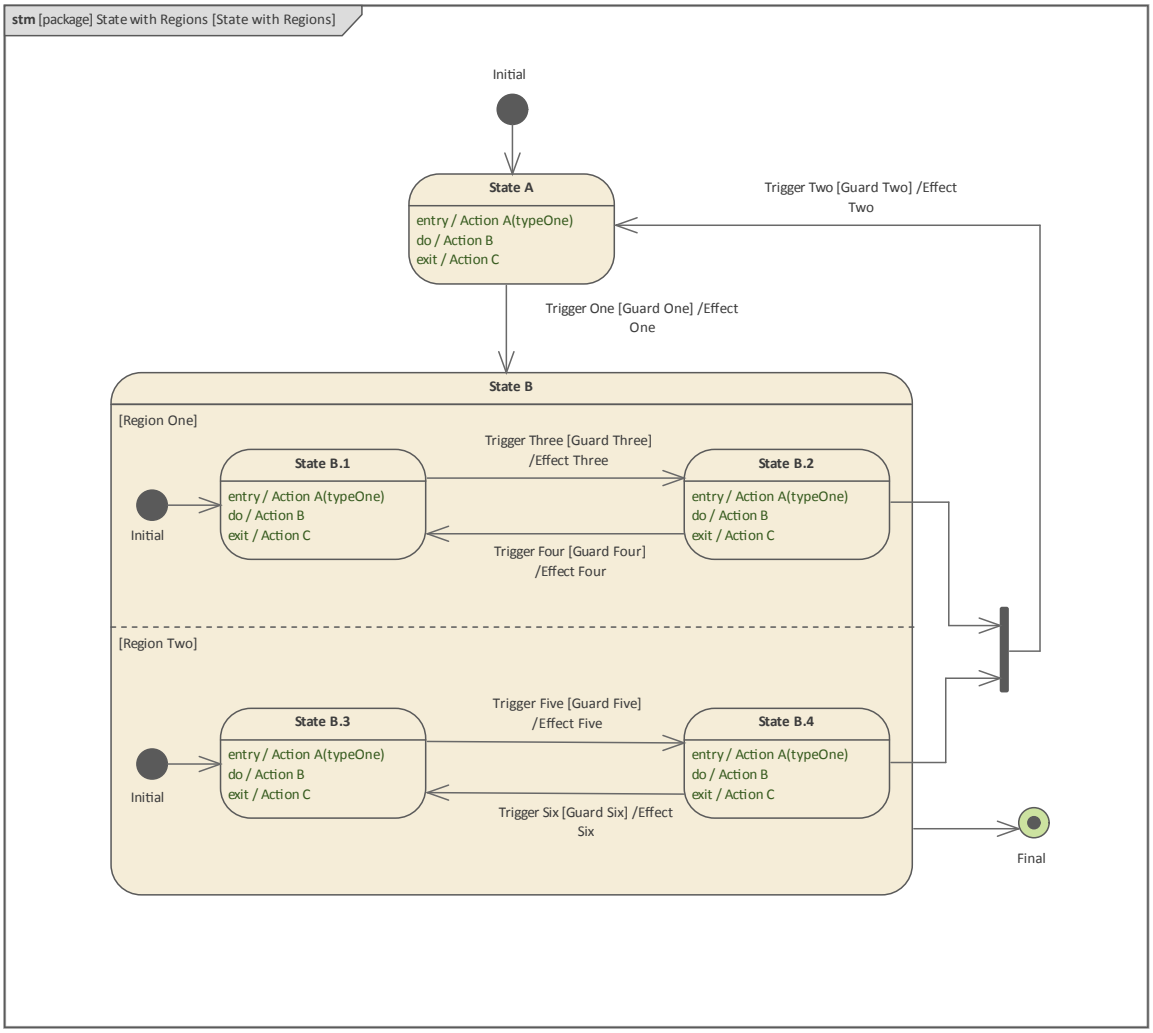
复合状态由 oo（'infinity'）符号表示并引用子图。可以通过单击 图标预览子图。双击图表预览上的元素或将打开图表进行编辑。



可以使用模型向导首页（ 创建自"选项卡）访问这些模式。



此图像显示了从状态区域模式创建的图表，该图表已简单地注入模型并充当建模者的起点。建模者可以通过替换状态和转换的名称以及根据需要添加或删除区域来定制图表。



状态B中的区域提供区域一和区域二中的并行流程。

伪国家 - 交警

转移

关系状态机图中的状态需要进行编排，以确保状态机代表一个块在物理世界中必须改变其状态的自由。伪状态是用于引导流沿转换的节点。节点可以出现在开头、过渡路径上或结尾处。伪状态的类型包括：

- *Initial* - 用于启动状态机
- 分叉和汇合——用 分裂和重聚一个转移
- 终止——用来结束一个状态机
- 汇合- 用于重新组合多个转换
- 连接点-用于分割一个转移
- 入口和出口——用 子机状态边界
- *Deep and Shallow History* - 当复合状态退出时用作纪念品

我们细节详细地研究每一个，并展示如何使用Enterprise Architect创建和管理这些重要节点。重要的是要了解状态终端的终点状态- 它具有初始伪状态图标，状态它本身就是一个。

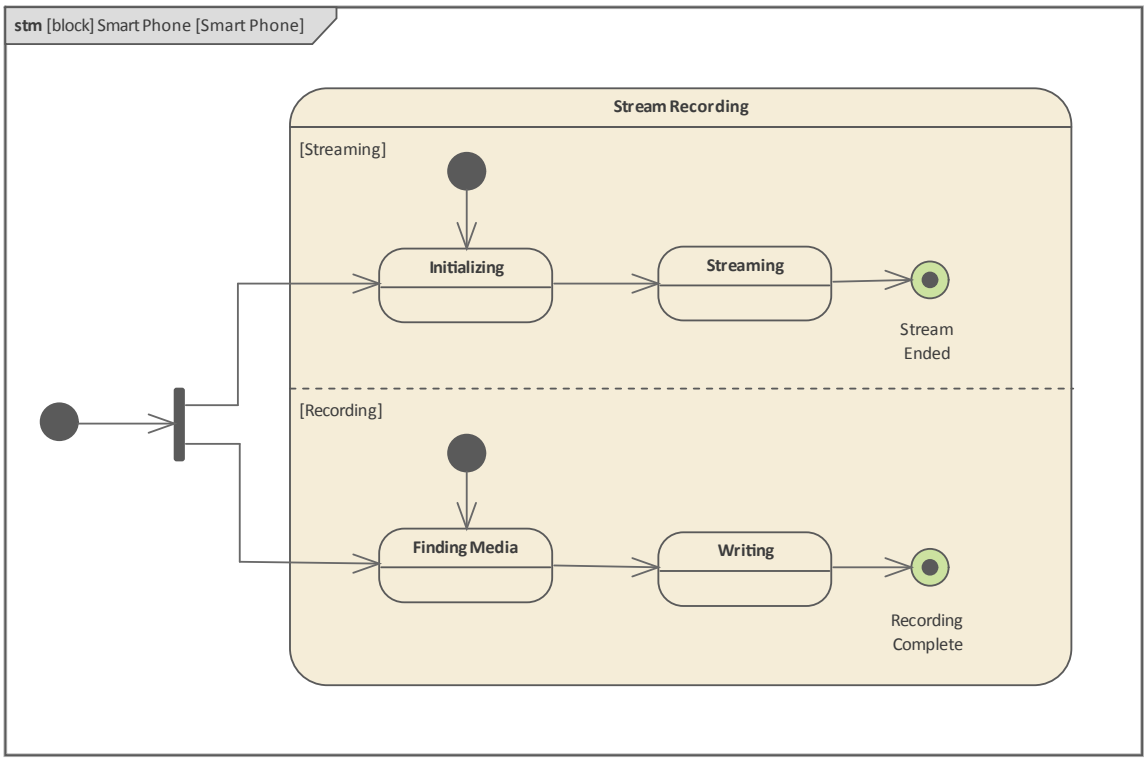
最初的

Initial 伪状态是所有节点中使用最广泛的，代表一个区域的起始点。一个区域中只能有一个*Initial* ，并且允许从 Initial 伪状态出现一个转换。因为它是起点，所以有一个触发器或一个守卫是没有意义的——当进入区域时，伪状态只是变得活跃，但是建模者可以定义一个效果。一些系统工程师将把这个最重要的伪状态从图表中删除，理由是它的位置是隐含的，但将它们包括在内被认为是一种好的做法，因为没有它们，起点是不确定的。但是，通常的做法是不命名 Initial 伪状态。

Enterprise Architect还将依赖于结构良好的机器，并且每个区域都有定义的初始伪状态，当建模者使用可执行状态机状态机运行模拟以可视化状态时，一个块实例将在其生命周期中转换。

分叉和汇合

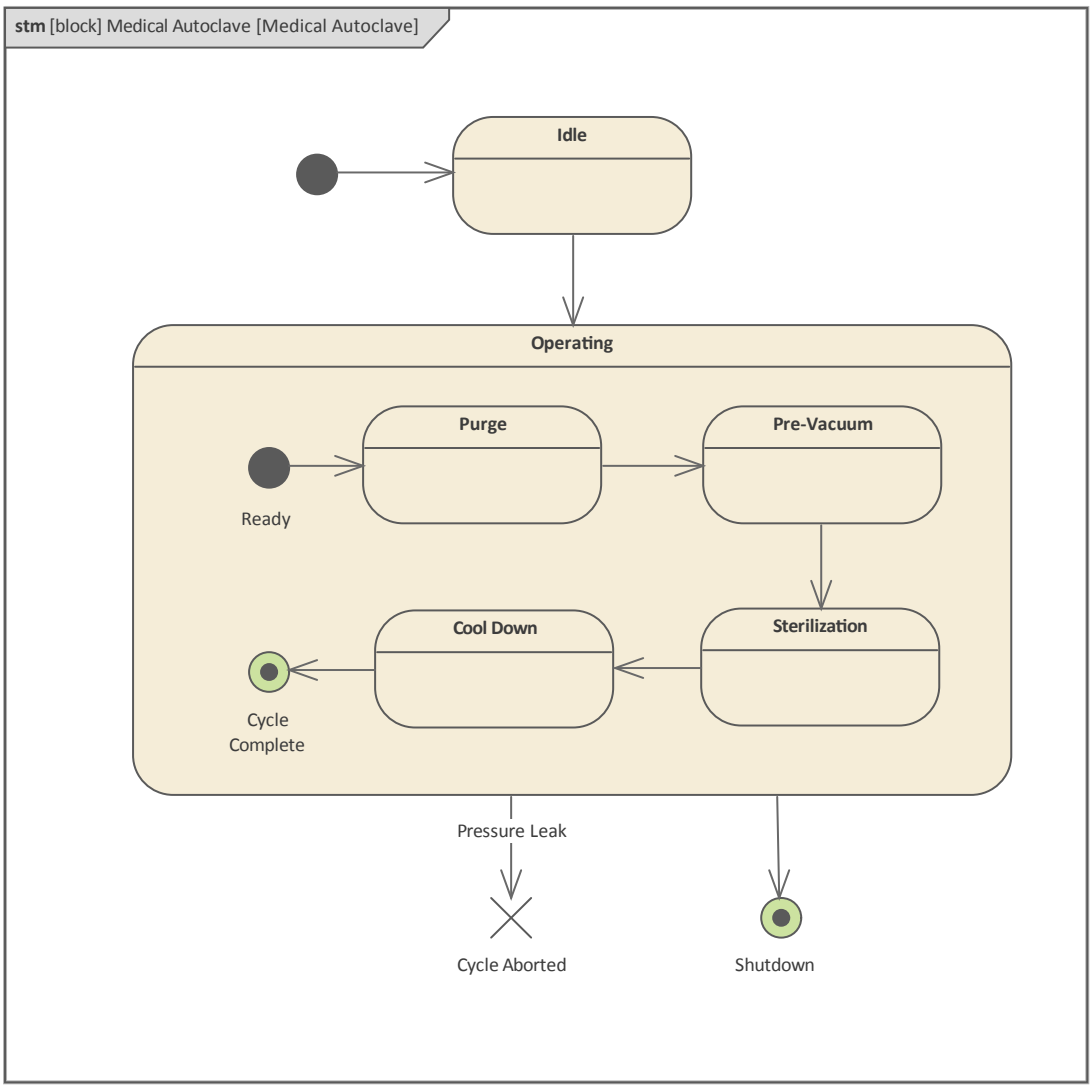
当区域用于状态机图中时，通常需要将一个以状态为目标的转换拆分为多个转换，以便每个传出转换以每个正交区域中的给定状态为目标。来自分叉伪状态的传出转换受到限制，不能定义保护或触发器，但可以定义影响。这确保了多个区域可以同时具有活动状态。



汇合以类似但相反的方式工作，以联合来自多个区域的传入转换。应用反向限制，使得传入的转换不能定义保护或触发器，但可以定义影响。必须先完成所有传入过渡的效果，然后才能触发传出过渡。

终止

终止伪状态是确保整个状态机关闭的有用节点。无论节点位于状态层次结构中的哪个级别，层次结构的所有区域和所有级别都将终止。它是最终节点，拥有状态机将立即停止执行。终止不是 优雅的”，当前正在执行的任何行为都将简单地停止；不会执行任何退出行为。该图显示了具有单个区域的复合状态；如果有压力泄漏，则必须立即终止高压灭菌器的操作，因此存在从操作状态到循环中止终止伪状态的过渡。



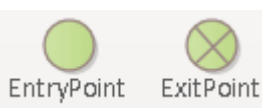
Enterprise Architect允许将此节点放置在任何级别和任何区域中，并将在可执行状态机模拟中遵守其语义。

连接点

连接点伪状态用于具有复合路径的模型转换 连接到连接点的入站和出站转换都有，但在状态机执行期间，**只有**一个入站和一个出站转换会触发。出站转换受守卫保护，只有表达式首先评估为布尔真的守卫的转换才会触发并携带传出令牌。

出口

入口点和出口点伪状态用于允许状态机在多个上下文中被重用为子机状态；它们也可以用于复合状态。这些伪状态显示为跨越复合状态或子机状态边界的小状态；**入口点是空的**，而出口点在圆圈内有一个小 x。



它们在边界上的位置很重要，因为从视觉语法的角度来看，它们允许在元素内部和外部之间传递消息。

深刻和浅薄的历史

History 伪状态就像一个书签或复合状态的纪念品，它只是存储退出区域时处于活动状态的子状态的名称。当随后重新进入该区域时，状态机从历史指定的子状态恢复其转换。有可能由于某种原因重新进入某个区域，而历史无法提供最后的状态；这种情况可以由建模者先发制人地创建一个转移

来处理转移

从历史伪状态到目标默认子状态；转移

只有在历史无法提供纪念品的情况下才会使用。

在此管式离心机示例中，系统工程师在运行复合状态中放置了一个浅历史状态，表示如果在给定状态处于活动状态时退出该启动，然后随后重新进入该状态，执行将在活动子状态恢复。退出历史记录的过渡表明，**如果重**新进入拥有状态并且机器不知道要激活哪个状态，则将选择**冲洗**。

State \ Next State			Initial	Idle	Operating									Shutdown
						Initial	Loading	Separating	Draining Liquid	Concentrating Discharge	Cleaning	Rinsing	History	
			S0	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11
Initial		S0		———										
Idle		S1			———									———
Operating		S2		———										———
	Initial	S3					———							
	Loading	S4						———						
	Separating	S5							———					
	Draining Liquid	S6								———				
	Concentrating Discharge	S7									———			
	Cleaning	S8										———		
	Rinsing	S9												
	History	S10										———		
Shutdown		S11												

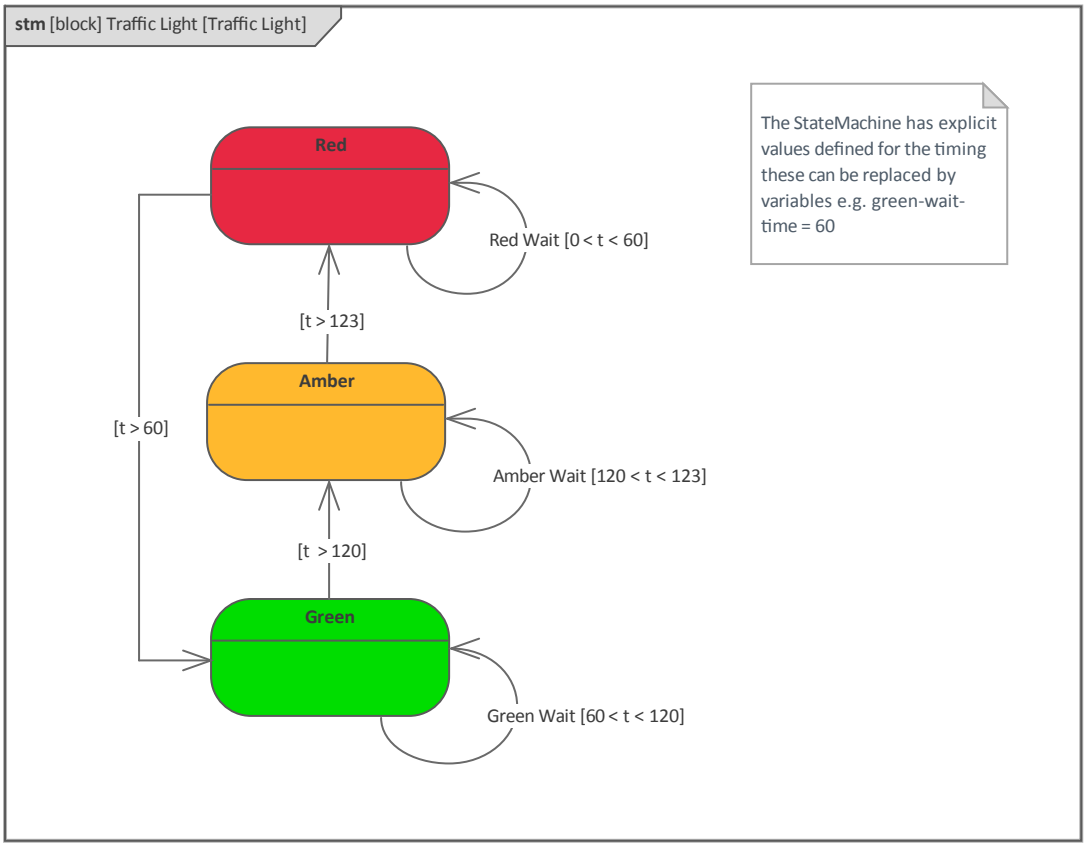
浅历史状态和深层历史状态的工作方式相同，只是浅历史伪状态只记住拥有区域中的活动子状态，而深层历史可以记住子状态层次结构中的任何级别。深度历史通过放置在 H* 之后的星号直观地表示。

状态表-另一种视图

状态机可以A多种不同的方式进行可视化。我们已经看过状态机的图表视图，但Enterprise Architect有一个有用的工具，它允许状态机在表格中可视表。表可视化有三种变体：

- 状态下一个状态视图- 其中状态和子状态被组织为行和列，单元代表转换
- 状态触发器视图- 其中触发器被组织为列，状态被组织为行，单元代表转换
- 触发器的状态视图——其中触发器被组织为行，触发器被组织为列，单元代表转换。

这张红绿灯变化图，和任何状态机图一样，都可以转换成状态库表。



这种视图对某些观众来说会更有吸引力且更自然，工程师可以简单地从一种视图切换到另一种视图。状态及其子状态在表的行和列上都表示，转换（表示状态之间的路径）在单元中表示。

Next State / State		Red	Amber	Green
		S0	S1	S2
Red	S0	Red ... 0 <...		t > 60
Amber	S1	t > 123	Amb... 12...	
Green	S2		t > 120	Gree... 60 ...

如前所述，另外两个表视图允许根据启动它们的简单性来查看触发器。当工程师对因果分析更感兴趣并希望查看或分析事件和触发器如何导致拥有块的状态行为时，这些视图很有帮助。此图显示了相同的交通灯机器，表示为触发器和表。

State / Trigger		Red	Amber	Green
		S0	S1	S2
Green Wait	E0			[60 < t < 120] / S2
Amber Wait	E1		[120 < t < 123] / S1	
Red Wait	E2	[0 < t < 60] / S0		
<None>	E3	[t > 60] / S2	[t > 123] / S0	[t > 120] / S1

Enterprise Architect还允许导出这些库表，以便使用电子表格对其进行分析。这是一种有用的机制，特别是当状态机复杂且有大量转换时。

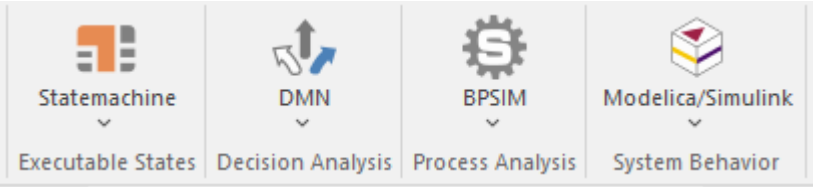
State
Trigger
Red
Amber
Green
S0
S1
S2
E0
Green Wait
[60 < t < 120]
S2
E1
Amber Wait
[120 < t < 123]
S1
E2
Red Wait
[0 < t < 60]
S0
E3
<None>
[t > 60]
S2
[t > 123]
S0
[t > 120]
S1
State Table Options...
Properties...
Lock Diagram
Save Current Changes
Control+S
Statechart Editor
Export Statechart to CSV file...
Execute Simulation
Help...

使用模拟进行可视化和实施

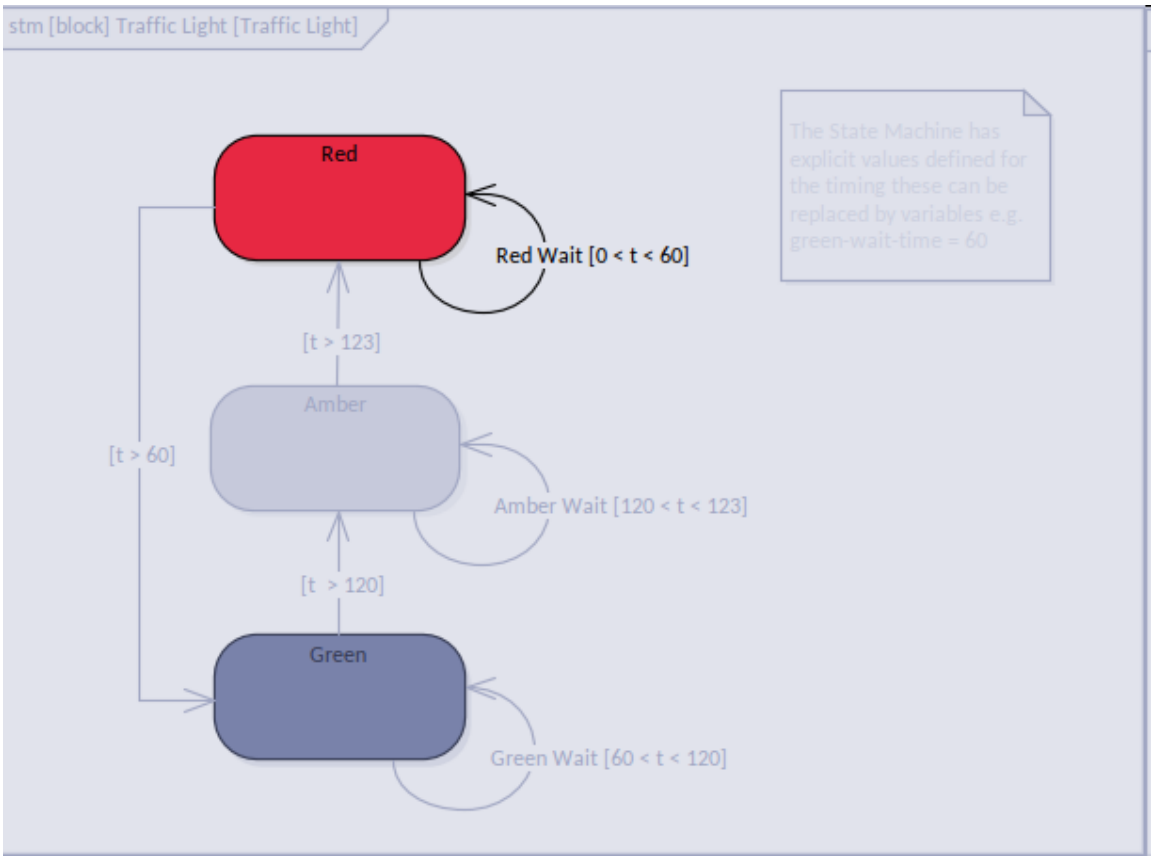
Enterprise Architect不仅是一个创建和管理状态机图表的平台，它还为工程师和其他利益相关者提供了复杂的仿真功能来可视化状态机。这使模型栩栩如生，并提供了一个可视化工具，不仅适用于开发模型的工程师，也适用于需要了解模型所说内容的其他技术人员和非技术人员。这有点像作者大声朗读一段新写的文本，它可以帮助工程师发现模型中的错误或模型中应该纠正或返工的方面。当模型变得更复杂时，它特别有用，具有嵌套的子状态、复杂的触发器和 Guards，以及诸如分叉和汇合之类的伪状态，它们可以拆分和重新组合转换。

凭借对触发器、触发器集合、嵌套状态、并发状态、动态效果和其他高级模拟功能的广泛支持，该特征提供了一个复杂的环境，用于构建有助于探索、测试和可视化跟踪复杂业务的交互式工作模型，软件和系统行为。有一个专门用于模拟的功能区，它提供了一系列可用于状态机的动态和可执行模拟的项目。此图显示了用于处理动态模拟的核心工具。

第二个图像显示了其他高级工具，包括用于直接从模拟状态机创建状态机可执行状态机执行以生成完全可实现和可编译的编程代码的模拟状态机。该图还显示了许多其他功能，包括 Modelica 和 Simulink，它们是用于运行复杂参数仿真的复杂工具。

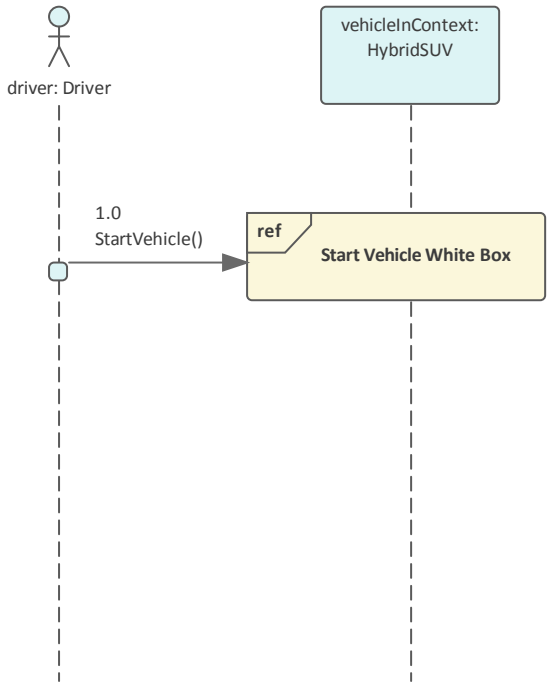


该图显示了我们在上一个练习中看到的交通灯系统的简单动态模拟。它显示了动态模拟的运行情况。



作为消息序列的交互

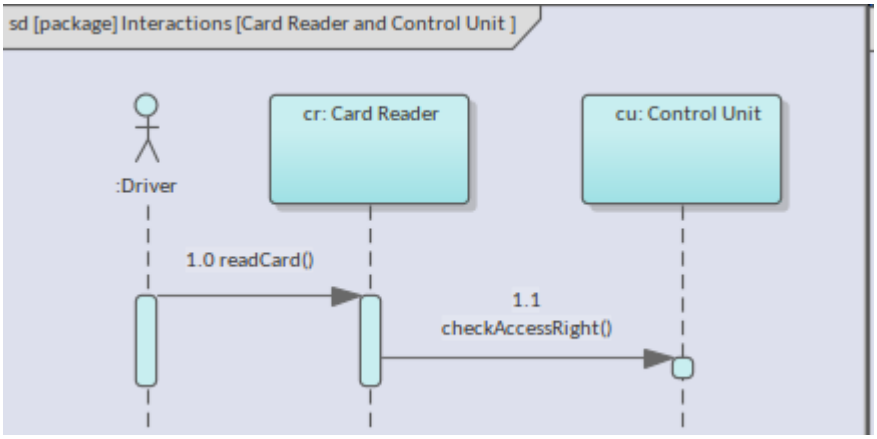
系统由部件组成，系统的整体行为由这些部件以协调的方式协同工作提供。通讯之间的通信和它们行为的同步是至关重要的，无论是从设计角度还是从可视化角度来看。系统的结构单元（主要由块表示）交换触发行为的消息和信号，从而产生代表系统功能的协调系统行为。



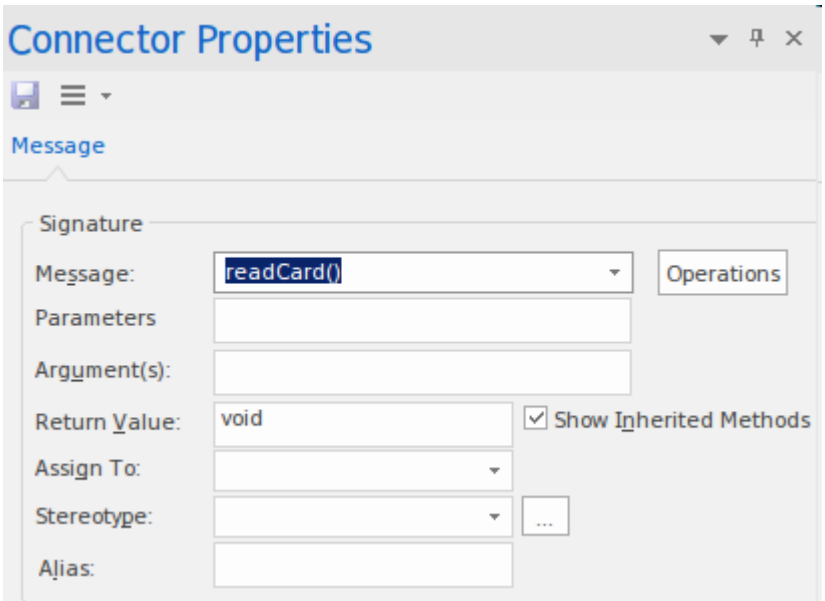
这种消息和信号的交换以及随之而来的行为可以用序列图表示，该序列图显示了参与特定交互的块实例之间的时间序列消息和信号。

生命线、消息和激活

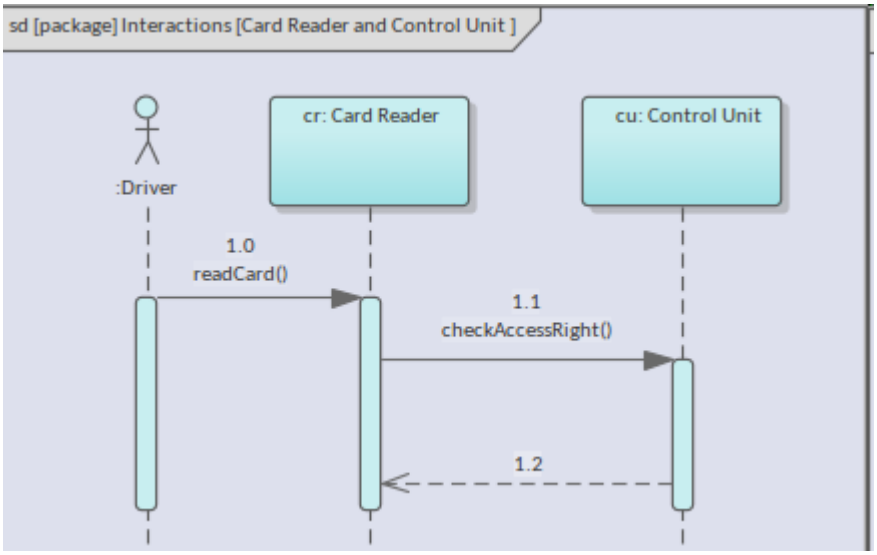
在序列图中，参与交互的块的生命周期由虚线表示，从元素的底部开始，并在元素的生命周期中垂直延续。元素可以在任何时候被创建或销毁。由序列图表示的周期，因此生命线表示它们的存在。出现在图表顶部的元素是在交互开始时创建的。发送者和接收者之间A消息交换将起源于一条生命线（发送者）并结束于另一条生命线（接收者）。



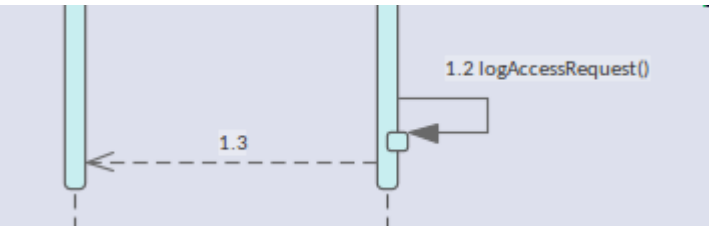
发送者正在有效地调用或激活接收生命线提供的某些行为。 Enterprise Architect提供了有用的机制来利用已经以操作形式定义的行为。此图显示了现有消息的属性，建模者可以从先前定义的操作的下拉列表中进行选择。或者，建模者可以通过选择操作按钮来定义新的操作。



形式上，当消息以生命线为目标时，就会发生执行，这意味着行为被启动或增强。此执行通过激活可视化表示，该激活在图表上绘制为生命线上的薄矩形覆盖，其长度表示行为的相对持续时间。当回复消息被发送回调用者时，矩形激活的范围结束。

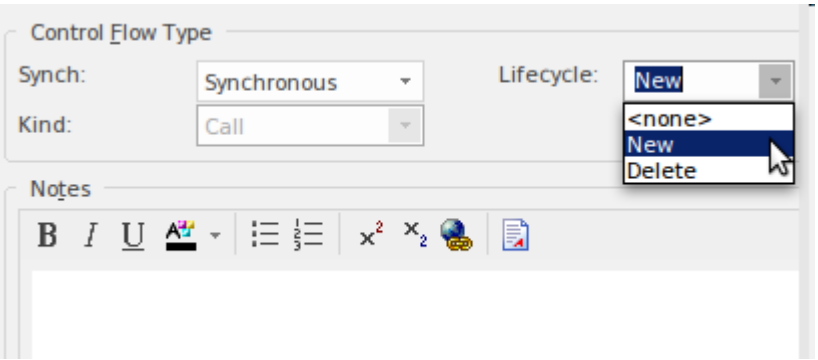


块（Instance）既可以是消息A发送者，**也可以是消息的接收者**，可以称为自反消息，因为它在同一条生命线上开始和结束。在这种情况下，第二个更短的激活矩形覆盖在第一个激活上，**但向右偏移**。

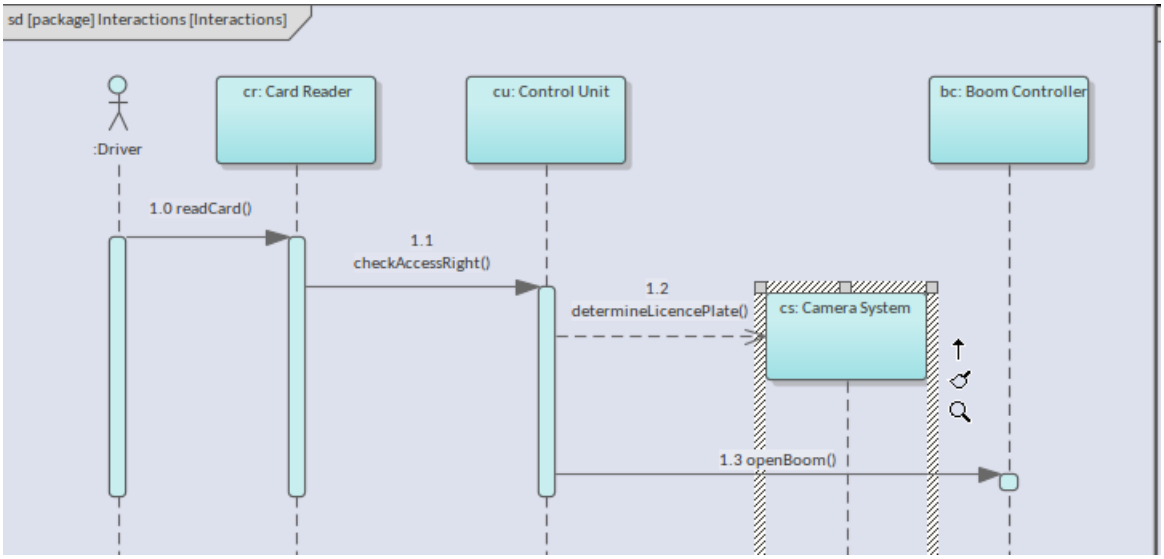


创建和销毁块的消息

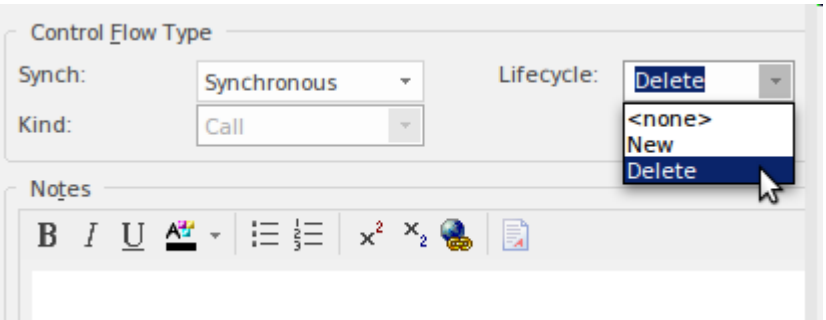
任意数量的块实例都可以构成交互的一部分，并且这些实例的集合通常会在序列图表示的时间段内出现。这些块将位于图表顶部的一行中。但是，可以在图表的序列期间创建和销毁块。例如，一个特定的块可能只需要很短的时间，因此可以被实例化，执行其函数然后被销毁。 Enterprise Architect允许工程师指定消息是创建**消息**，这意味着接收块将在图表的时间序列中的那个点被实例化。这是通过将**生命周期**消息操作设置为 新建”来实现**的**，如图所示。



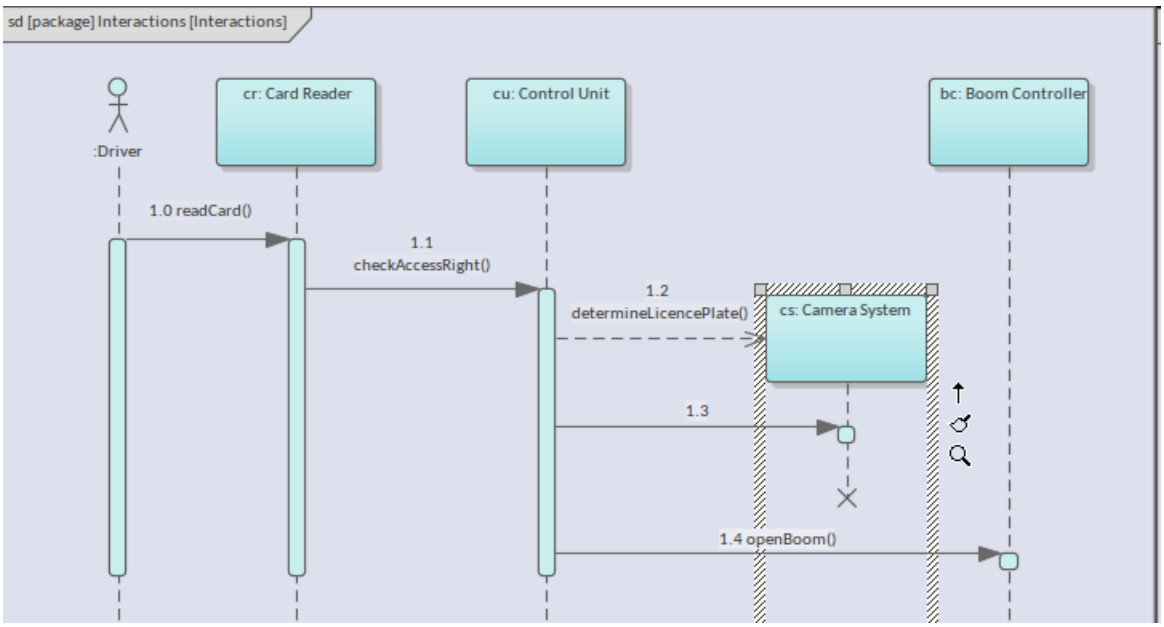
设置此属性具有将目标块移动到图表中较低位置的效果，表示在图表的整个时间序列中直到该点才会创建块实例。消息行样式也更改为带有空心箭头的虚线，以反映它是一个创建消息。此消息的其它和效果的其他方面没有改变。



以类似的方式，可以通过发送*Destroy*消息来结束 Instance 的生命周期。实际上，这意味着该实例已达到其目的并且不再需要。这可以通过再次设置消息属性轻松实现，但这次我们将**生命周期**消息操作设置为“删除”。



设置该属性具有立即终止Instance**的生命周期的效果**；这在视觉上由虚线生命线在消息到达的时间点被生命线上的一个小十字架阻止。

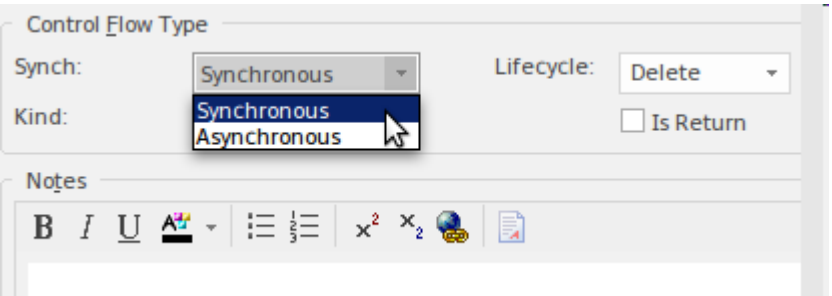


当序列图表示具有有限内存可用的软件系统时，目标实例的破坏将导致分配的内存返回到内存池。在系统工程中，管理机电对象的生命周期可能会带来各种其他收益，例如功耗、过热、可用性，甚至是安全等风险。

同步和异步消息

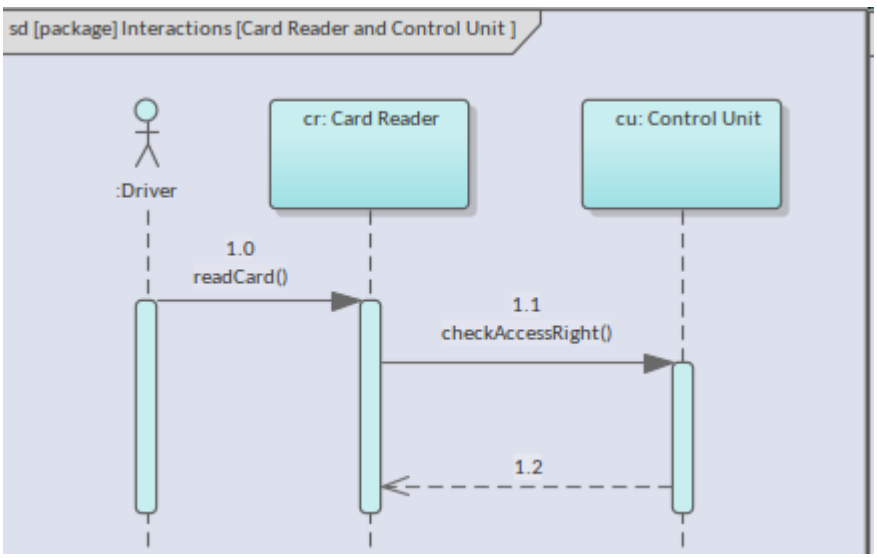
消息本质上表示从发送者向接收者发送某种类型的请求。消息的发送者可以通过两种基本方式与接收者进行交互。第一种类型的消息称为**同步消息**，因为消息发生在大致相同的时间点。对于这种类型的消息，发件人会等到收件人回复后再发送其他消息。第二种类型的消息称为异步**消息**，因为发送者在继续执行之前不会等待接收者的回复，包括向该接收者或其他接收者发送附加消息。

Enterprise Architect默认创建同步**同步**，但可以通过在 ‘属性’属性中设置 ‘同步类型’来更改消息类型。当消息类型设置为同步（默认）时，这会将线条样式设置为实心，并带有指向收件人实例的闭合箭头。该行可以用消息的名称和参数进行注释。



异步消息表示对操作的调用，或发送给接收者的信号；在任何一种情况下，发送者都不会期待回复，也不会暂停执行以等待回复。在调用操作的情况下，操作本身将被定义为异步的，并且图中表示的系统或机器将知道不**等待消息返回**（回复）。 Enterprise Architect允许通过 ‘同步’属性设置此消息类型，如前所述。当消息类型设置为异步时，这会将线条样式设置为虚线，并带有指向接收者实例的开放箭头。该行可以用消息的名称和参数进行注释。

有第三种类型的消息可以选择与同步消息一起使用，这就是回复信息。此消息表示已在接收者上调用的操作，返回该行为已执行且已完成的收据。在图表中包含回复消息是一种风格决定。一些工程师和建模师更愿意将它们放在图表之外，以减少视觉混乱。如果设置了返回类型和值，这将作为回复消息的一部分返回。



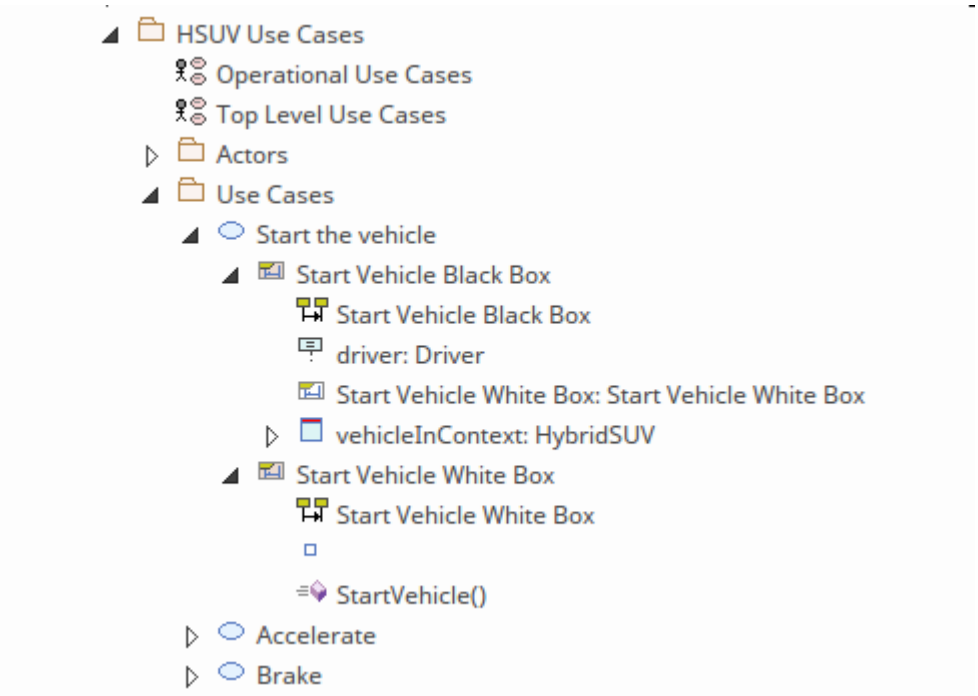
块可以同时指定**操作和**信号接收作为其定义的一部分；该范例描述了操作和信号接收与消息和信号的关系。

- 同步调用操作 -**同步信息**
- 操作的异步调用 - 异步信息
- 接收信号的接收 - 异步信息

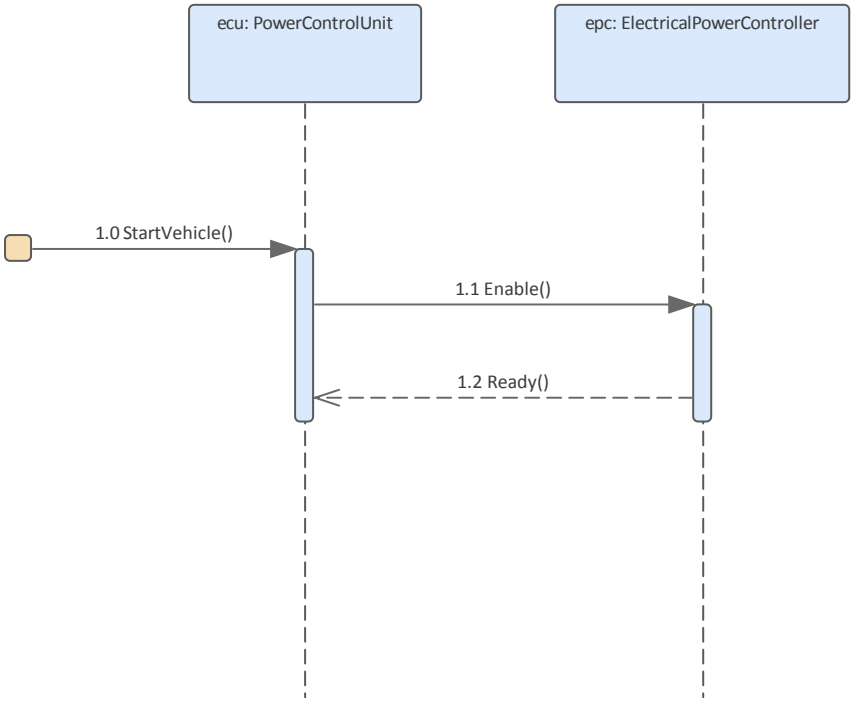
介绍序列图表

序列图起源于Unified Modeling Language ，该语言主要用于表示以软件为中心的系统中组件之间的交互。它的使用在上下文中得到了系统工程，它以更通用的方式用于表示系统的结构单元或系统的一部分之间按时间顺序交换的消息和信号。

序列图有两个轴；按照惯例，**水平** (x) 轴代表参与交互的块（实例），**垂直** (y) 轴代表时间。块不必以任何规定**的方式排序**，但建模者通常会将它们按照最能说明问题的顺序排列，并且该顺序通常基于它们在交互中的使用时间。时间运行以线性比例运行并且任何两个图表之间的时间尺度都可能完全不同。例如，表示高速摄影系统的序列图上的时间刻度与表示杂货收银机的图表上的刻度会有很大不同。此图显示了两个序列图（ 开始车辆用例"和 开始车辆白框"）**的位置**，它们是名为 开始车辆"的子节点。



树形结构充当导航辅助工具，通过双击浏览器窗口中的项目，您可以从此视图打开图表。



在第二个图中，我们看到了一个简单的序列图，它表示启动车辆所涉及的消息的序列。从图中可以看出，有两个Blocks（Instances）构成了交互的一部分，两个Blocks与交互的发起者之间进行消息交换，最终形成了用例。

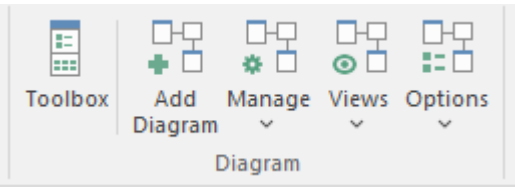
创建图表序列

可以A以下命令从用户接口中的多个位置序列图表：

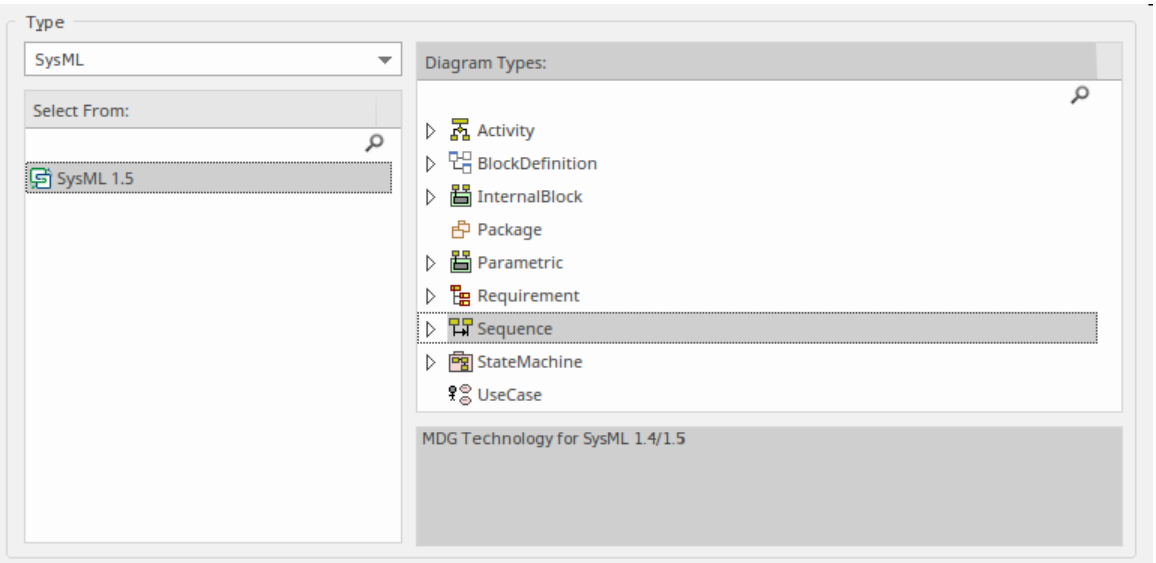
- 设计ribbon -在图表面板上添加图表图标
- 浏览器窗口工具栏 -新的图表图标
- 浏览器窗口上下文菜单- 添加图表

我们将使用设计功能区创建一个序列图。首先，您需要在浏览器窗口中选择要创建序列图的位置。与所有图表一样，这可以是包或元素，但通常将序列图添加到包中，因为它通常涉及包中的许多对象。在浏览器窗口中选择包位置后，选择：

功能区：设计>图表>**添加**



选择此选项将打开 新建图表"对话框，允许您命名图表；**最初**，名称将默认为包含图表的包或元素的名称。选择 SysML 透视图和选择的 SysML **版本后**，将显示图表列表，允许选择序列图。选择确定按钮后，将在浏览器窗口中选择的位置创建一个新的序列图。将打开图表画布，允许您开始添加描述对象之间重要交互的元素和连**接器**。 Enterprise Architect还将显示包含 SysML 规范定义的元素和关系的序列图工具箱页面，这些元素和关系适用于构建序列图。如果需要，除了始终可用的公共元素和公共关系页面之外，还可以打开任意数量的其他工具箱页面。



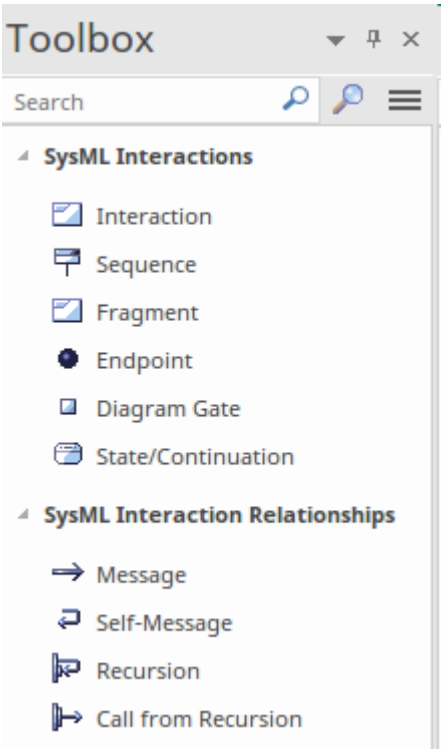
与序列图一起使用的最重要的元素和连接器是：

元素

- 交互
- 序列
- 分段
- 端点
- 图表门
- 状态/延续

连接器

- 信息
- 自我信息
- 递归
- 调用从递归

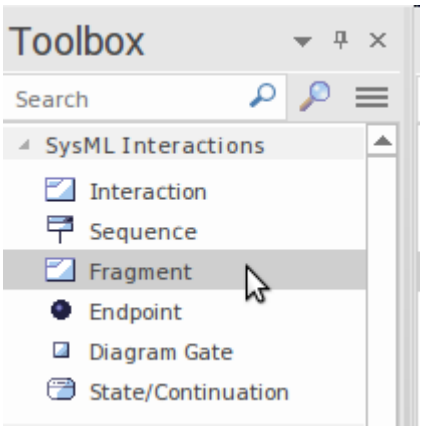


可以通过将元素从浏览器或工具箱到图表画布上来将元素添加到图表中。典型的过程是重用现有元素，例如 Blocks，这些元素具有操作形式的行为，可以选择作为生命线之间交换的消息的基础。元素可以作为链接添加到图表中，但更常见的是它们作为生命线添加。

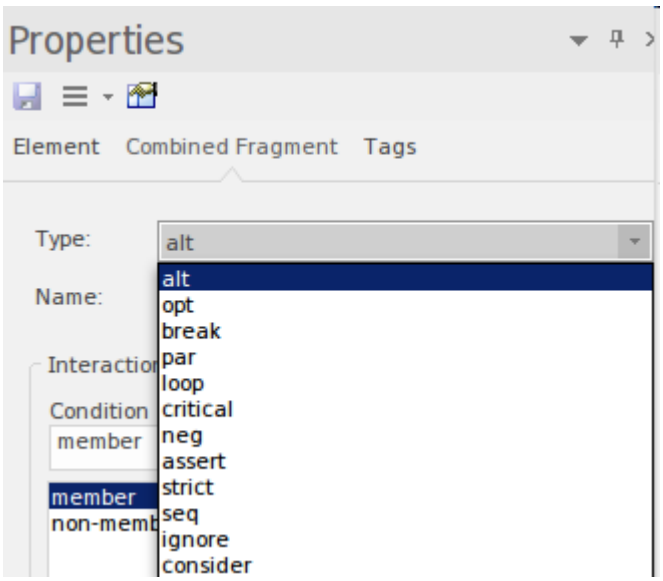
一旦创建了基本图表，并且随着对领域和系统交互的进一步了解，可以添加片段、端点、图表门和状态/延续。

信息配片段

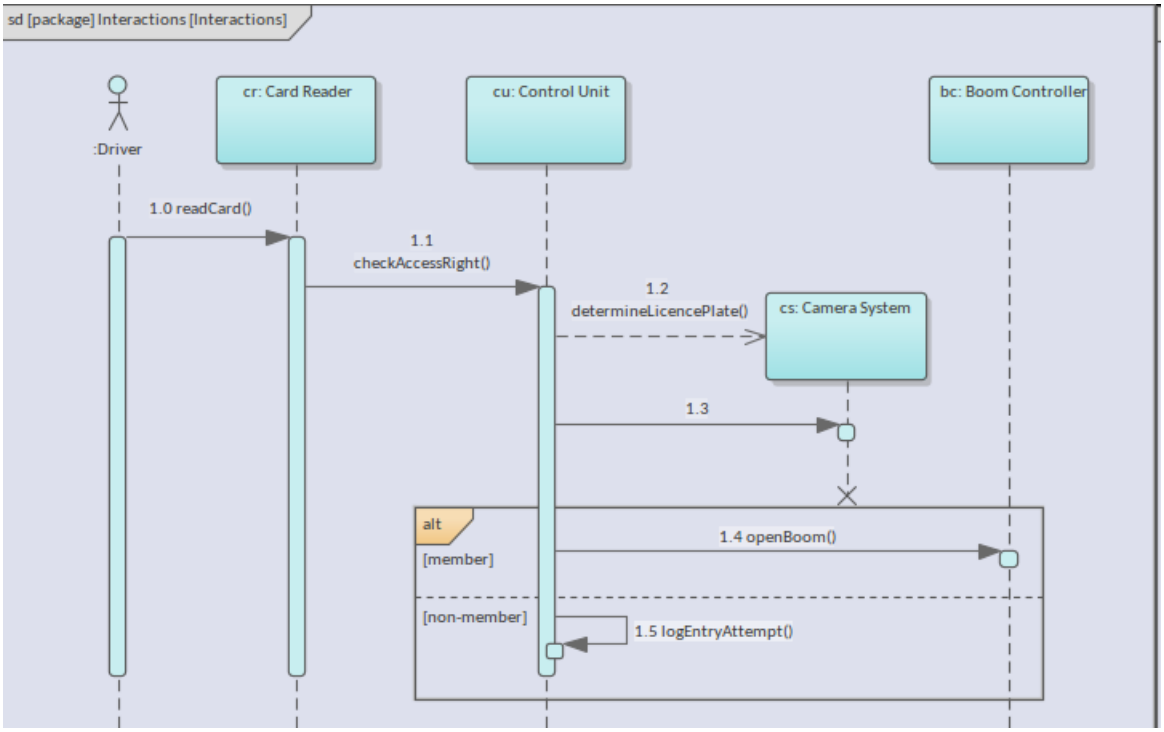
许多系统本质上是复杂的，虽然简单的序列图对于传达一个软件或机电设备的整体图很有用，但它们需要被增强以允许创建这些更复杂系统的复杂模型。在消息流中对复杂性进行建模的选项之一是组合片段。这些可以用于对消息进行不同的序列，包括能够在某些情况下选择特定消息或多次执行消息。有一组可以使用的组合片段，它们的运算符确定片段的类型。 Enterprise Architect支持所有操作员，允许工程师创建能够充分描述正在建模的复杂工程系统的图表。可以直接从交互工具箱页面将片段添加到图表中，并且可以定位以覆盖适当的消息组。



放置在图上的元素是一个通用的 Fragment，需要设置它的操作符。这可以通过在组合片段属性表中选择类型的值来实现。



这会将片段更改为适当的类型，并允许（例如使用*alt*类型）设置任意数量的替代条件，这些条件将确定将触发哪个消息。


sd [package] Interactions [Interactions]
:Driver
cr: Card Reader
cu: Control Unit
bc: Boom Controller
1.0 readCard()
1.1
checkAccessRight()
1.2
determineLicencePlate()
cs: Camera System
1.3
alt
[member]
1.4 openBoom()
[non-member]
1.5 logEntryAttempt()

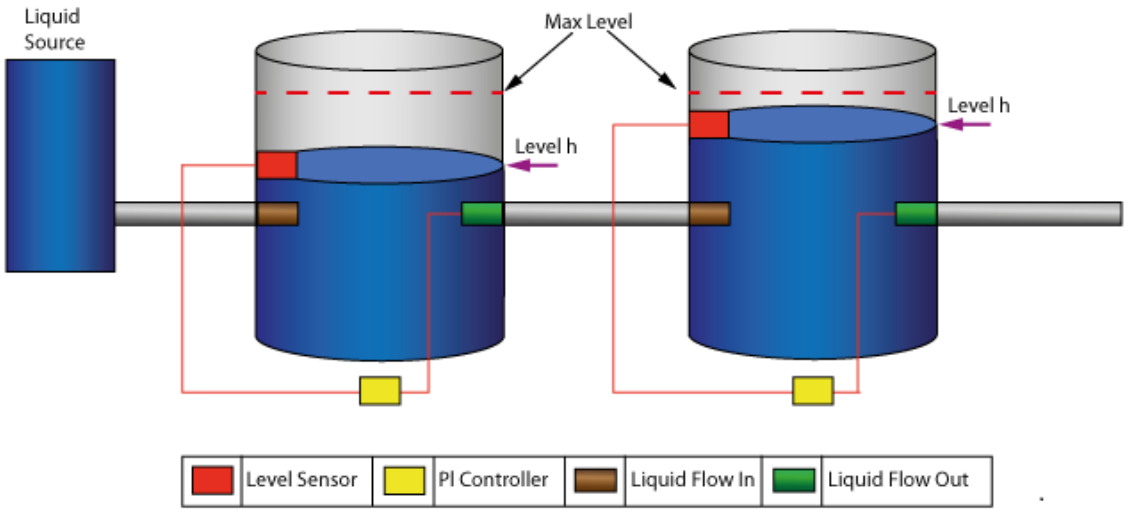
模拟可视化

几个世纪以来，人类通过对数学和物理学的研究以及将思维系统和方程应用于现实世界的问题，扩展了他们对世界的认识。利用这些知识，我们制造了汽车和飞机，将人类送上了月球，分裂了原子并解决了无数其他复杂的问题。在古代，这种知识是通过口耳相传、泥板和纸莎草纸卷轴以及后来抄写员辛苦创作的书籍传播的。然后，随着印刷机的出现，知识以文章、期刊和书籍的形式记录下来并广泛传播。在互联网出现之前的四百年过去了，大部分现有知识将以文档、页面和网站的形式转移到在线材料中，这些材料专门用于这些学科 - 使全球各个角落的大量人都可以使用.现在已经出现了一个新的知识可以用来构建模型的模型，这些模型允许我们可视化这些运动和时间的上下文，具有特定于我们领域的参数以及我们试图理解和解决的问题。

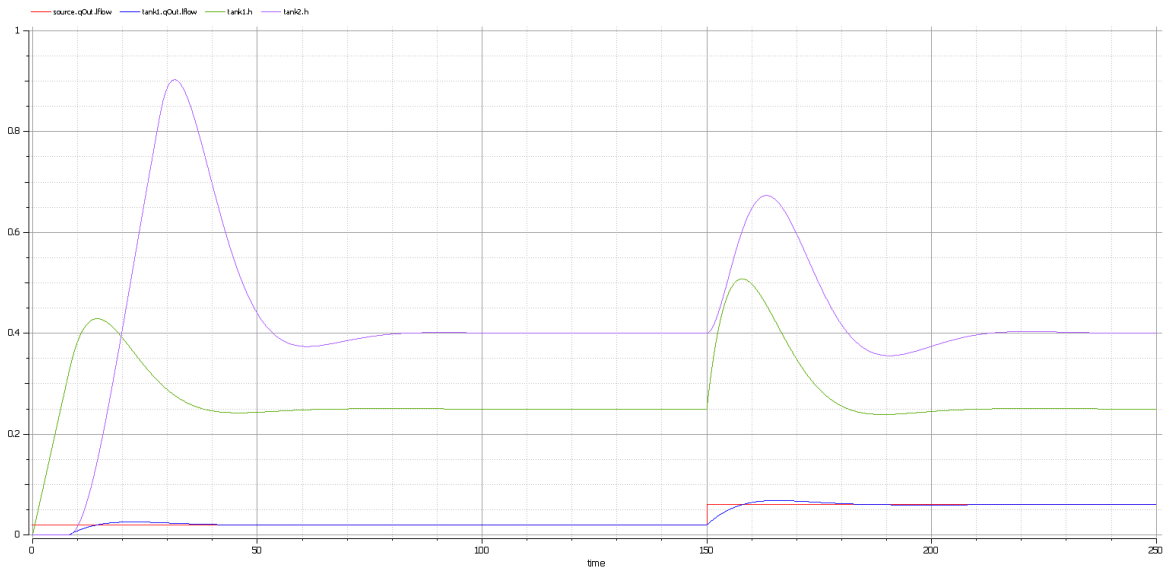
Modelica 和 Simulink 中的 SysML仿真

Enterprise Architect作为领先的系统工程工具，允许使用符合行业标准的建模技术和语言构建模型，以表示网络机械系统。这些模型充当协作工程师、顾问团队和其他人之间通信的设备，但也可用于使用*OpenModelica*和*MATLAB* 的 *Simulink* 使用的行业标准建模语言生成高级可视化。

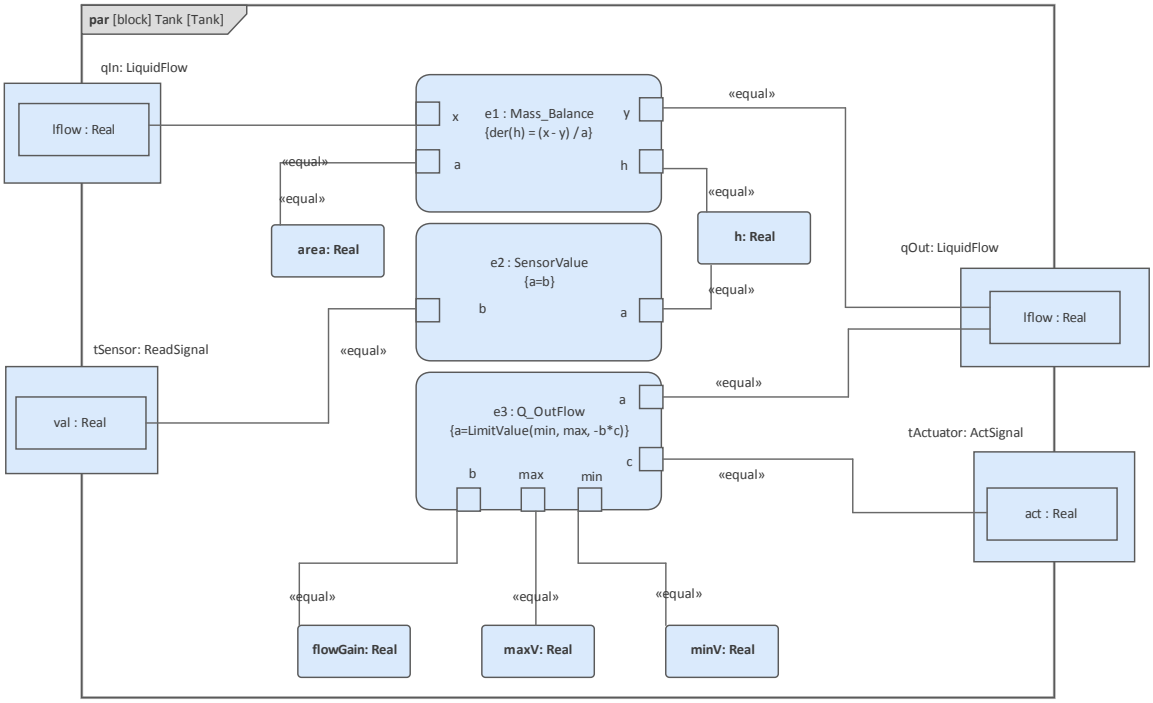
此示例展示了Enterprise Architect在利用现有开放标准来可视化解决方案方面的强大功能。该图描绘了两个连接在一起的源，以及填充第一个水箱的水源。两个连续控制器用于调节从第一个水箱到第二个水箱的水流量，以及第二个水箱的输出。



在Enterprise Architect中创建了A组图表，这些图表对罐的物理方面和表征罐之间流动的方程（用Modelica/MATLAB 语言编写）进行建模。配置好模拟并选择 求解" 按钮后，输出类似于下图：



我们将在本主题的后面部分讨论约束和参数化建模的详细信息，并了解我们创建的模型是如何简单的块定义和我们在指南前面部分中了解的参数图。此参数图显示了两罐问题建模的示例，使用约束属性和将参数绑定到方程组的连接器。其它图是产生结果所必需的，但这是显示质量的主图，流量和传感器确定罐中的液位。


par [block] Tank [Tank]
qIn: LiquidFlow
lflow : Real
x
e1 : Mass_Balance
{der(h) = (x - y) / a}
y
a
h
«equal»
«equal»
«equal»
«equal»
area: Real
h: Real
e2 : SensorValue
{a=b}
b
a
«equal»
qOut: LiquidFlow
lflow : Real
tSensor: ReadSignal
«equal»
val : Real
a
e3 : Q_OutFlow
{a=LimitValue(min, max, -b*c)}
c
b
max
min
«equal»
«equal»
tActuator: ActSignal
act : Real
«equal»
«equal»
«equal»
flowGain: Real
maxV: Real
minV: Real

SysML仿真如何工作

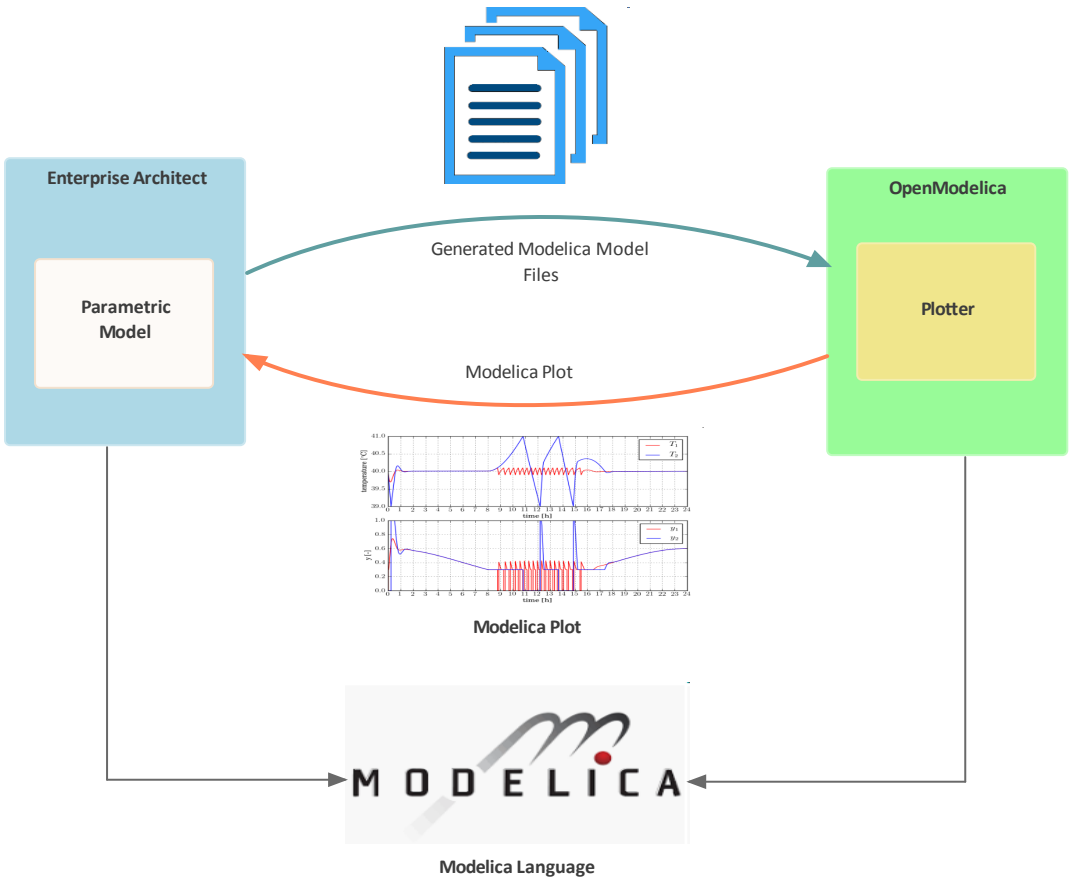
Enterprise Architect建立在开放标准之上，产品团队始终意识到利用现有解决方案而不是重新发明轮子的能力和效率。该工具使用这些标准来实现可视化：

- 系统建模语言(SysML) - 由物件管理组 (OMG) 管理
- OpenModelica - 由源联盟 (OSMC) 管理
- Modelica - 由 Modelica关联管理

我们在早期的主题中花费了大量时间来学习 SysML；事实上，我们已经了解了为这些可视化创建块定义和参数示例所需的大部分内容。此外，我们将学习如何添加一些 OpenModelica 需要的额外信息；这将在下一节中讨论。

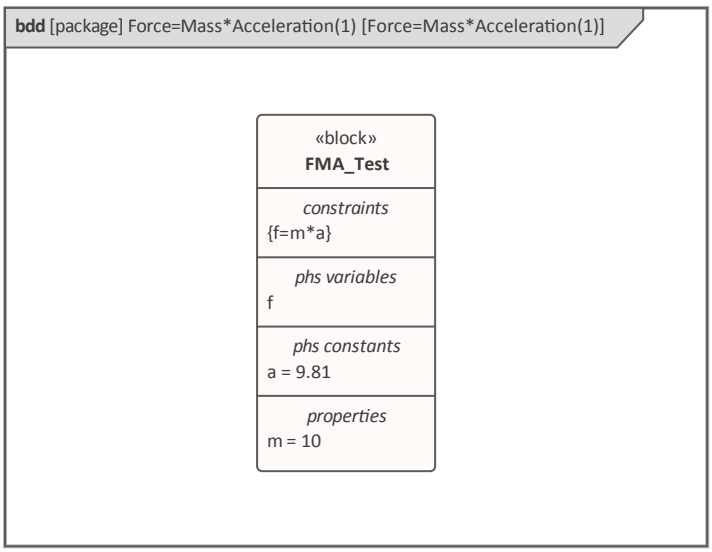
Modelica 是一种基于方程的开放和面向对象的语言，允许利用子组件对网络机械系统进行建模。像它的数学表亲一样，Modelica 是一种跨领域语言，具有广泛的应用，包括机械、电气、电子、液压、热、控制、电力和面向过程的领域，命名为可能的子组件a Modelica模型以及可以使用该工具建模的系统类型。

Enterprise Architect能够为流程图和状态机执行基本模拟，但对于复杂的网络机械系统建模，它利用了OpenModelica 的强大功能，并以 Modelica 语言本身为基础来完成繁重的工作。 Enterprise Architect允许这些网络机械模型与广泛的其他系统和软件工程工件相关联，包括任务、利益相关者要求、状态机、编程代码、决策表、架构、权衡分析等等。

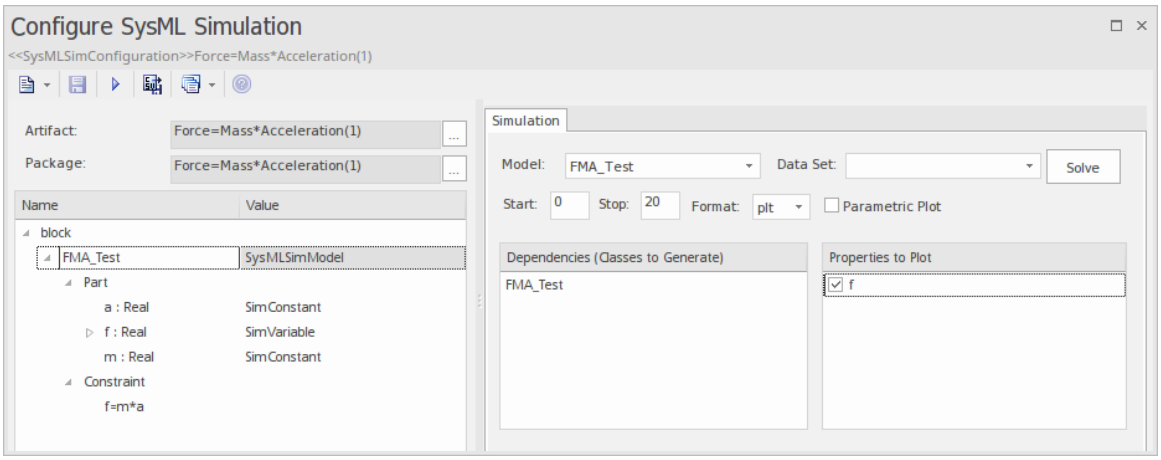


可以通过一个简化示例最好地概述其工作原理。创建牛顿第二定律的简单可视化 - 劲量变化率 作用力成正比，并发生在该力的方向上。" (F = m*a)，使用Enterprise Architect的工程师将：

1 .使用约束和值创建描述方程的块。



2.配置配置工件用于定义OpenModelica所需的信息）



3. 选择窗口上的运行按钮运行仿真器。

将绘制A图表，其中 f = 98. 1 （它是质量 (10) 和加速度 (9.81) 的乘积，用方程 [f = m * a] 表示，其中的值是 98. 1 = 10 * 9.81 ）。这是一个简单的示例，旨在展示创建可视化的基本要素；我们将在后面的部分中查看更高级的示例，展示约束属性和用户定义数据集的使用。

使用开始

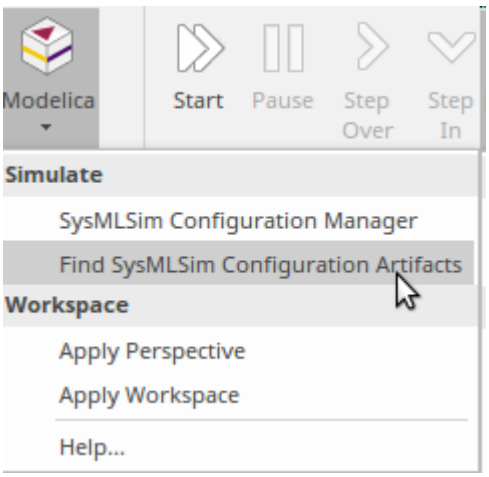
开始使用 OpenModelica 和Enterprise Architect的机器来生成参数模拟的一种简单方法是查看一些现有示例。对于Enterprise Architect中的任何特征来说，这都是一个有用的学习工具，但在学习 OpenModelica 的强大功能时尤其有用，因为有许多新东西需要学习，最好通过一些学习辅助工具来完成。我们将从查看示例模型中的完整示例开始，该示例模型随Enterprise Architect的每个安装一起分发，可从开始功能区上的帮助项目中获得。

我们将探索的示例是 Pendulum 示例，但首先我们需要打开示例模型，我们通过选择功能区选项来完成：

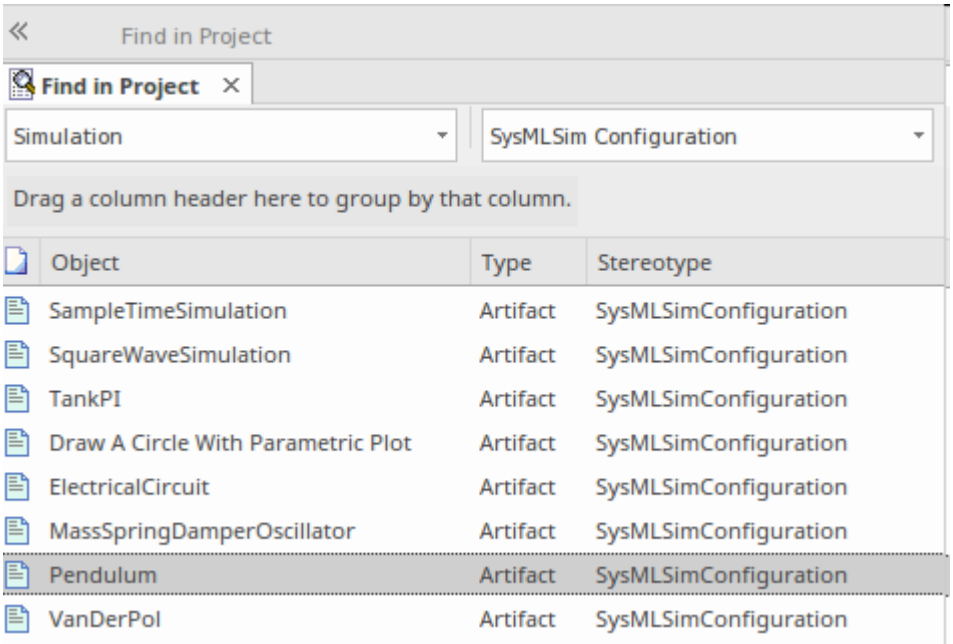
开始>帮助>帮助>资源>打开示例模型

OpenModelica特征与其他仿真功能方便地组合在一个位置，并且可以使用 仿真"功能区的 系统行为"面板中的 Modelica/Simulink 菜单启动。 OpenModelica 为公司提供其他仿真工具，例如可执行状态机、决策建模功能和功能。

加载模型后，我们可以使用Enterprise Architect中的 OpenModelica特征来定位 Pendulum 模型中的例子。复杂系统的模型通常非常大，并且可以设置任意数量的现有模拟，因此Enterprise Architect提供了一种机制来搜索这些模拟。

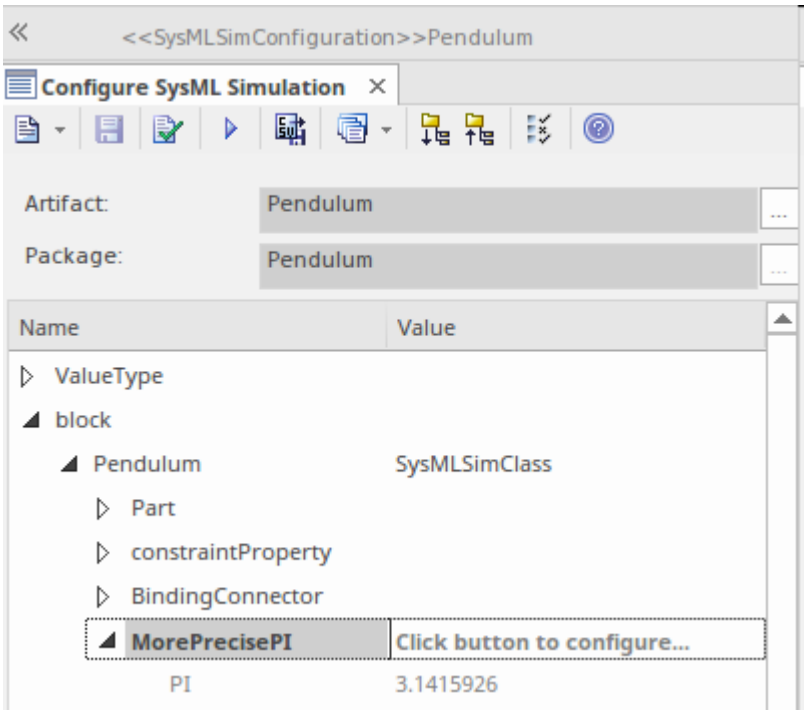


选择此选项将返回 SysML 模拟配置工件的列表，它们是指定 OpenModelica 详细信息的元素。从这个列表中，我们可以选择 Pendulum 示例，它将启动一个窗口，该窗口可用于配置和解决示例中编码的问题。为了能够运行模拟，必须在运行Enterprise Architect的机器上安装 OpenModelica 版本。下一节中包含有关如何安装 OpenModelica 的详细信息。

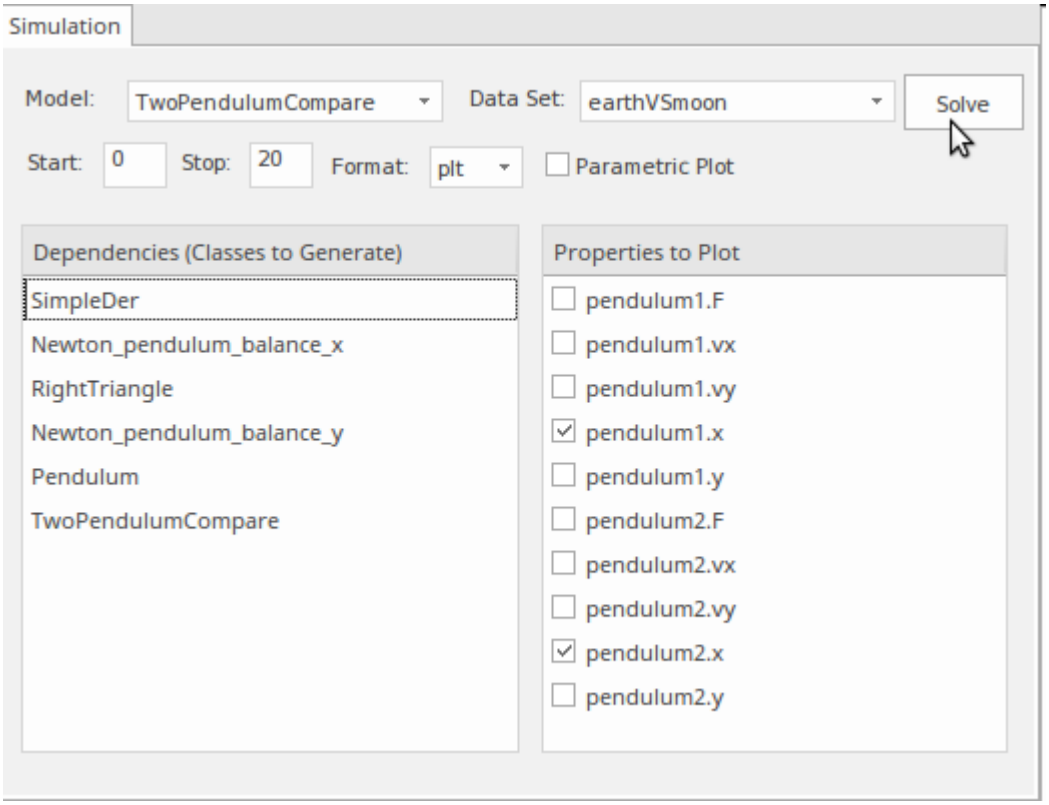


窗口有两个部分：

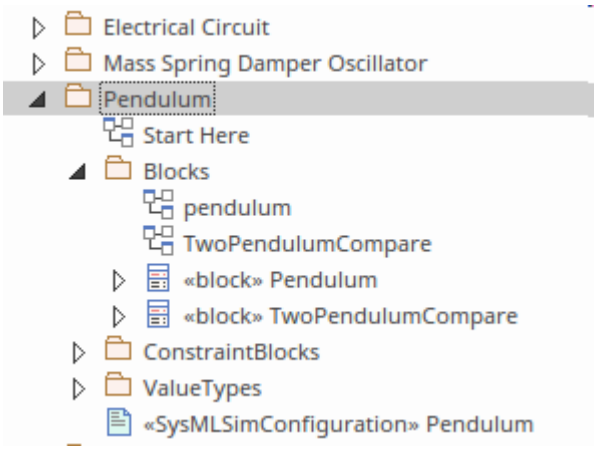
- **配置**（左侧）面板定义了值类型、块和约束属性，所有这些都在存储库中定义，并且 - 正如我们稍后将看到的 - 可以在浏览器窗口和图表中查看



- 模拟（右侧）面板用于选择和指定将影响模拟的选项，例如数据集、模拟的等待时间和持续时间、输出文件类型等。



窗口中的信息是从浏览器窗口中可见的模型自动填充的，属性可以使用 在项目中查找浏览器"选项找到元素的位置（包括值类型、块和约束）从上下文菜单。还有一个类似的选项可以在它出现的任何图表中找到选定的元素。



安装 OpenModelica

Enterprise Architect利用了运行平台的强大功能，因此当您从仿真窗口运行仿真时，它实际上是在调用OpenModelica（安装在同一台机器上）来完成繁重的工作并返回仿真结果。这确保了Enterprise Architect能够利用这个开放工具的强大功能以及所有为其卓越做出贡献的聪明才智。 OpenModelica 有窗口和 Linux 版本，您需要安装适合您环境的版本。这里总结了这些步骤。

1. 下载 OpenModelica 软件（窗口或 Linux）。
2. 安装软件。
3. 选择安装。
4. 通过在Enterprise Architect中指定路径来配置Solver。

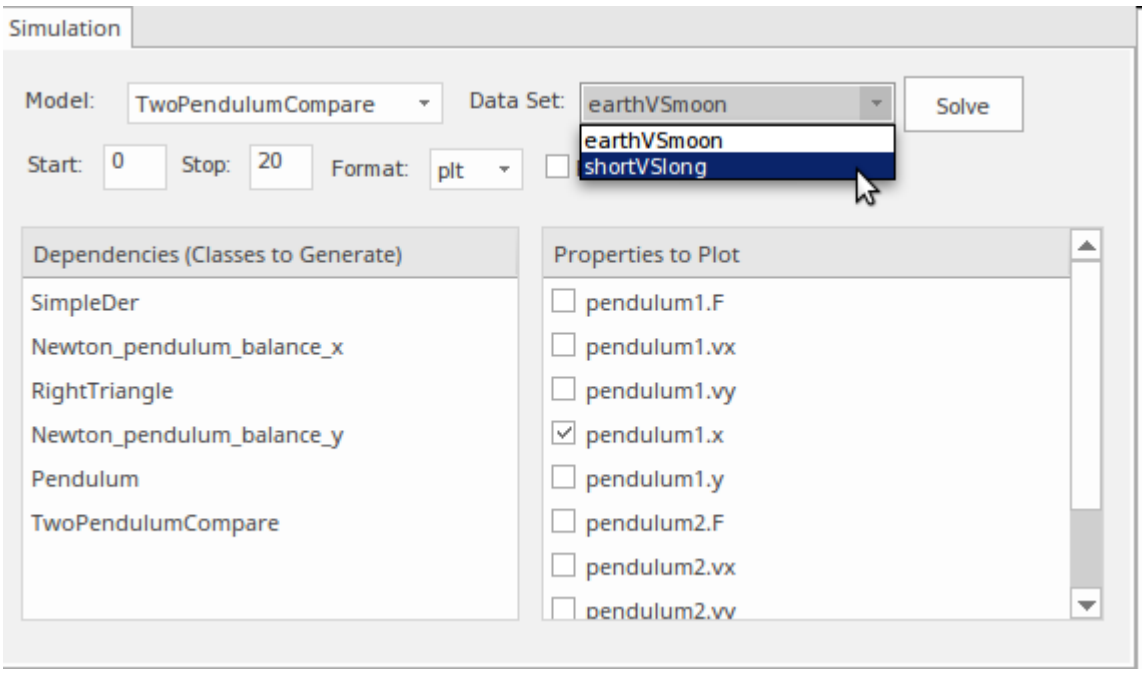
安装和配置的完整细节可以在OpenModelica Integration帮助主题中找到。

使用数据集进行仿真

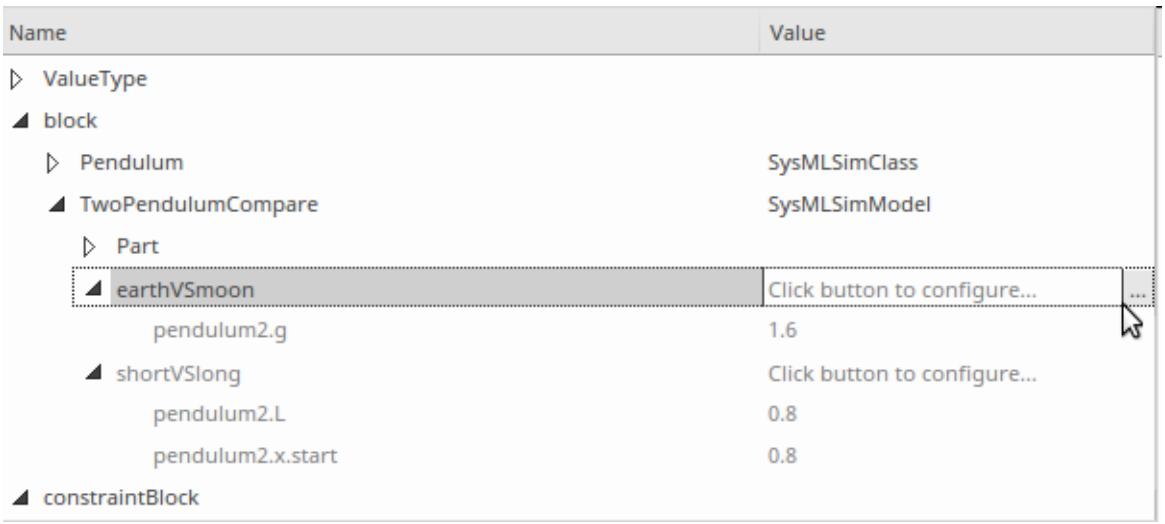
仿真是工程分析的一个重要方面，并提供了一种分析系统行为的有用且具有成本效益的方法。该系统可能已经建成，或者可能正在计划开发；无论哪种方式，仿真都可用于可视化组件或系统设计考虑、性能优化、安全工程等等。为了提供这种能力，重要的是能够使用不同的变量和常数值来运行模拟。例如，在我们的钟摆示例中，我们可能想要分析系统在不同行星上的表现，例如：

- 地球与月球（或其他行星）引力的影响
- 不同string长度的影响
- 不同质量的影响
- 不同起点的效果
- 以上任意组合

Enterprise Architect为可以在块级别应用的模拟提供了数据集功能。使用配置窗口的 仿真"选项卡（右侧面板），我们可以选择预定义的数据集。



可以定义任意数量的数据集，并将其添加到仿真窗口的 配置"（左侧）面板中的适当块中。这些可作为每个块下的节点使用，并且可以使用数据集行上的浏览 [...] 按钮在窗口中查看和编辑数据值。



选择后，将打开配置仿真数据窗口，允许查看和编辑、导入或导出值。这种机制意味着模拟机器可以在许多不同的环境中重复使用，而聚焦于特定类型问题的工程组织可以创建可以在多种环境和工程问题类型中重复使用的模拟库。此处所示的窗口显示了一个数据集，其中包含与我们一直在研究的两个钟摆问题有关的值，我们可以看到作为示例，由于月球重力引起的加速度被定义为1 .6 m/s2 的近似值，大约为 16.6地球表面价值的 %。这种模拟可以与适用于火星或木星的不同数据集或在具有不同质量或长度的string的更陆地示例中重复使用。

Attribute	Stereotype	Type	Default Value	Value
◢ pendulum2	SimVariable	Pendulum		
x	SimVariable	Real	0.5	
▷ L	SimConstant	Real	0.5	
▷ m	SimConstant	Real	1	
F	SimVariable	Real		
▷ PI	SimConstant	Real	3.1415926	
vy	SimVariable	Real		
▷ y	SimVariable	Real	0	
▷ vx	SimVariable	Real		
▷ g	SimConstant	Real	9.81	1.6
▷ pendulum1	SimVariable	Pendulum		

创建仿真模型

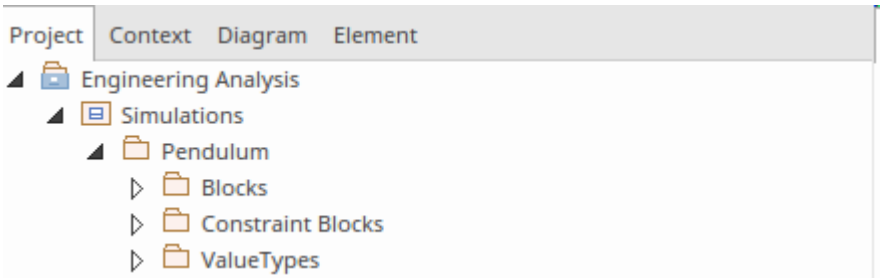
我们在本指南的 "工作原理"部分中看到的示例是故 微不足道的；在本节中，我们将详细细节这些步骤，包括配置模型的选项。 Enterprise Architect提供了很大的灵活性，允许定义简单的一次性模型，或者创建基于库的复杂多用途模型，这些模型可以使用不同的数据集和上下文在多个项目和域中重用。在本主题中，我们将探讨其中一些用于配置模型的选项，以便它们适合用途并创建所需的工程成果。

仿真模型包

可以在浏览器窗口中A任何位置设置模型包，但通常认为最佳实践是在每个项目或计划下设置模拟包。然后，这可以包含每个模拟的子包。预计对于给定的项目，可能需要许多不同的模拟。正如数据集部分所讨论的，组织也可能希望在多个项目中重复使用一些模拟。这些模拟可以在项目浏览器中设置并在超项目级别定义 - 例如在企业、组织或工程部门级别。然后可以将它们包含在项目级别的图表中，表明它们是适用于给定的项目或问题上下文。工程团队也可能希望在项目之间重用值类型及其伴随的数量和单位库，正如本指南前面的主题中所讨论的，最好在超项目级别定义和建模。

Enterprise Architect使用 SysML包导入机制来确保在企业级别定义的 Value Types 可以在每个仿真包级别等级中包含和重用。该结构通常包含以下包：

1. 值类型（特定于此项目）
2. 块
3. 约束属性



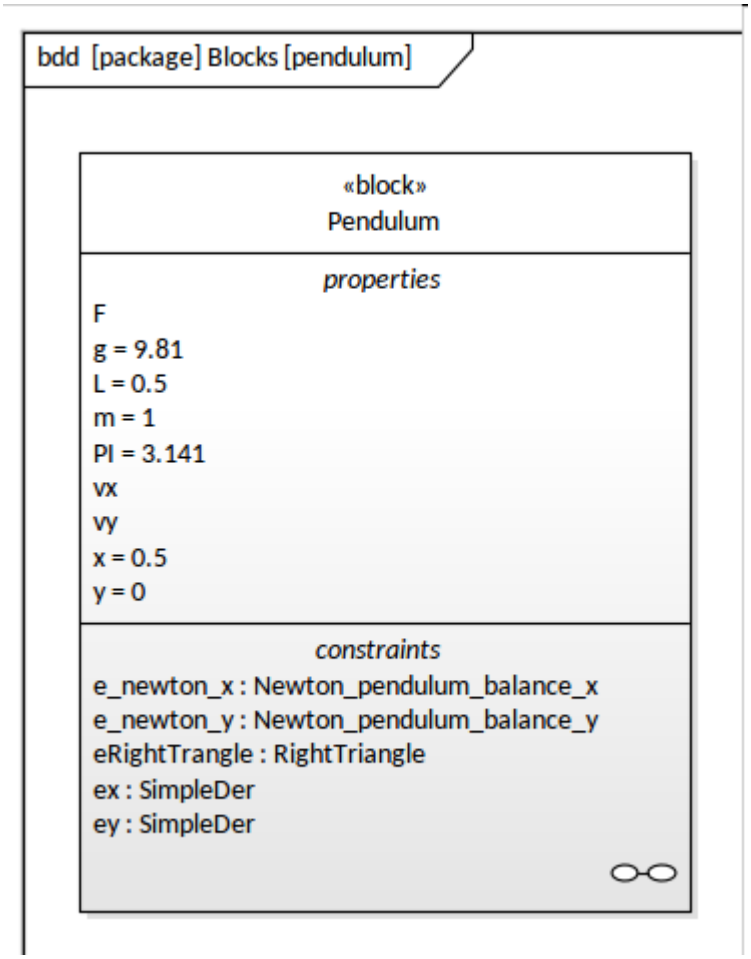
在接下来的部分我们将学习如何创建 SysML工件和配置为 <<仿真>>。

创建仿真元素

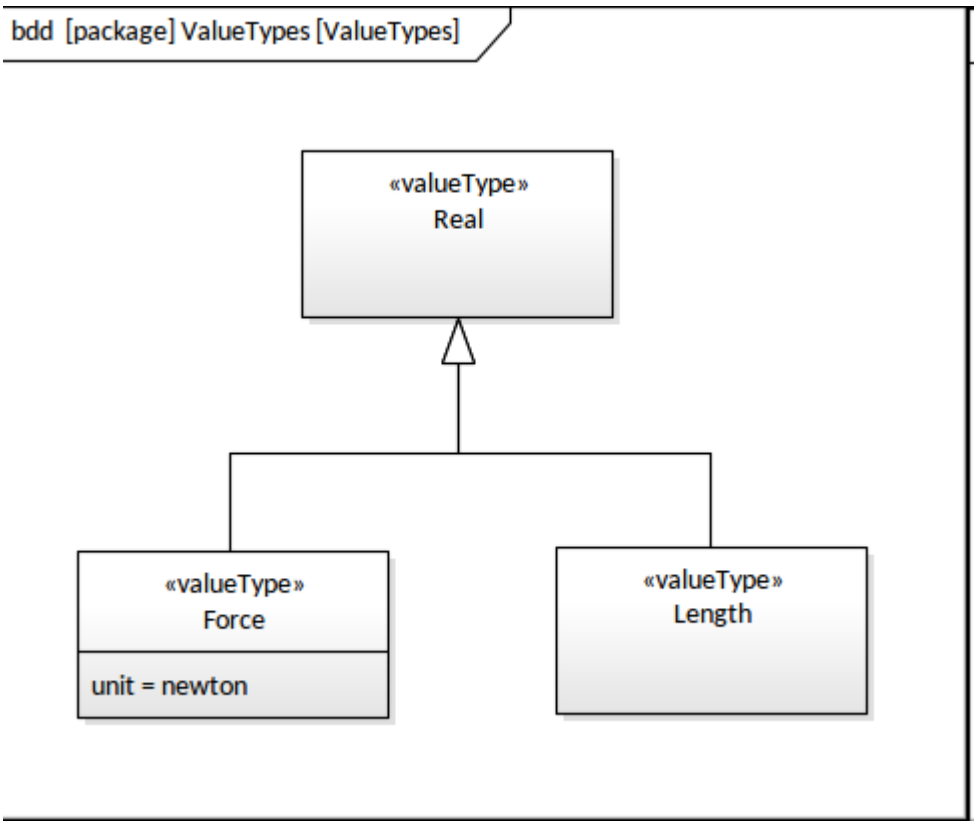
准备模拟的主要工作是定义模型元素，使用具有适当精度的运行以允许 OpenModelica 平台运行模拟。定义模型的方法有很多种，在本指南中，我们将聚焦介绍最稳健、最灵活的方法，因为大多数实践工程师和团队都会使用这种方法。

在这一点上，值得关注模拟主题，以确保我们了解它与模型其他部分的相关性和联系，以及模拟元素如何连接到其他模型元素，例如需求、测试案例等。通常，模拟被用作调查某些网络物理问题的一种方式，而无需构建耗时且通常昂贵的物理模型。模拟可以是问题分析、权衡分析、性能分析或许多其他调查的一部分。被定义为模拟一部分的块可以分配给行为元素并最终分配给需求。

要创建的第一个元素是块，它们是解决方案的基本结构元素。我们已经在前面的示例中学习了如何执行此操作；该图显示了许多具有元素的隔间，即属性。



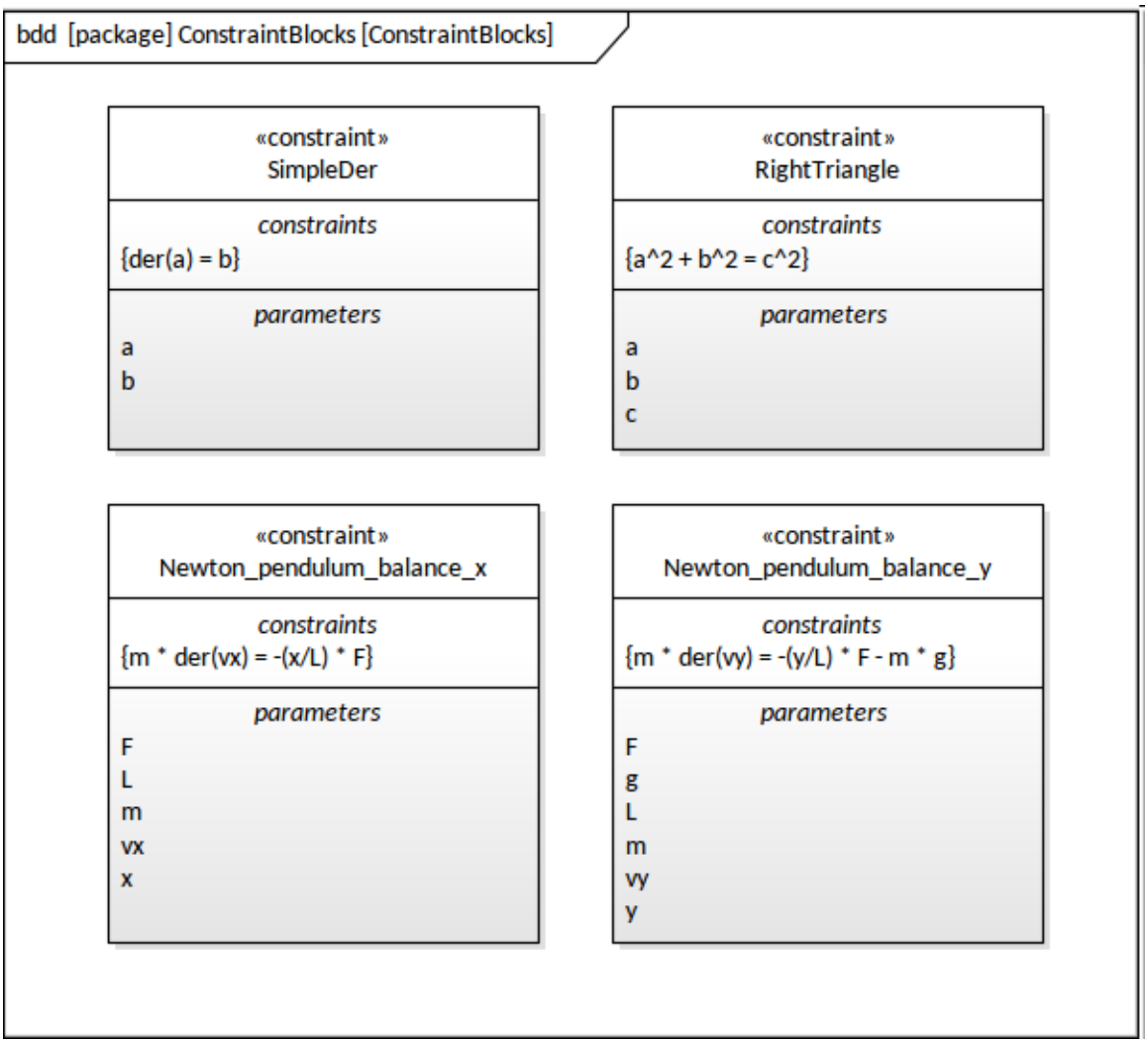
在准备定义定义摆系统行为的方程时，我们将定义确保模型精确所需的值类型，并检查仿真参数是否正确指定。这是使用块定义图 (图表) 来完成的，使用的是图形工具箱工具中提供的值类型元素。



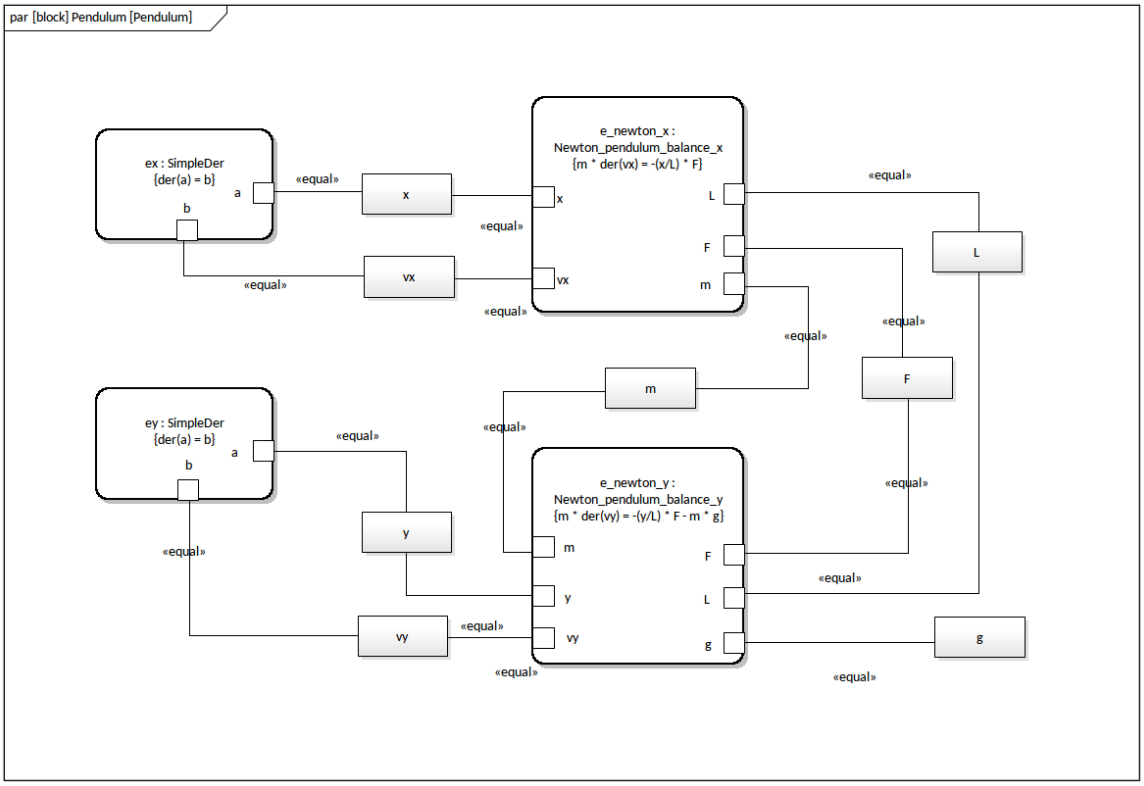
下一步是使用 ConstraintBlocks 定义约束。这些元素是我们将在其中定义 Modelica 方程的地方，这些方程将控制被建模系统的行为——在我们的例子中，是一个钟摆。如本指南前面所述，ConstraintBlocks 是在块定义图上定义的，并定义了一系列参数和一个用 Modelica 编写的方程来表达这些参数的约束。例如，约束摆的垂直方向的方程写成：

m * der(vy) = -(y/ L) * F - m * g

请注意 Modelica 关键字，例如表示一阶导数的 "der"。 L是摆的长度（参数），g 是重力加速度（常数），m 是摆的质量（参数），x 和 y 是二维空间中的坐标，F 是力.请注意，Modelica 使用：



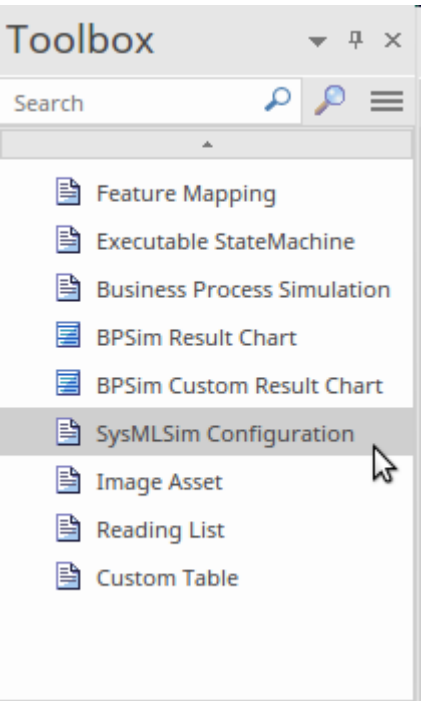
下一步是创建将方程绑定在一起的参数图。如指南前面所述，这些图是一种特殊类型的内部块图，包含称为 ConstraintProperties 的 ConstraintBlock 实例，这些实例公开了它们的参数，这些参数由连接器绑定到其他 ConstraintProperties 上的参数。



指定配置工件

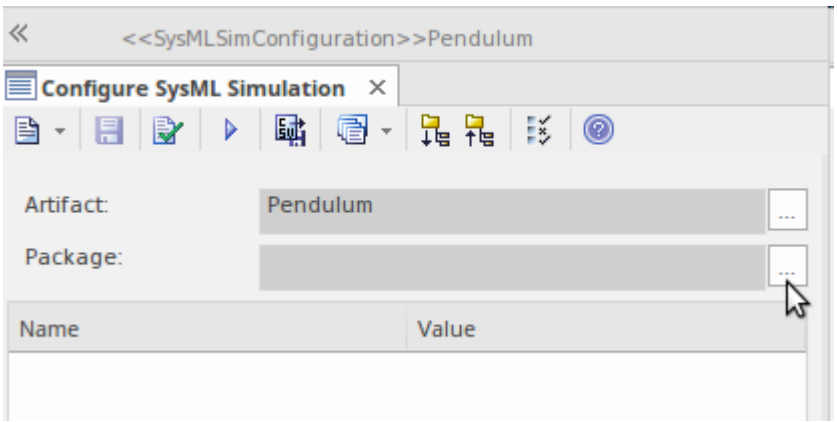
仿真元素是将模型元素绑定到工件平台的元素。在 SysML 中以 Blocks、ConstraintBlocks 的形式表示的模型元素及其在参数图上绑定在一起的相关 ConstraintProperties 出现在仿真窗口中，并且可以使用其他设置进行配置以驱动仿真。

第一步是设置此功能的元素配置，可从 "组件"中的 "工件"工具箱页面中创建此屏幕截图所示。

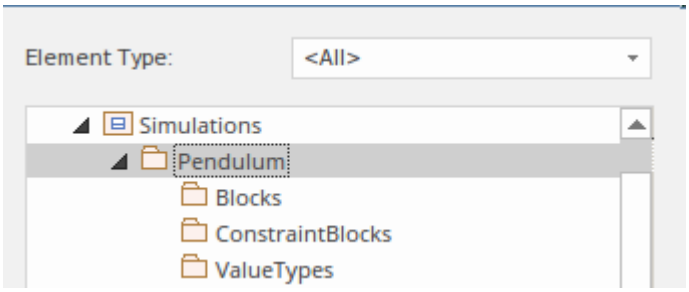


Enterprise Architect在填充配置SysML仿真窗口方面完成了大部分繁重的工作。工程师只需要选择包的 "配置"面

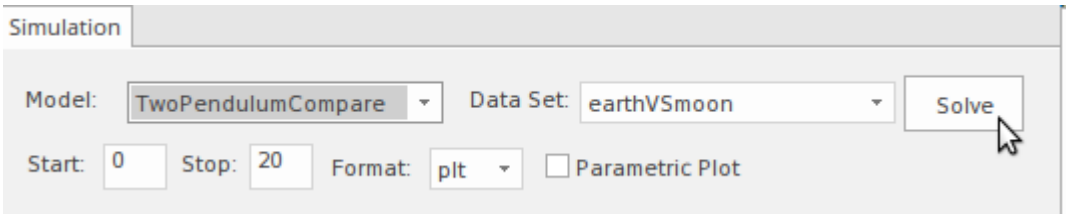
板上的仿真，就可以工件了。



Enterprise Architect将显示包选择窗口，一旦选择包， 配置"面板将填充模型中的块、约束块和值类型。从这一点开始，可以为各种参数输入值，或者可以定义数据集。可以输入仿真配置参数，仿真就可以运行了。



一旦为模拟输入了数据，包括开始和停止值以及输出格式，仿真可以通过选择 求解"按钮来运行模拟，如此屏幕插图中所示。



示例SysML模型

与任何语言一样，系统建模语言 (SysML) 必须学习。我们没有足够的幸运能够在家中听到我们的父母说这种家，但许多读者可能已经熟悉这种语言，原因有很多，包括：

- 已被同事或合作伙伴在项目中使用
- 它已作为大学课程的一部分教授
- 您参加过培训或阅读或查看过文档
- 您已经休假 6 周，并从头到尾阅读了规范

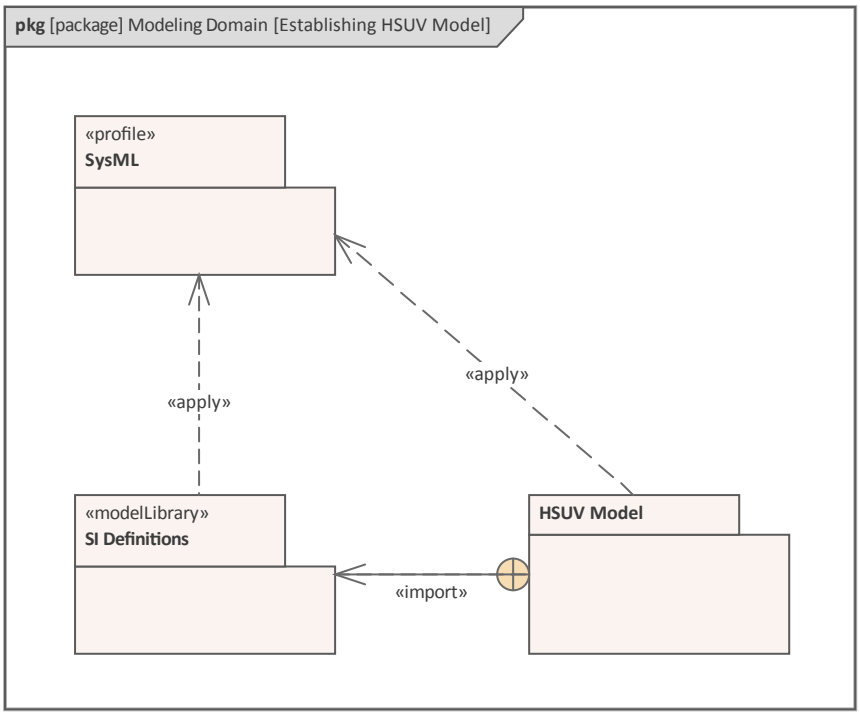
如果最后一个是正确的，您可能对该语言有深入的了解，但可能需要一些休闲假期，因为它是一个相当密集的文档，您还需要深入了解Unified Modeling Language规范。更多的读者很可能很少或根本没有接触过该语言，第一个示例旨在让您快速、高级地了解在Enterprise Architect中工作时可以预期到的基于模型的模型使用 SysML 的系统工程项目。它基于规范中的示例问题附件中出现的混合动力汽车的示例

包Overview (Structure of the Sample模型)

包图演示了一种可视化存储库内容的方法；当在Enterprise Architect的浏览器窗口中查看图表内容时，可以导航结构。在包图中还可以看到重要的结构和命名空间关系，这些有助于阐明存储库中元素组之间的重要高级关系。

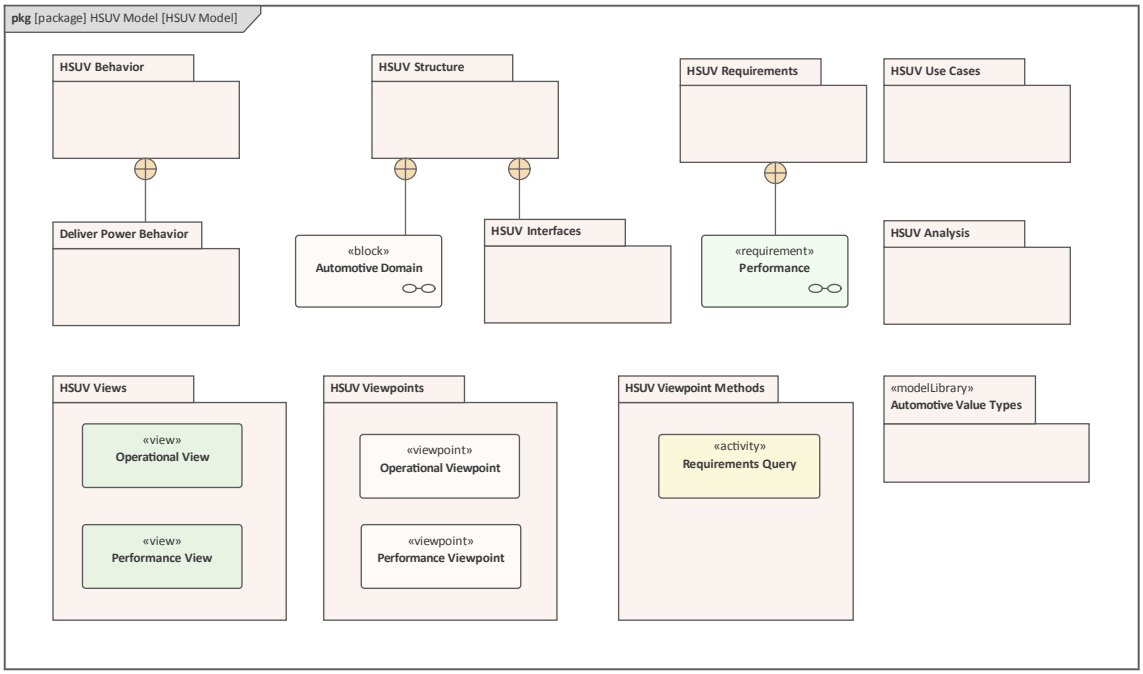
包图-应用SysML配置文件

如图所示，HSUV模型是一个代表用户模型的包。将 SysML配置文件应用于此包，以便包含配置文件中的构造型。 HSUVModel 可能还需要模型库，例如 SI Units Types模型库。模型库按指示导入到用户模型中。

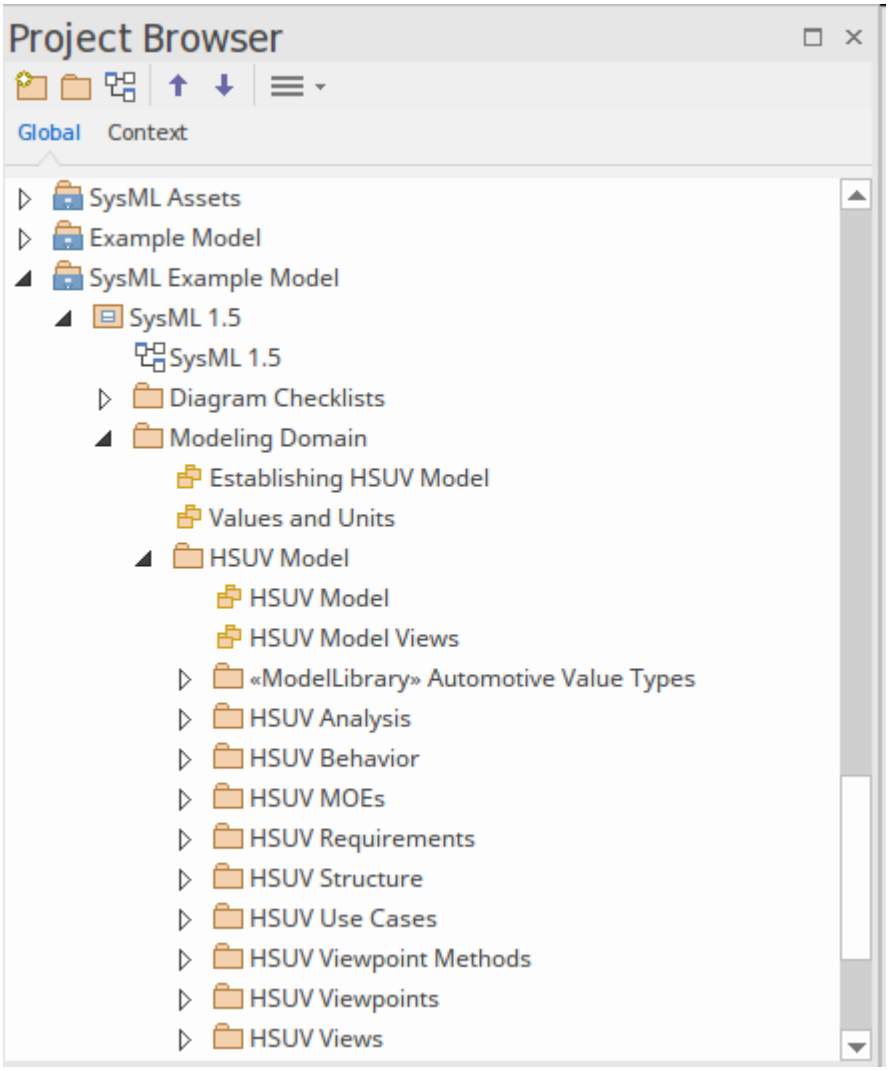


包图-展示包模型结构

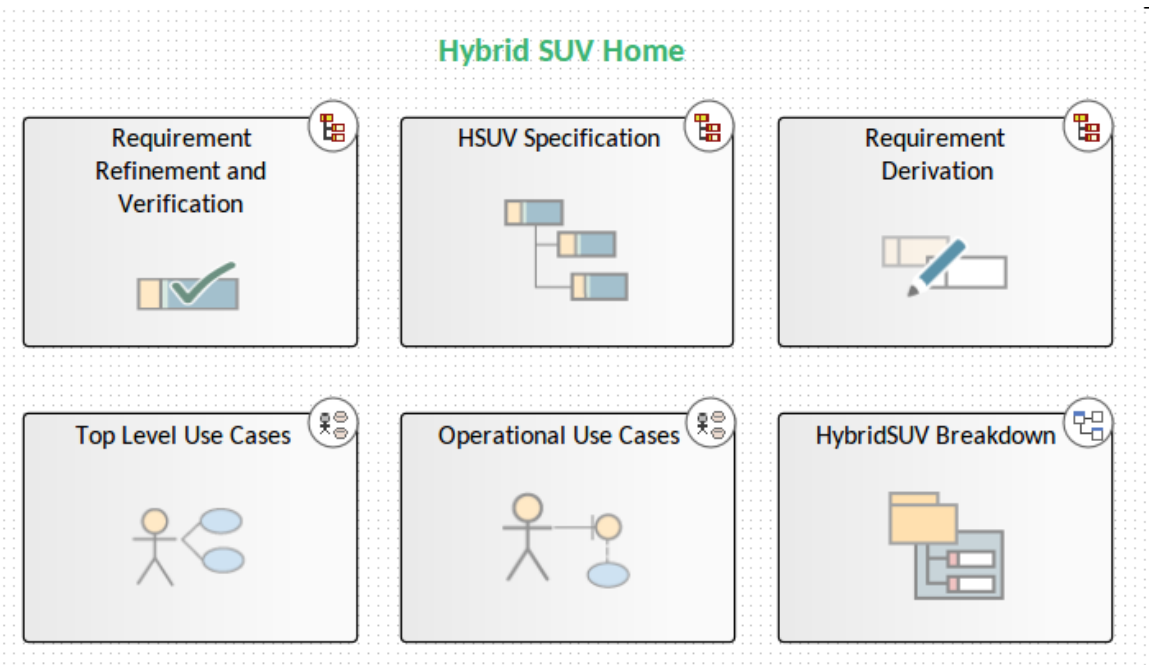
该软件包图显示了用于定义包系统的模型结构。该图提供了一种有用的方式来查看用于构建存储库并定义 HybridSUV 系统的结构、行为和要求的高级容器 (包)。



操作视图和性能视图之间的关系以及用户模型的其余部分使用«import»关系明确表示。图中出现的包已定义，也可以使用浏览器窗口在分层视图中可视化。



虽然浏览器窗口为浏览存储库提供了一种重要的机制，但还有许多其他视图，**包括** - 在这种情况下 - 一个图表。Enterprise Architect还提供了一种方便的方式来创建用户定义的图表，可以作为导航存储库的替代方式。这种机制允许系统工程师和其他人创建任意数量的导航单元以提供对模型的受众定制访问，从而使用户无需知道或了解如何遍历模型。



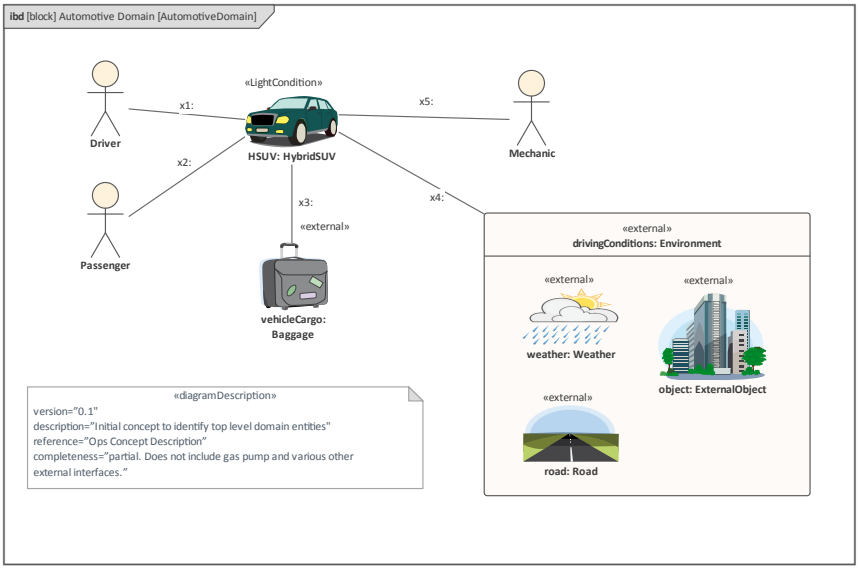
这些在通过网络浏览器查看模型时特别有用。

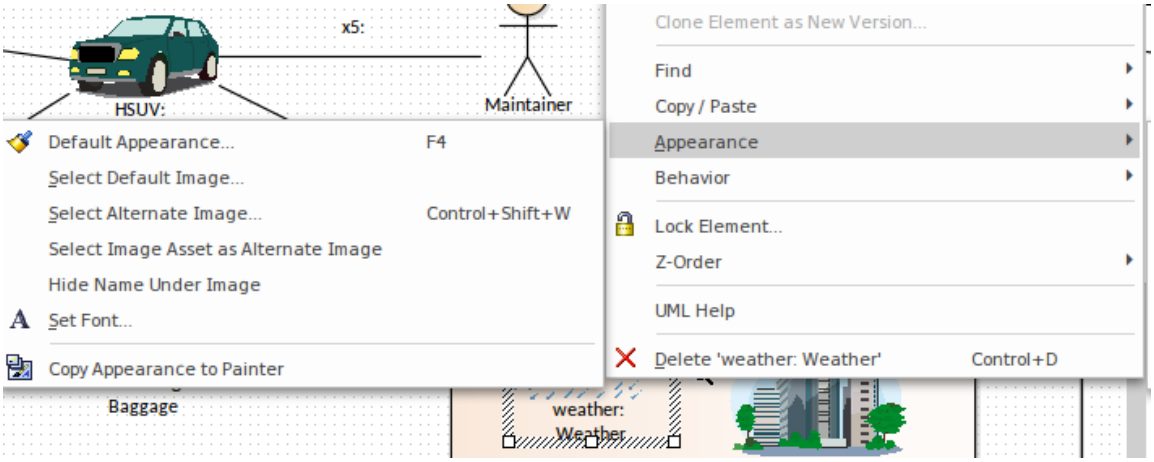
设置上下文（边界和使用案例）

系统的时间对于所有利益相关者了解其上下文中的系统至关重要。用例图是 SysML 工具包中最简单但最具描述性的图之一。它的强大之处在于它将驻留在系统之外的实体（参与者）与他们希望从系统中获得的好处（使用案例）联系起来，而无需阐明系统将如何交付价值。使用案例可以在描述性级别上编写，但如果需要更多细节，可以使用Enterprise Architect的场景构建器来指定每个场景的步骤，在一个工具中，无需记录使用行为。作为活动图表和用例文档也可以从该工具中自动生成。

运营域模型- 设置上下文

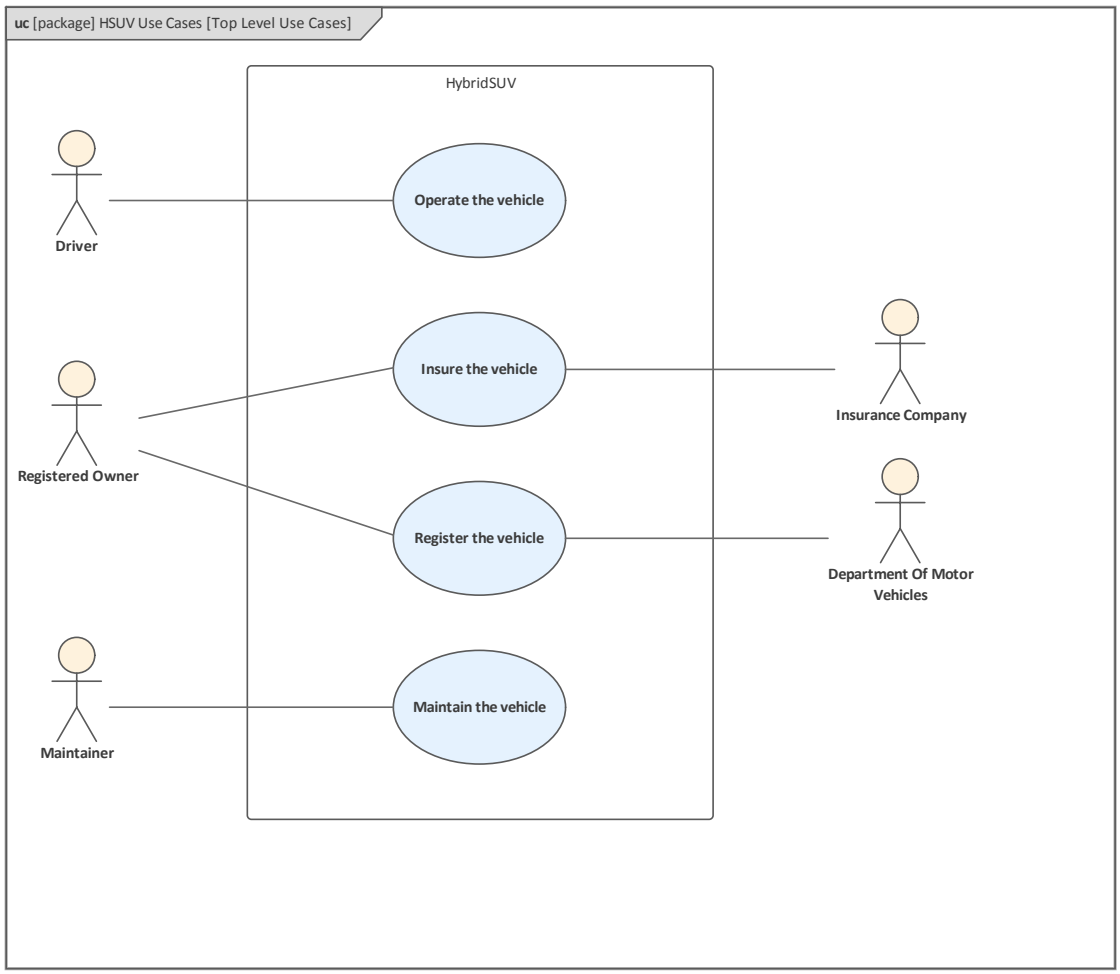
此内部块图描绘了图表类型的用户定义用法，它上下文了系统在其环境的时间。 «system» 和 «external» 构造型是用户定义的，未在 SysML 中指定，但有助于系统工程师描述与其环境相关的感兴趣系统。



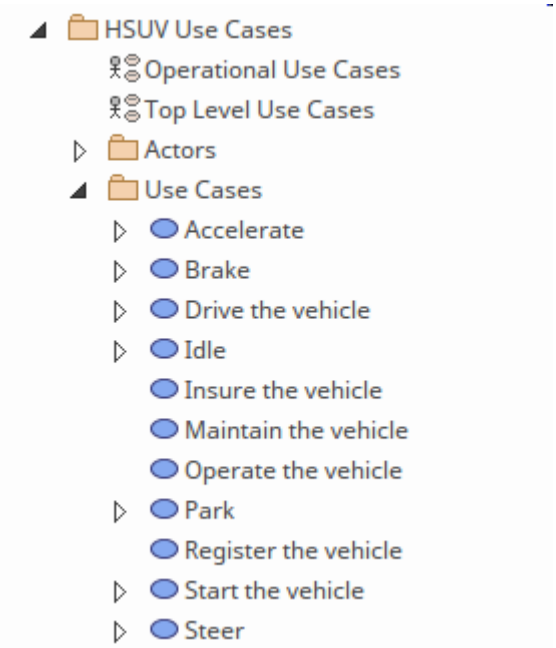

x5:
HSUV:
Maintainer
Default Appearance... F4
Select Default Image...
Select Alternate Image... Control+Shift+W
Select Image Asset as Alternate Image
Hide Name Under Image
Set Font...
Copy Appearance to Painter
Clone Element as New Version...
Find
Copy / Paste
Appearance
Behavior
Lock Element...
Z-Order
UML Help
Delete 'weather: Weather' Control+D
Baggage
weather:
Weather

用例图表- 顶级使用案例

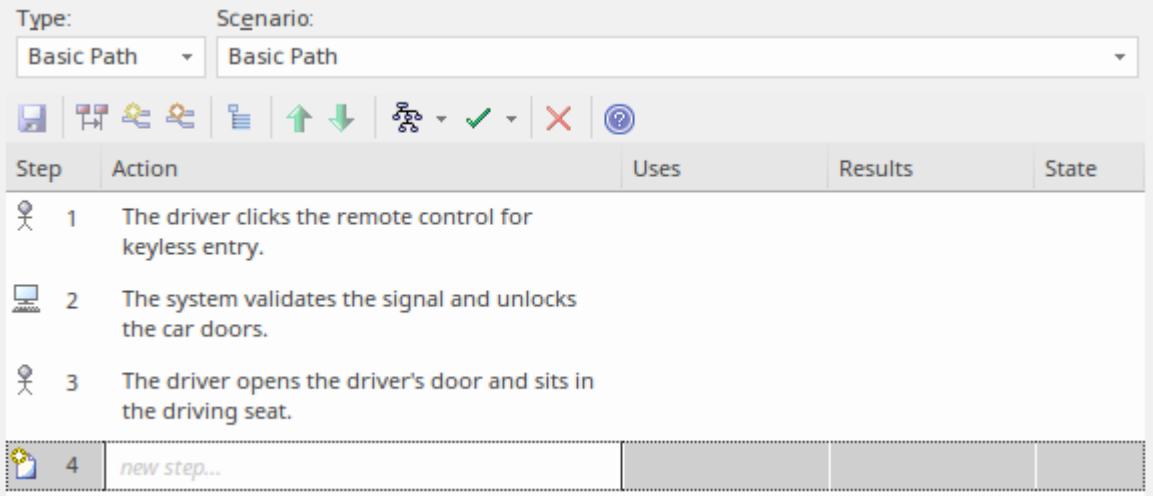
用例图将 HybridSUV 描述为一个由多个外部角色使用的系统。该图有助于设置与系统交互或从系统获得上下文的所有角色（人类和其他系统）的时间。用例图虽然看起来很简单，但提供了一种机制来确保定义和理解所有潜在的系统交互。系统在图中由一个边界元素表示，它充当使用案例的容器，Actor 位于该在边界。在此图中，除了驾驶员之外，还有许多将与拥有着系统交互的外部角色，包括注册拥有者、维护者、保险公司和机动车辆部。



使用案例出现在浏览器窗口中，可以方便地分组为演员和使用案例。然后可以定义任意数量的使用用例图，允许系统工程师可视化使用案例。



Enterprise Architect还为系统工程师提供了许多有用且独特的工具来有效地描述使用，并定义代表参与者之间细节的步骤，以帮助参与者和使用系统。一旦定义了这些，该工具就可以直接从模型中自动生成行为图。

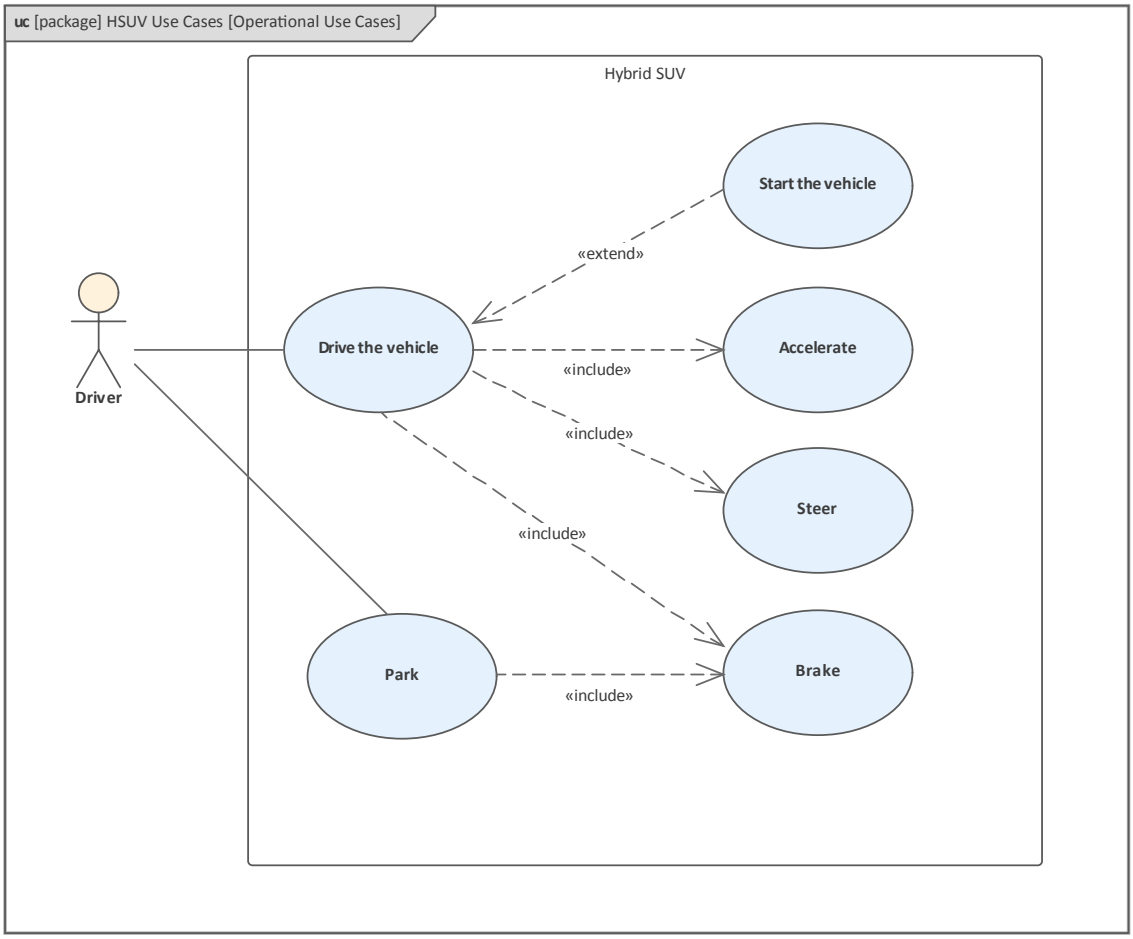


一旦将步骤生成为模型元素，**就可以添加可追溯性**：

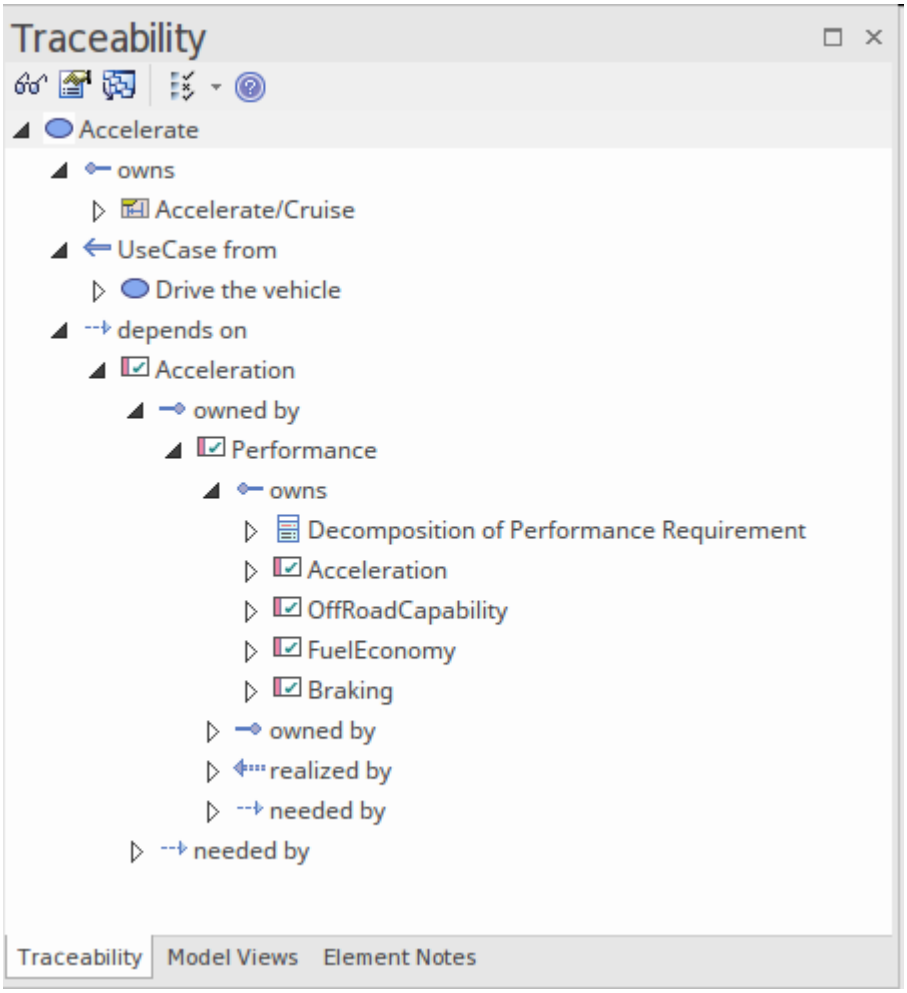
- 参与者模型步骤可以追溯到人机接口
- 系统步骤可以追溯到部件模型。

用例图表- 操作性使用案例

下图中描述了与 操作车辆”相关的目标级使用案例。这些使用案例有助于充实与驾驶和停车相关的特定类型的目标。该图侧重于作为中心参与者的车辆驾驶员。更高级别的使用案例，例如车辆的维护、注册和保险，在一组单独的上下文使用案例中定义。

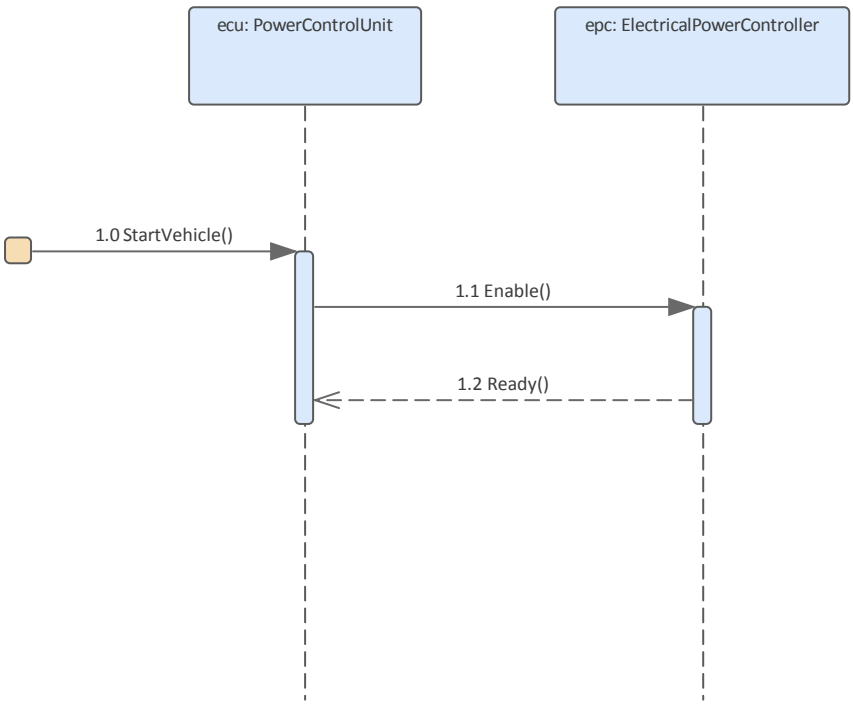


除了上一节中描述的用于阐明场景及其详细步骤的广泛特征外，该工具还提供了许多其他特征，可在使用案例时协助系统工程师。这些特征中最有用的一个是可性可追溯性窗口，它提供了一个令人信服的可视化，显示给定的用例与什么相关联，以及相关联的元素与什么相关。当在浏览器窗口或图表中选择不同的使用案例（或任何其他元素）时，窗口会刷新以显示所选元素的连接。


Traceability
Accelerate
owns
Accelerate/Cruise
UseCase from
Drive the vehicle
depends on
Acceleration
owned by
Performance
owns
Decomposition of Performance Requirement
Acceleration
OffRoadCapability
FuelEconomy
Braking
owned by
realized by
needed by
needed by
Traceability
Model Views
Element Notes

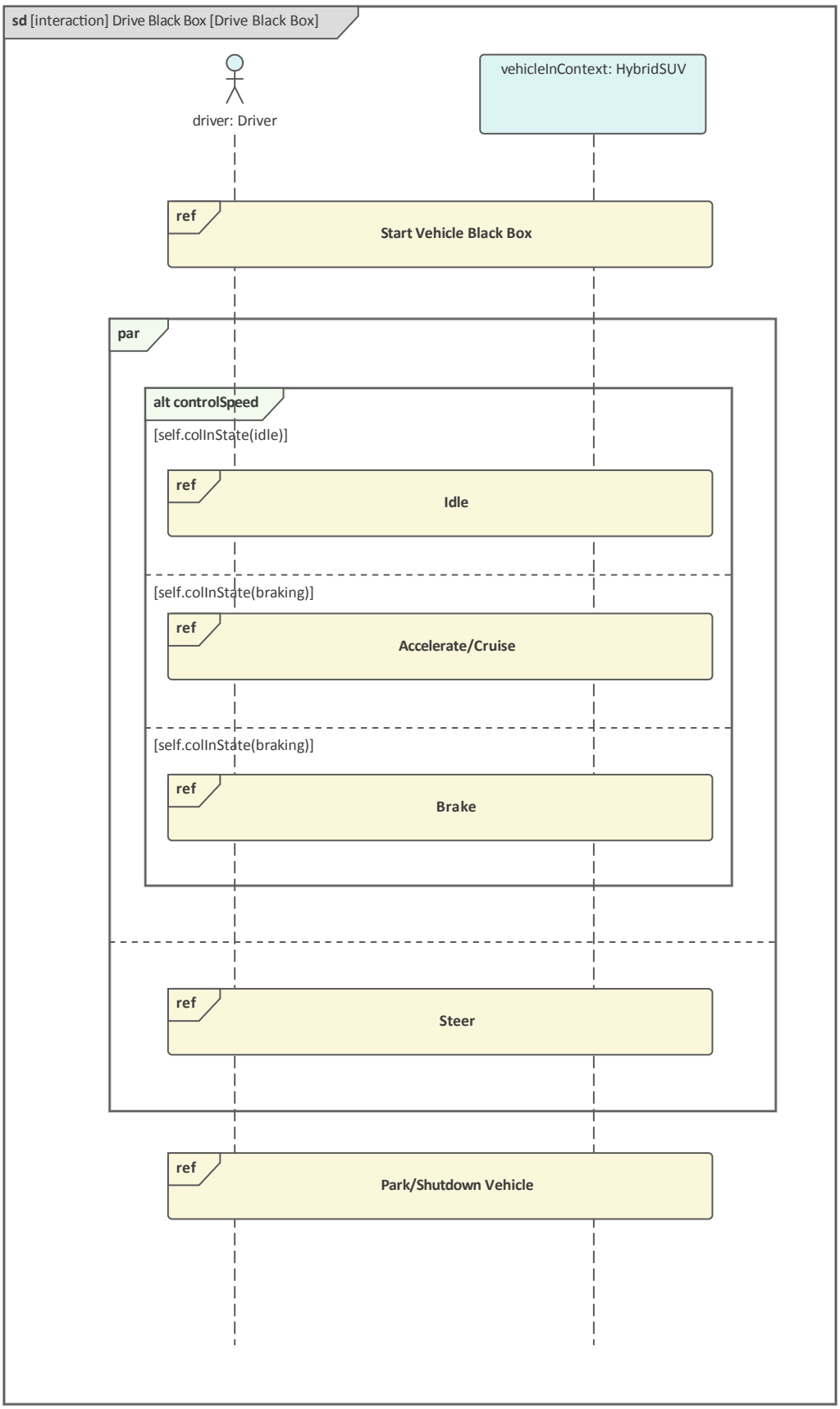
细化行为（序列和状态机图表）

这些主题和部分展示了如何使用序列和状态机图来表示和阐述行为。

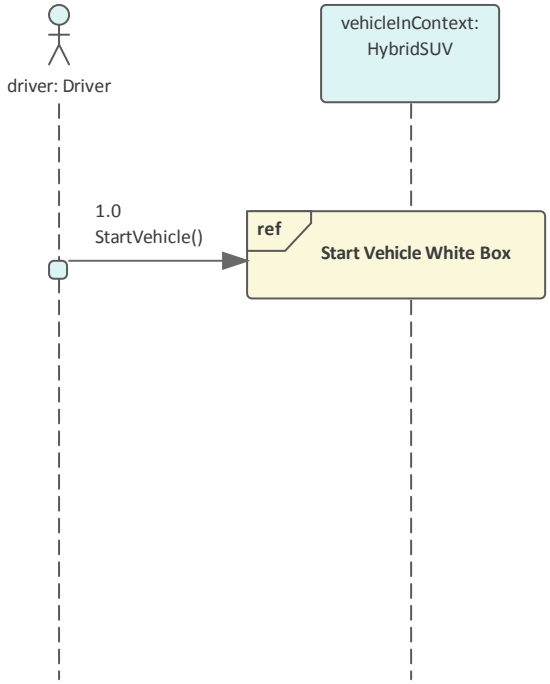


序列-图表Black Box

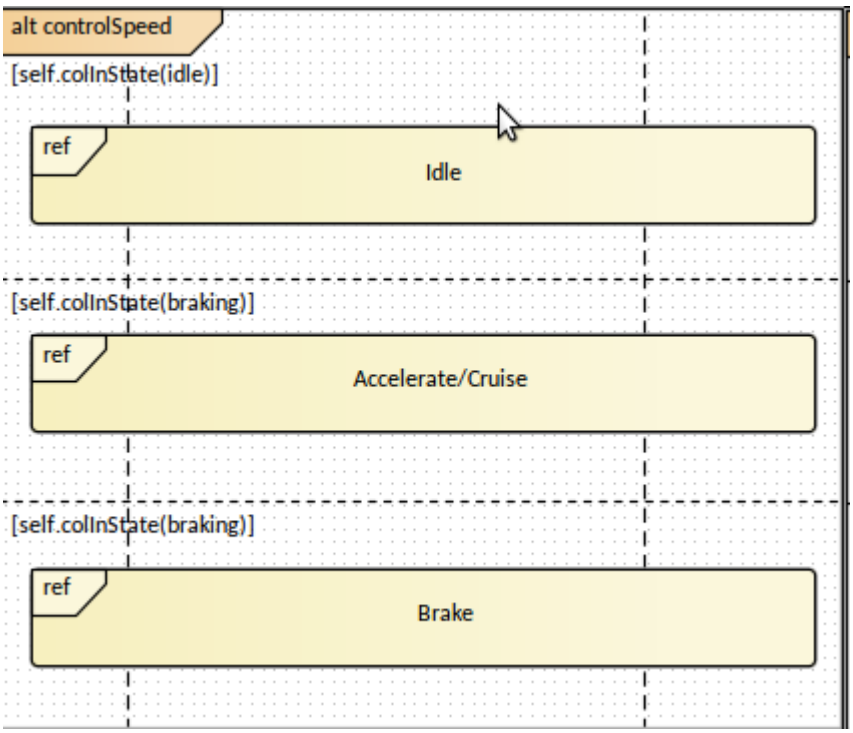
序列图是一种交互图，描述了司机在特定上下文与车辆的交互方式。按照惯例，参与的元素在图表的水平访问中从左到右列出，时间在图表中垂直进行。从对象发出的虚线称为生命线，代表元素存在的时间。



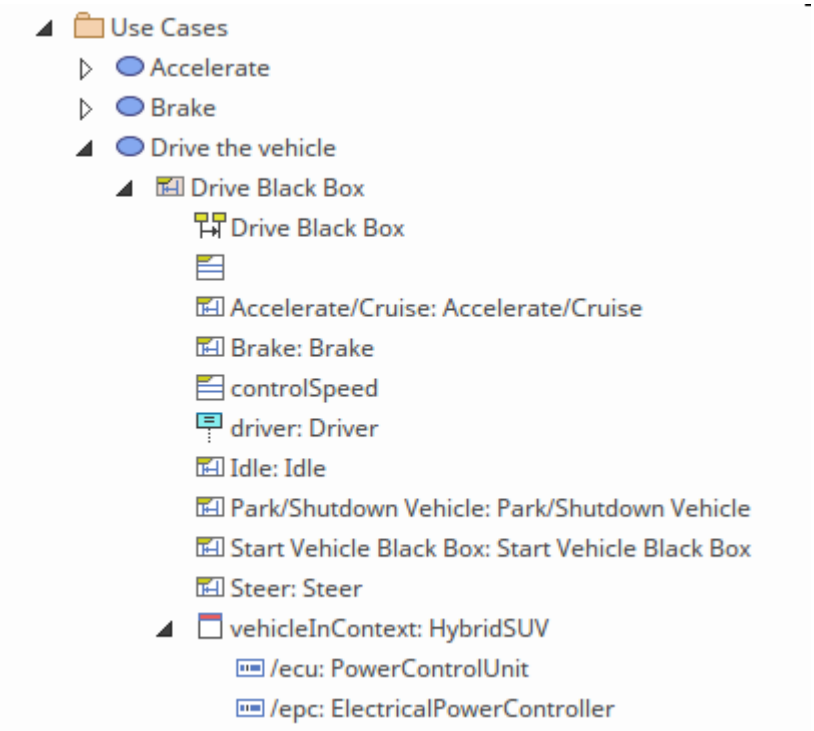
该图使用了一种称为元素的参考，它充当另一个序列图图表占位符。 Enterprise Architect可以方便地通过双击图中的"ref"元素来打开这些图。



该图使用 Parallel (par) Combined Fragment 来指定控件Speed 和 Steering 交互同时发生（并行）。该图还使用带有选择（alt）名称的组合片段，它指定了驾驶员可以控制速度的多种（替代）方式。

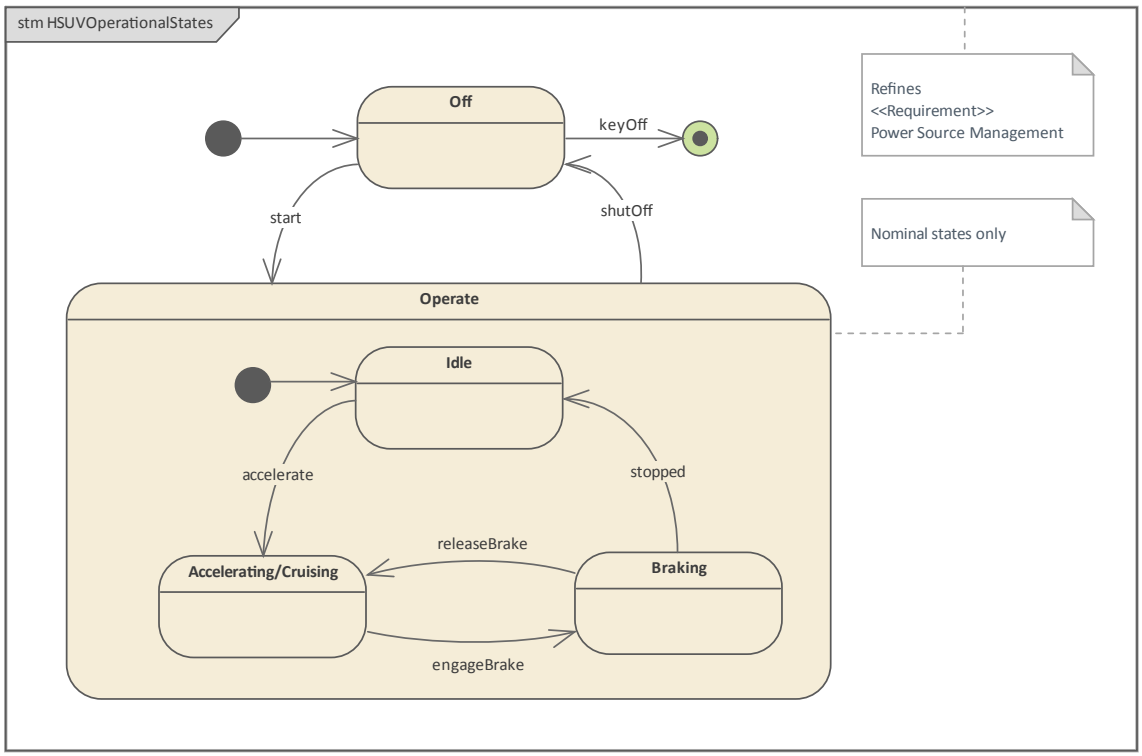


图表Enterprise Architect允许将序列图定义为 驾驶车辆”的子项，示例使用例者可以轻松访问图表并在驾驶员想要实现的车辆目标的上下文查看它.

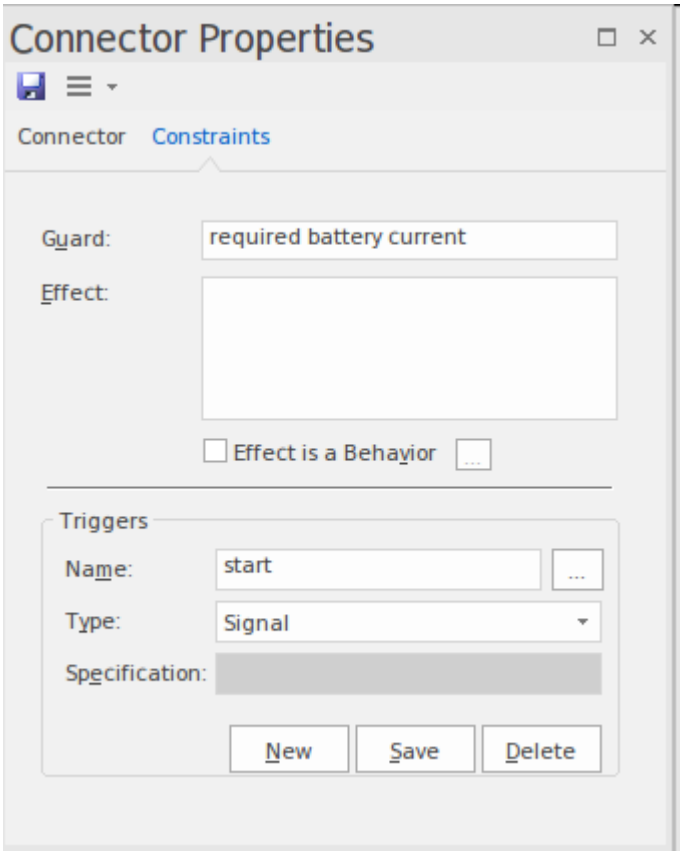

Use Cases
Accelerate
Brake
Drive the vehicle
Drive Black Box
Drive Black Box
Accelerate/Cruise: Accelerate/Cruise
Brake: Brake
controlSpeed
driver: Driver
Idle: Idle
Park/Shutdown Vehicle: Park/Shutdown Vehicle
Start Vehicle Black Box: Start Vehicle Black Box
Steer: Steer
vehicleInContext: HybridSUV
/ecu: PowerControlUnit
/epc: ElectricalPowerController

状态机图表-图表运行状态

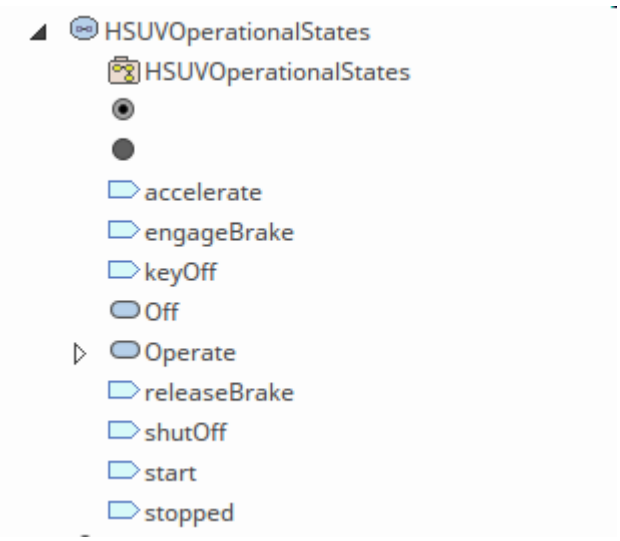
此状态机图用于描述混合动力 SUV 在其生命周期的指定部分中表现出的离散状态。取决于观察者A视角，系统或系统的一部分可以具有广泛的状态，因此建模者必须始终指定用于图表的视角或视图。然后，该图阐明了实体指定生命周期内的重要和相关条件。



序列开始是使车辆从Off状态过渡到Operate(on)状态的触发器。 Enterprise Architect允许在细节中定义这些转换。

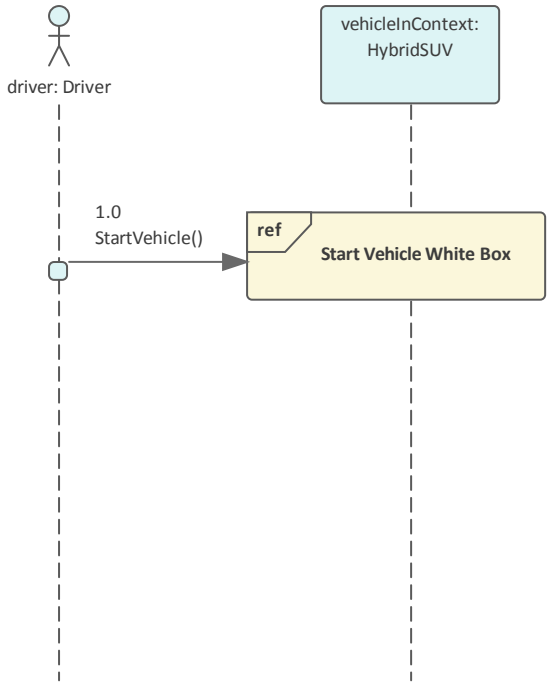


状态出现在浏览器窗口中，并在状态机节点下方便地分组在一起。

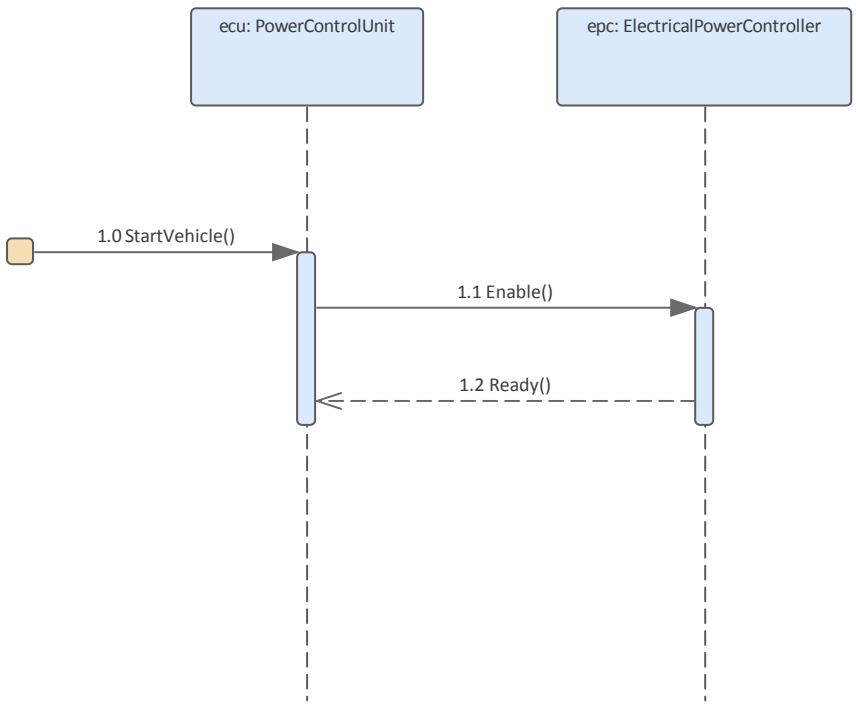


序列图表开始车辆黑盒和白盒

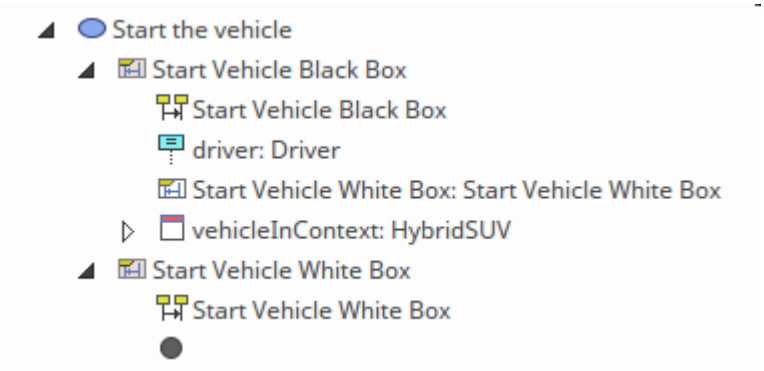
该序列图描述了驾驶员启动车辆的交互。它使用指示使用（**交互**）来制作参考到序列元素图。该图本质上很简单，但这些图可以通过许多其他元素和连接器来扩充，这些元素和连接器详细说明了其他更复杂交互的细节。



Enterprise Architect提供了一种方便的机制，允许建模者点击到引用的图表，在这种情况下是开始White Box 图表。



Enterprise Architect中的元素可以出现在多个图表中，允许在模型中建立富有表现力的叙述，并为建模者提供一种机制来创建同一元素的多个视图。图表上的元素可以位于浏览器窗口中，显示它们与模型其他部分的结构关系。在这种情况下，开始交互的 Black 和 White Box 视图都位于开始的子项中，用例使它们之间的关联变得容易。



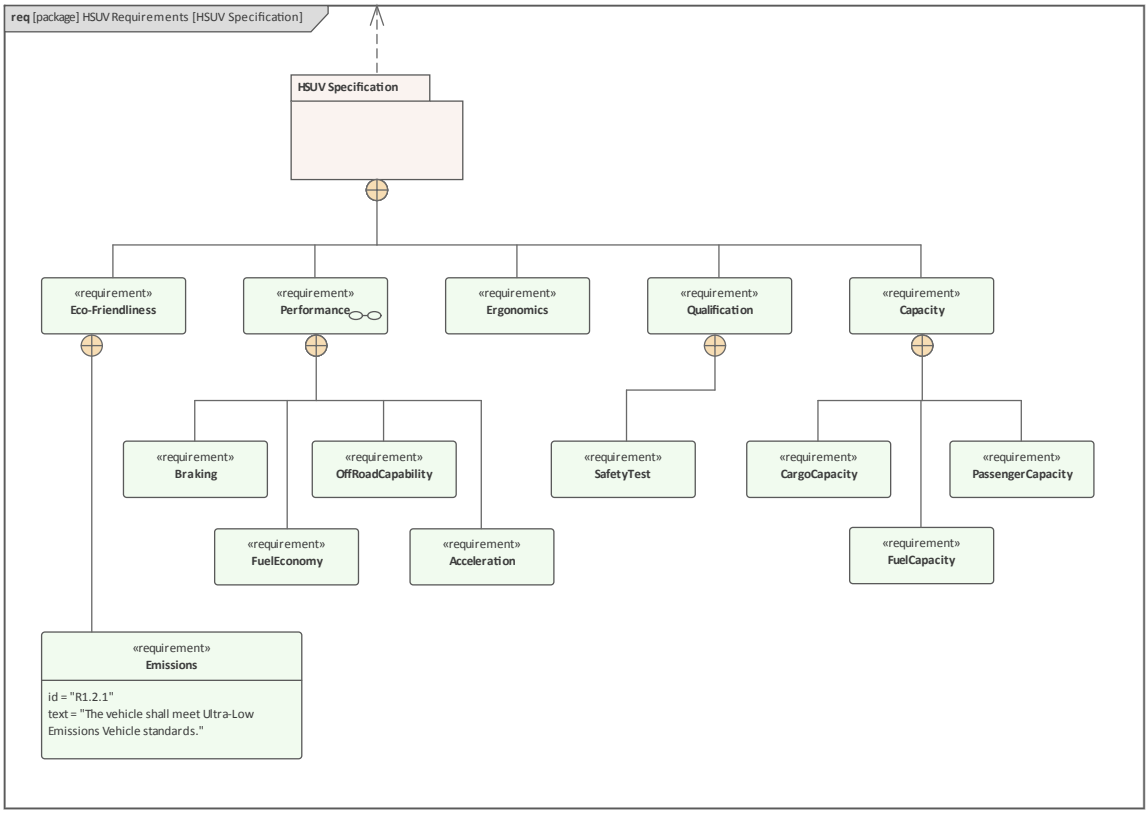
建立需求（需求图表和表）

这部分模型的目的是演示需求的可视化建模。车辆系统规范包含许多已在Enterprise Architect中重新创建的基于文本的要求。已经对一系列要求进行了建模，包括车辆通过排放标准的要求，为了说明的目的，该要求进行了扩展。

Enterprise Architect为创建、开发、分析、管理和测试需求提供了广泛的工具。它还集成了DOORS需求管理工具。

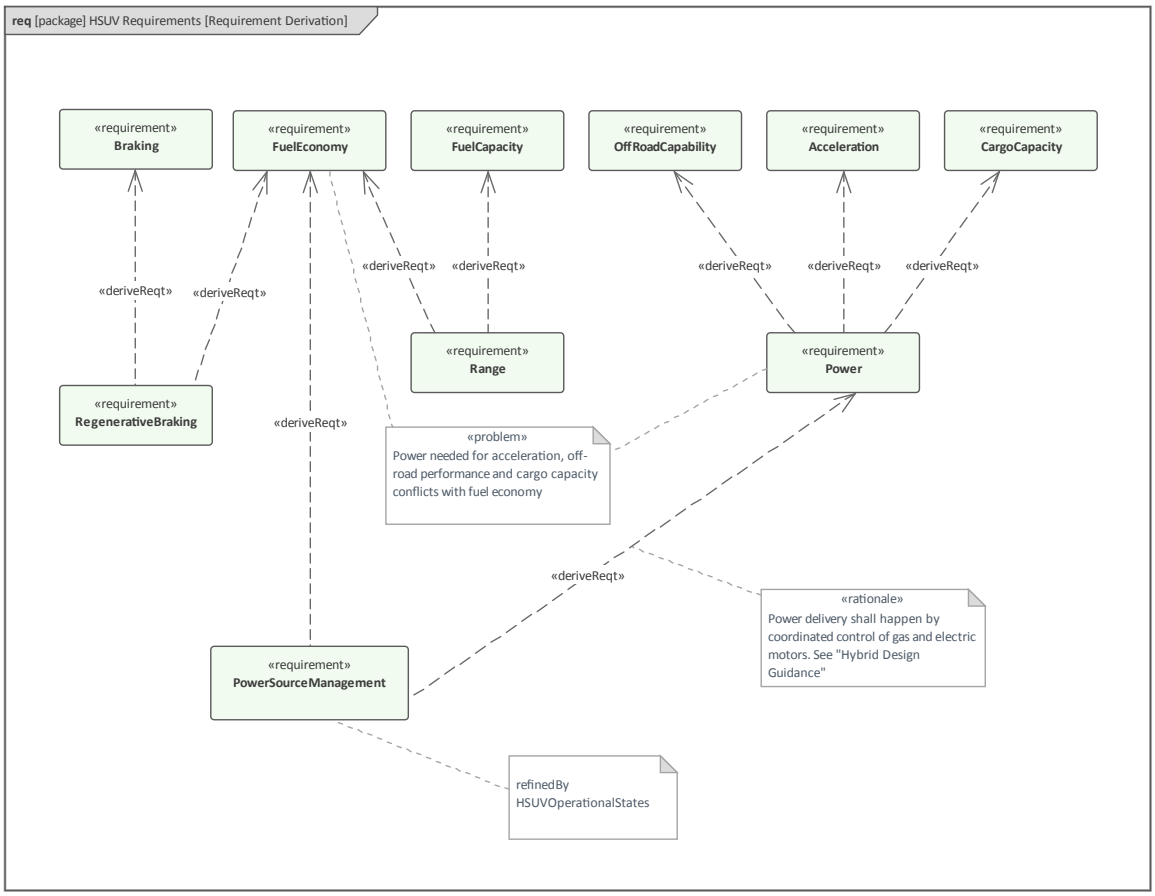
需求图表-图表需求层次结构

此图演示了使用包含（交叉影线）连接器显示父子关系的层次结构中的需求可视化。更高级别的需求充当一种分组或包含系统，涵盖了一系列高级问题，这些问题被分解为较低级别且可能可衡量的陈述。



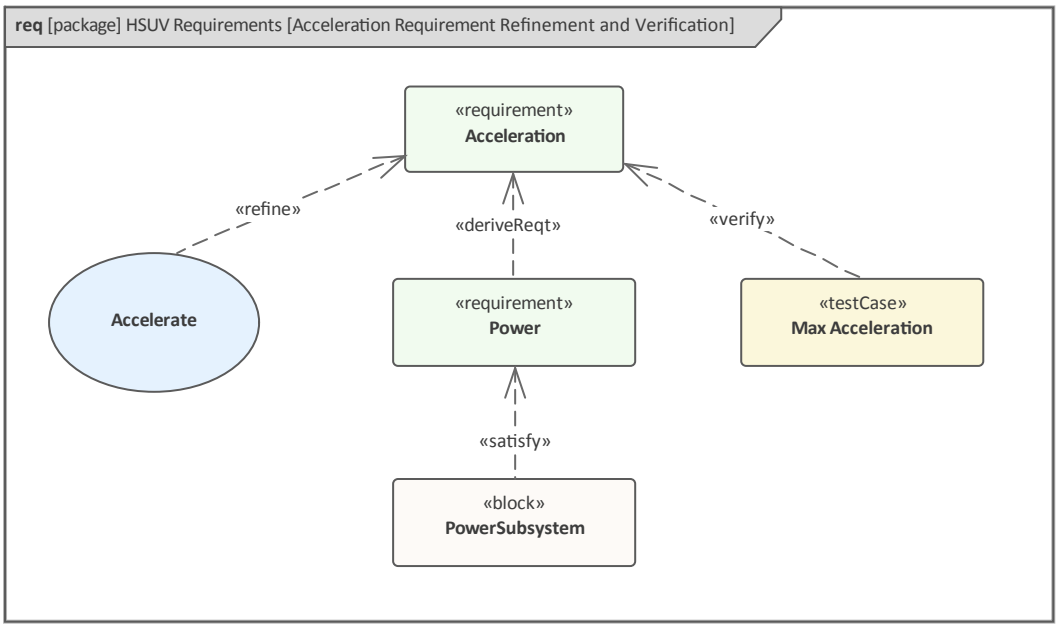
需求图表- 衍生需求

这个需求图显示了一些需求是从其他需求中衍生出来的。这种关系意味着已经进行了一些分析，并且派生的需求代表了可以满足原始需求意图的需求。例如，再生制动不是一项要求，但它源于对制动和燃油经济性的需求，如关系箭头末端的要求所示。



需求图表- Acceleration需求关系

该图演示了一些需求关系，这些关系通常会在完成一些分析和建模后出现在需求图上。重点元素是加速要求，该图显示了许多其他元素与此要求的关系。引入细化关系是作为将类似命名的用例与加速要求相关联的一种方式。我们有另一个派生的要求，随后由块满足。图中还显示A “Max Acceleration”测试用例，并通过测试用例关系与中心要求相关。该图还演示了可以添加除需求之外的元素，并有助于使图更具表现力。



库表-需求库表

这些示例演示了需求以表格形式显示的方式，以替代图表中的图形表示。对于更习惯于使用电子表格的一系列利益相关者来说，这是一种受欢迎的演示风格。第一个表列出了需求及其 ID 和文本声明。第二个表列出了参与派生关系的源和目标需求。

Enterprise Architect还提供了一系列可视化需求（**和其他元素）的工具和方法**，包括列表视图、看板、规范视图、甘特图、图表等。

req [requirement] Performance [Decomposition of Performance Requirement]

Decomposition of Performance Requirement

I..	NAME	TEXT
2	Performance	The Hybrid SUV shall have the braking, acceleration, and off-road capability of a typical SUV, but have dramatically b...
2.	Braking	The Hybrid SUV shall have the braking capability of a typical SUV.
2.	FuelEconomy	The Hybrid SUV shall have dramatically better fuel economy than a typical SUV.

Showing 1 - 3 of 10 items

Tree of Performance Requirements

DERIVED_ID	DERIVED_NAME	RELATION	ID	NAME
2.2	FuelEconomy	deriveReqt	d.3	Range
2.2	FuelEconomy	deriveReqt	d.2	PowerSourceManagement
2.2	FuelEconomy	deriveReqt	d.1	RegenerativeBraking
2.1	Braking	deriveReqt	d.1	RegenerativeBraking

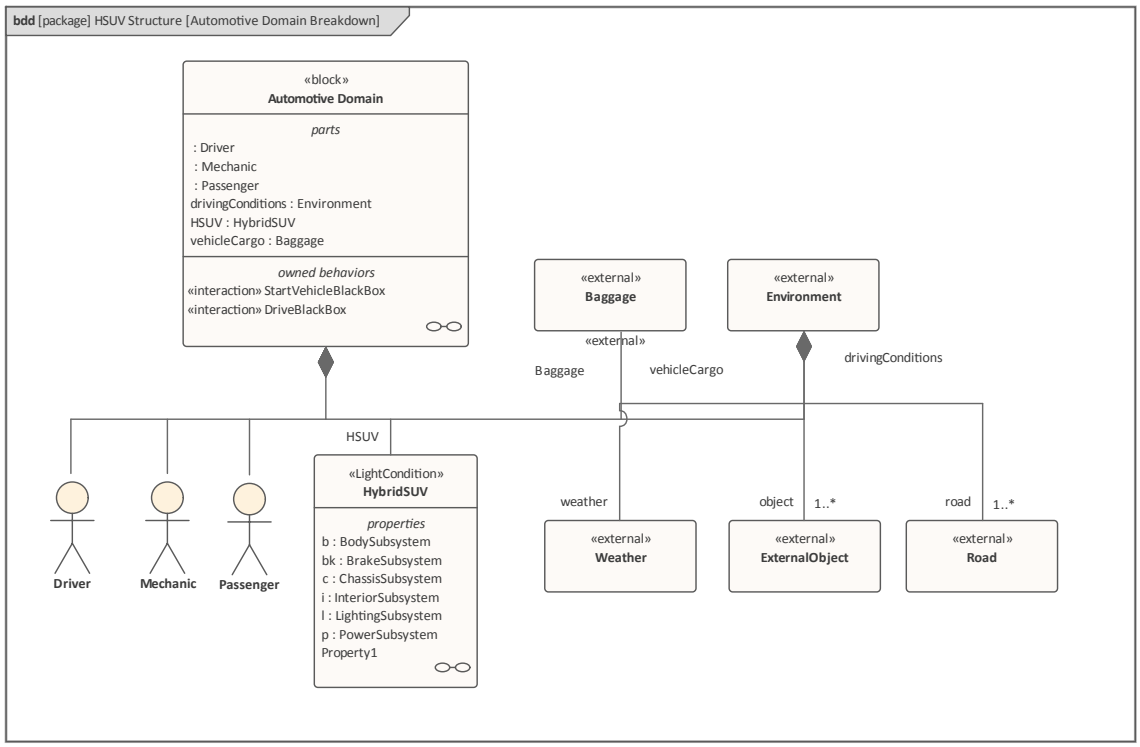
Showing 1 - 4 of 23 items

图表Down the图表（块定义图，内部块图）

块是结构的基本单元，可用于块定义和内部块图表来描述系统的结构方面。对于许多想要了解系统知识并了解其结构的工程师来说，块定义图通常是一个起点。块本身由结构组成，由特征表示，这包括本身由其他块键入的部件，例如车轮组件可能具有盘式卡钳部件。还有 Value属性，即具有数量的项目，这些项目代表物理和其他可测量的尺寸，例如汽车可能有重量和颜色，并且 0-100 公里的加速时间为 5 秒。还有一些交互点显示了一个块可以与其环境交互的点。

块定义图表-汽车领域

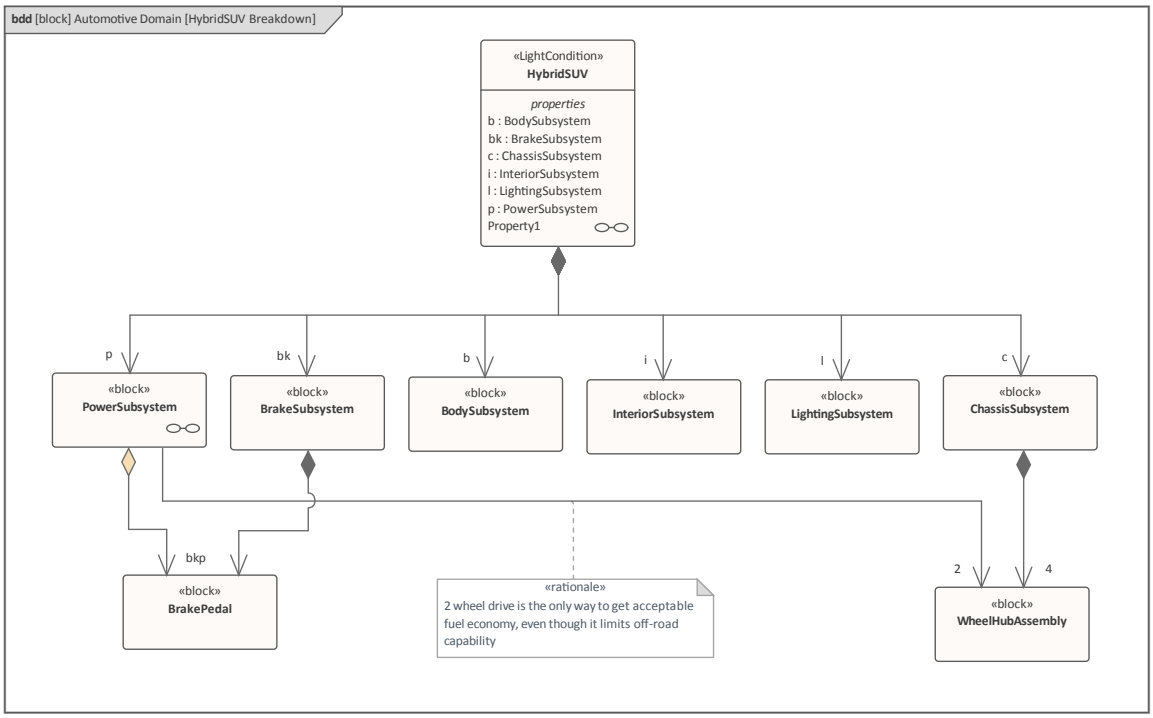
此图显示了使用块定义图来描述组成汽车域的部分。该域包括在执行角色时将与系统交互的人员，包括驾驶员、乘客和机械师，因此在修复制动子系统后执行测试时，可能会有许多特定的人员扮演驾驶员和机械师的角色。将他们的角色从机械师转变为驾驶员。



块定义图表-混合动力SUV

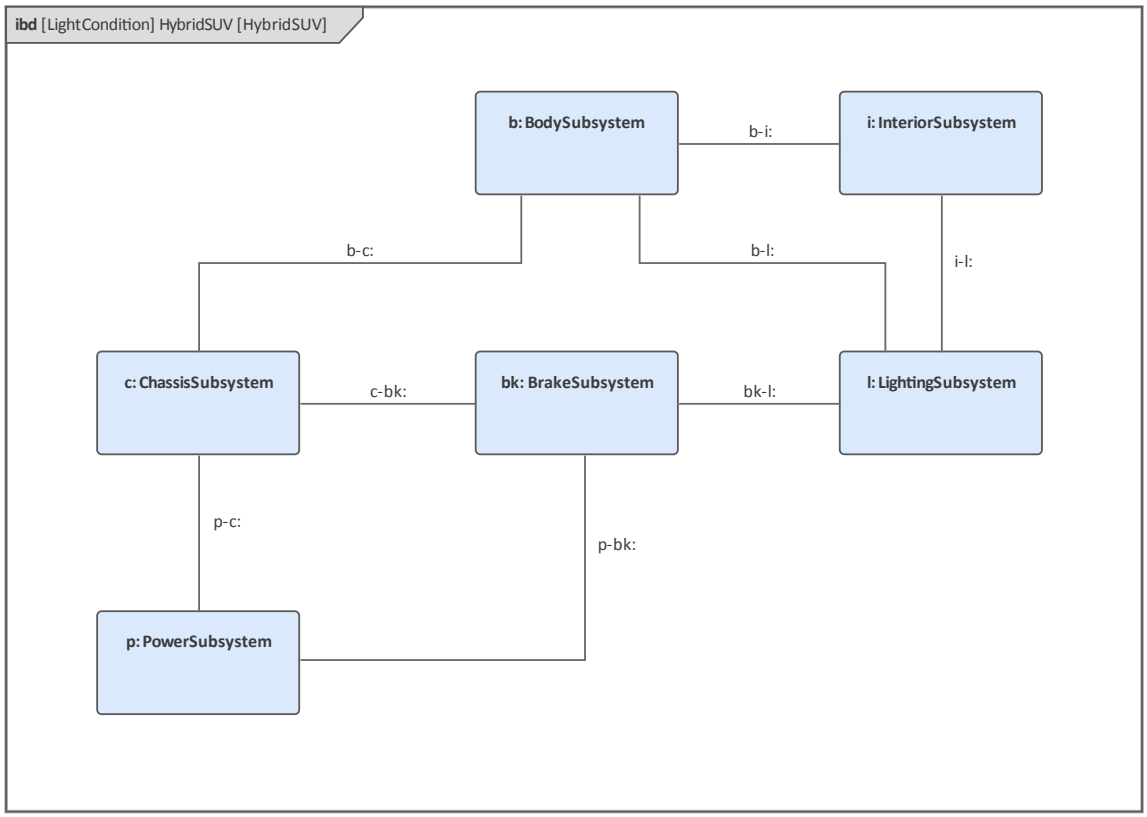
所有需要工程分析和设计的系统都将具有一定程度的复杂性，需要将系统分解为多个部分，以帮助降低复杂性并促进项目管理。第一级故障通常被称为子系统，对于混合动力 SUV 或任何其他汽车，子系统将包括诸如动力、制动、照明和底盘等模块。

这些子系统本身又会分解成许多组成部分，例如制动子系统可以分解成盘组件和液压部件。



内部块图表- 混合动力SUV

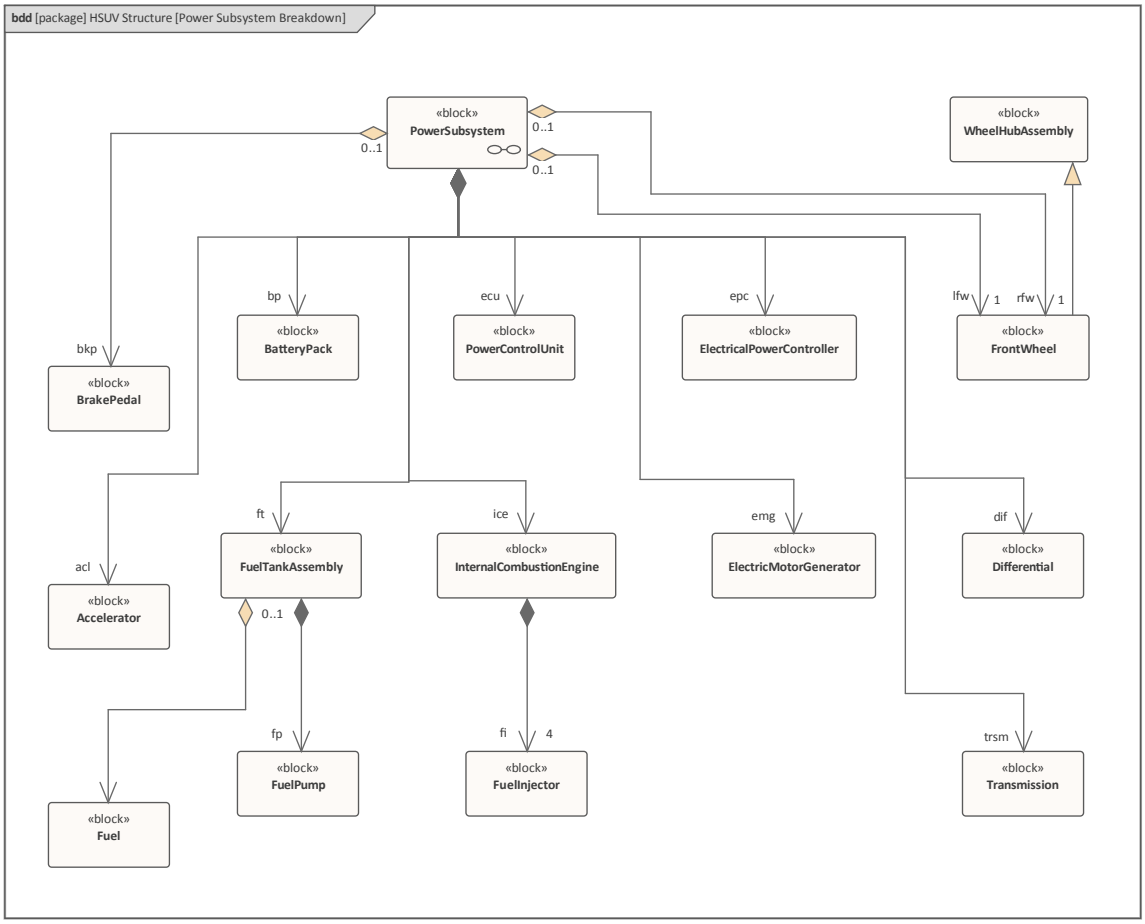
内部块图提供了一种机制来描述各部分在上下文或拥有块的时间中如何相互关联或相互连接。在我们的 Hybrid SUV 示例中，我们可以看到 Power 子系统和 Braking 子系统之间的连接可能与模型动力辅助制动有关。因此，虽然块定义图显示了组成方面的结构，但内部块能够查看块内部并查看它是如何 连接”在一起的。



我们将在后面的章节中看到如何将内部块图的一种特殊形式，即参数图用于模型系统的数学方程。

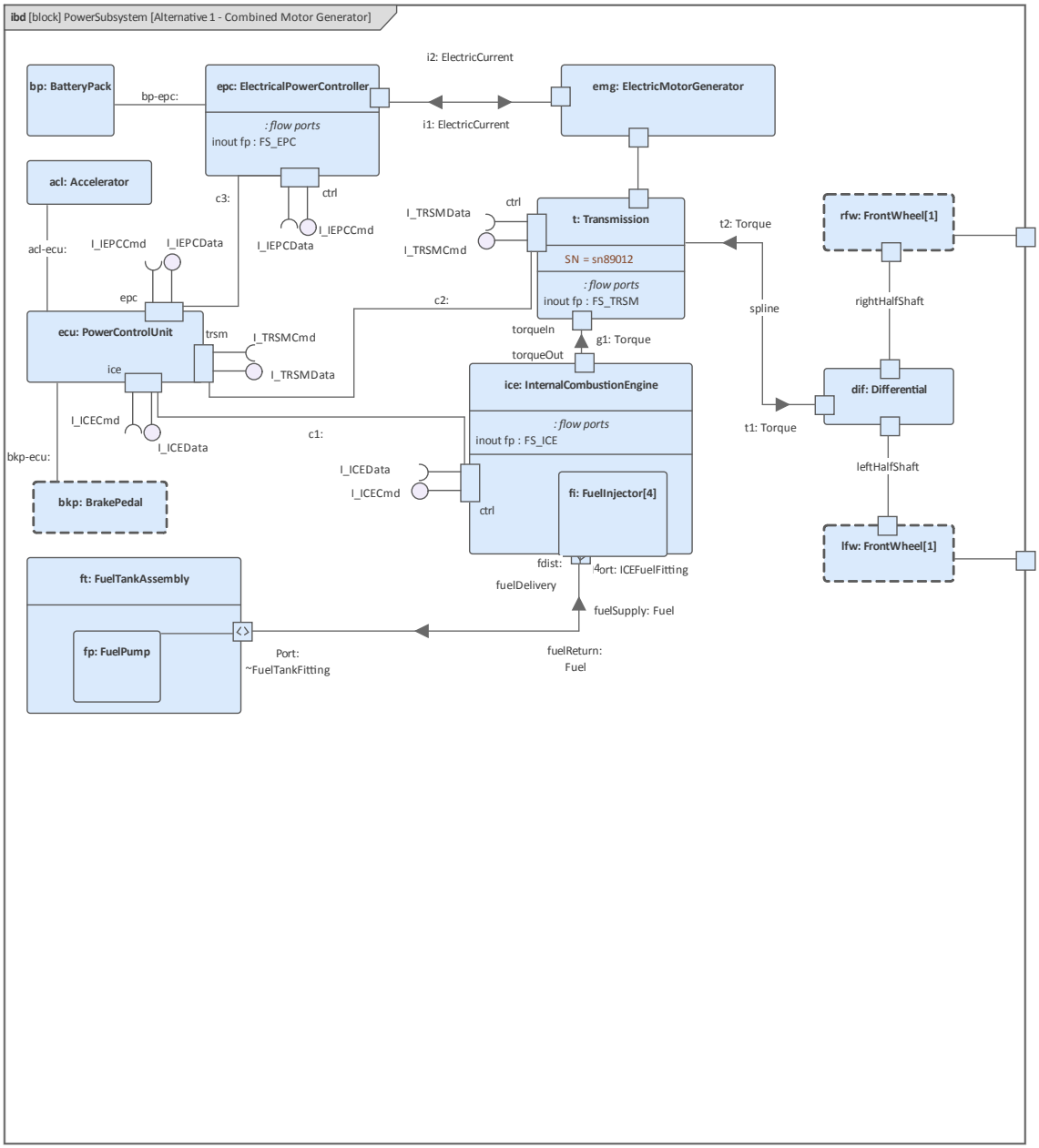
块定义图表-电源子系统

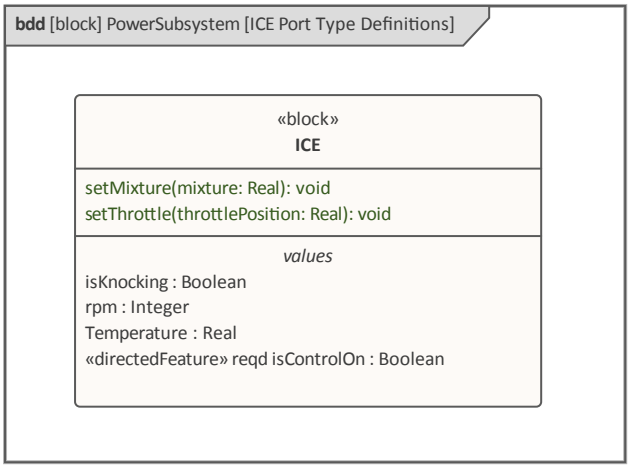
在前两节中，我们已经看到如何将系统分解为多个子系统，并且可以在块定义图上表示这种分解。我们还看到了子系统如何在整个系统的时间中与每个子系统相关上下文，以及如何在内部块图中表示这些子系统。我们现在将现在如何使用块定义图再次将这些子系统之一（电源子系统）分解为多个组成部分。



电源图表内部块模块图

类似于我们展示子系统之间的连接的方式，使用内部块图，我们可以做同样的事情来表示子系统的各个部分连接在一起的方式。因此，我们再次看到块定义图和内部块图这两种图表类型如何串联使用来描述系统的结构，以及我们如何将部件层次结构向下移动到可以理解复杂性的程度。不需要进一步建模。




bdd [block] PowerSubsystem [ICE Port Type Definitions]
«block»
ICE
setMixture(mixture: Real): void
setThrottle(throttlePosition: Real): void
values
isKnocking : Boolean
rpm : Integer
Temperature : Real
«directedFeature» reqd isControlOn : Boolean

定义端口和流

本节主题中的图表显示了如何使用端口、流和块定义上的流规范、内部块和参数图对流项进行建模。大多数物理系统都会有流动的项目，这可能是系统工作方式的重要组成部分。我们可以考虑一些例子，包括：

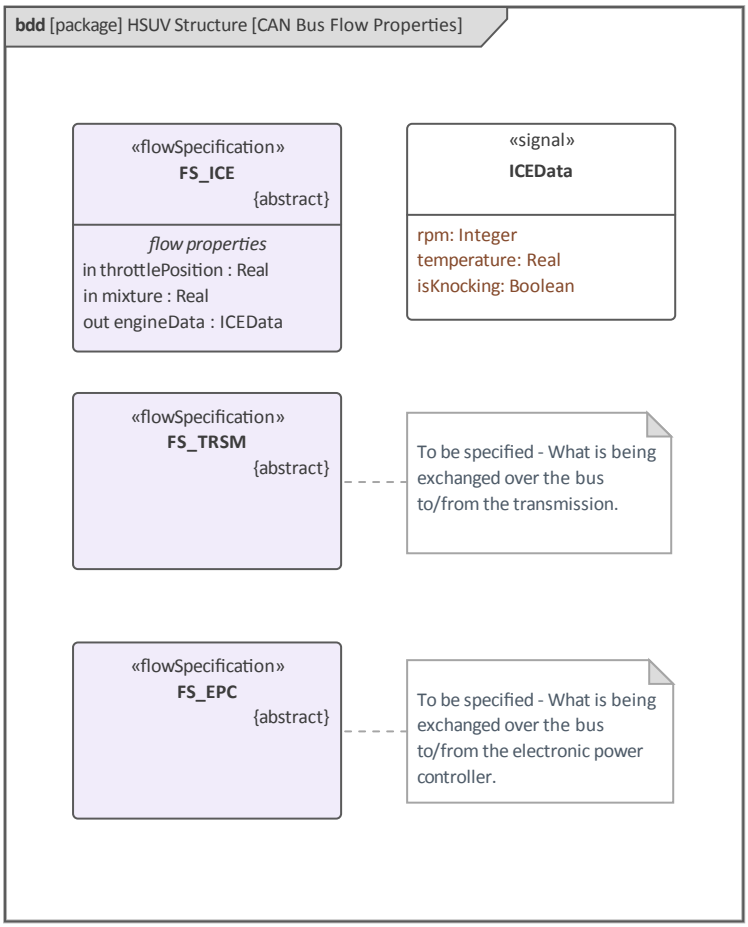
- 海水淡化厂A盐和淡水以及电力都流过系统
- 装配、零件、动力和机器人控制指令流经系统A生产线
- 火车、电车、公共汽车、渡轮和乘客流经该系统的城市交通系统
- 燃油、空气、控制信号、液压油、乘客、乘务员流经系统的飞机

这些图表首先定义了控制器区域网络 (CAN) 总线架构，并显示了如何使用一些流量规范来定义项目在电源子系统的各个部分之间流动的方式。燃料流使用块定义图建模，该图展示了流端口（在 SysML 版本1 .5 中已弃用），显示逻辑 管道”允许燃料从燃料组装和内燃机流出。内部块图采用此此外，最后一个参数图用于显示如何使用约束中定义的燃料流量数学方程来模型方程。然后使用Enterprise Architect的模拟功能可视化模拟图。

块定义图表- ICE Flow属性

此图显示了定义和改进总线架构的第一个（未完成的）步骤。建模者使用流量规格来模型物品在车辆中流动的方式；例如，已为以下内容定义了流规范：

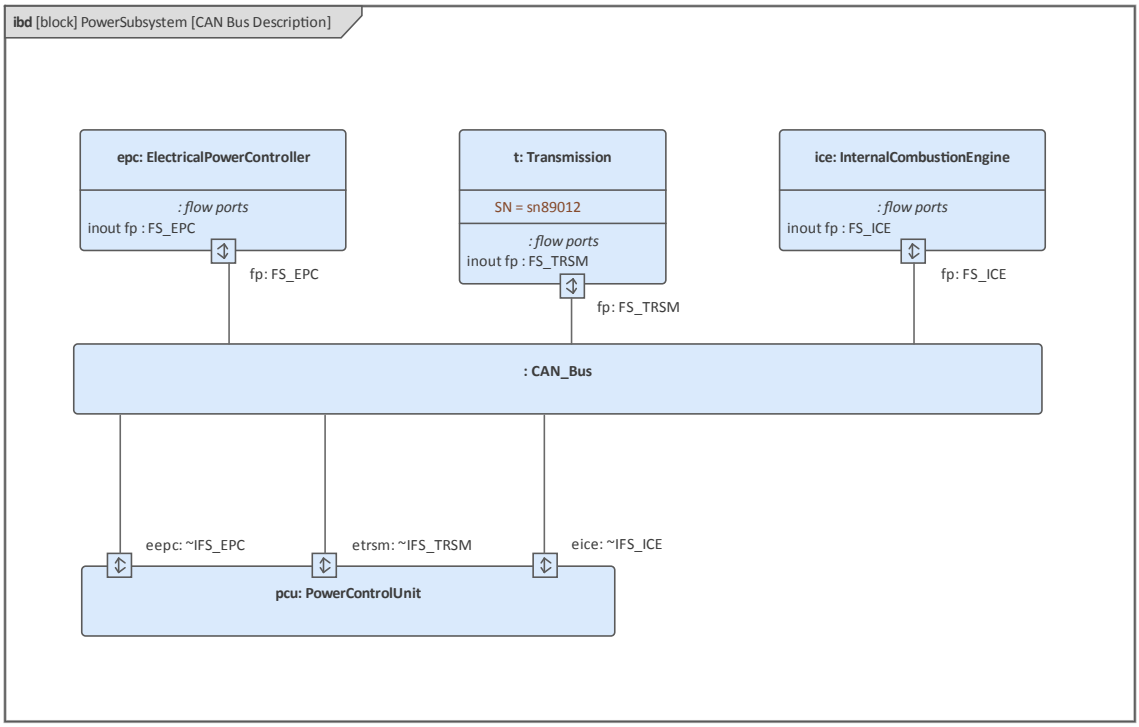
- 内燃机FS_ICE
- 系统
- 电力控制器



该图将在流程的后续迭代中进行细化，模型端口和流程将用于在各个子系统之间流动的项目。

内部块模块- CAN图表

此图继续细化总线架构，使用内部块图显示各种系统如何集成到控制器区域网络 (CAN) 总线中。该 CAN 总线架构是用于控制和集成混合动力 SUV 子系统各个部分的中央设备。

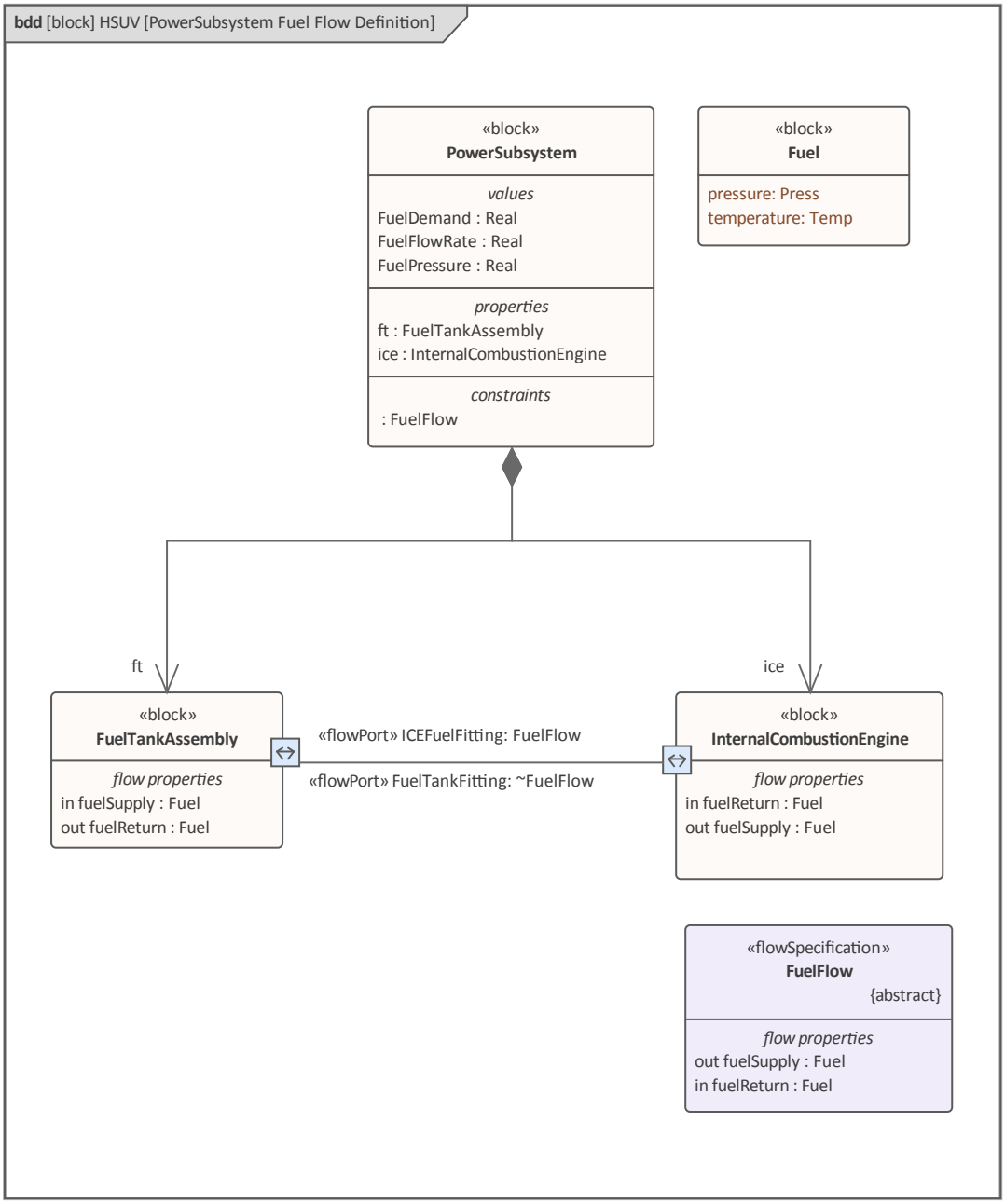


块定义图表- 燃料流量属性

此块定义图继续对从燃料箱组装到内燃机的燃料流进行建模，详细说明了燃料流的定义。燃料本身被建模为一个块，并且有两个值属性定义了重要的物理特性，即：

- 燃油温度
- 燃油压力

动力子系统块分为两个重要部分，即油箱组装和内燃机。这两个部分定义了流端口，并且在两个端口之间绘制了一个连接器，表明该项目可以从燃油箱流向发动机。

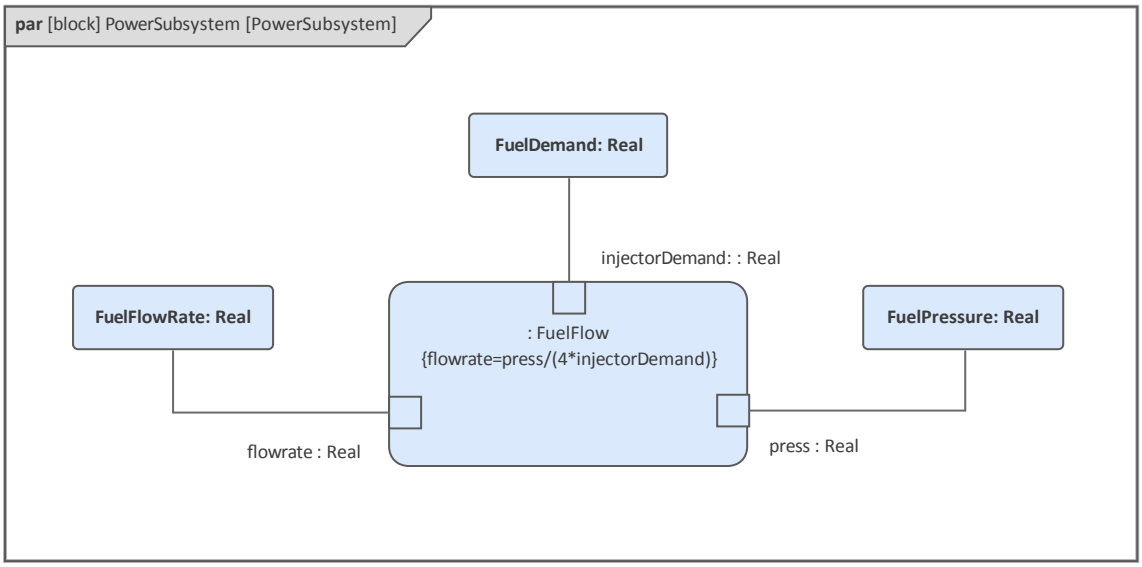


参数图表- 燃料流量

此参数图演示了如何使用 ConstraintProperty 和绑定到 ConstraintProperty 周边的参数对数学方程进行建模。在此图中，我们看到流量与燃料需求和燃料压力有关，使用以下等式：

{flowrate=press/(4*injectorDemand)}

约束在约束块中建模，并且可以在许多不同的上下文中使用，使用参数图上的 ConstraintProperties。 Enterprise Architect具有高级仿真功能，它使用 OpenModelica 或 Simulink 创建建模方程图。



分析性能（约束图表、时序图表、视图）

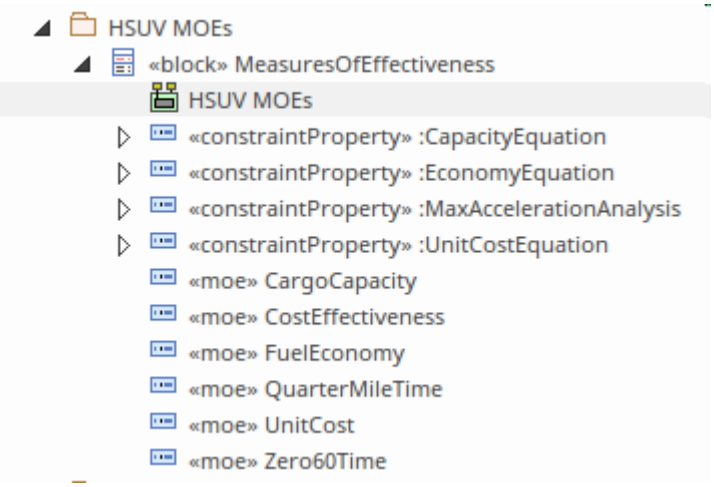
示例的这一部分中的图表主要是描述观点和视图的包图，用于解决利益利益相关者的担忧。这些主题还介绍了可用于贸易研究以评估候选解决方案和架构的有效性度量 (MOE)。视点和视图的概念在 ISO-视图（以前的 IEEE-1471）中进行了阐述，SysML 视点和视图结构与该标准一致。视图的典型示例包括操作、制造或安全，然后这些与模型元素相关。

视点和视图模型最好被认为是一种叙述或描述模型，它有助于阐明和解释系统模型。视点和视图模型展示了A或多个系统模型的元素。更具体地说，视点是用于查看系统模型的特定框架，并且是用于构建视图以解决对利益相关者具有重要意义的一组关注点的规则规范。例如，性能架构师对安全架构师会有不同的关注。视图旨在从指定的视点可视化系统。这为利益相关者提供了一种机制，可以从他们的观点指定系统模型的重要方面，然后在特定的视图中表示系统的这些方面。

观点描述了一组利益相关者查看系统模型的观点或镜头，并且通过将利益相关者的关注点与生成视图的方法一起视图，可以解决他们的关注点。该方法描述：

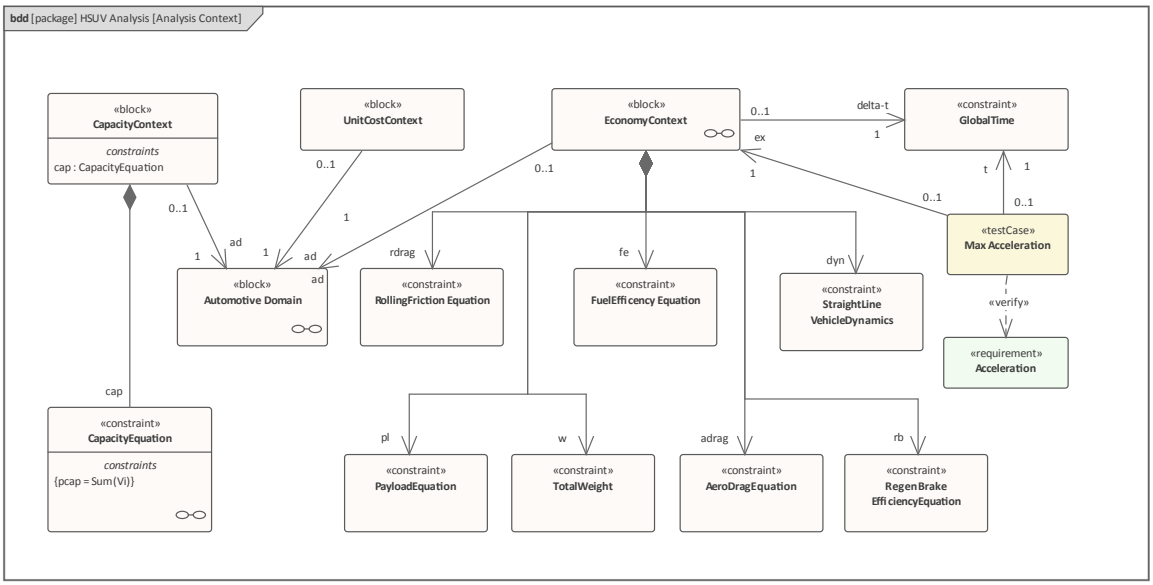
- 利益相关者希望从模型中看到的期望
- 利益相关者希望如何组织和呈现信息
- 利益相关者以工件方式消费信息。

换句话说，过程是一组规则，描述了视图应该如何表达来自模型的信息以解决利益利益相关者的担忧。当在Enterprise Architect中对视图和视点进行建模时，可以定义与建模元素的关系。



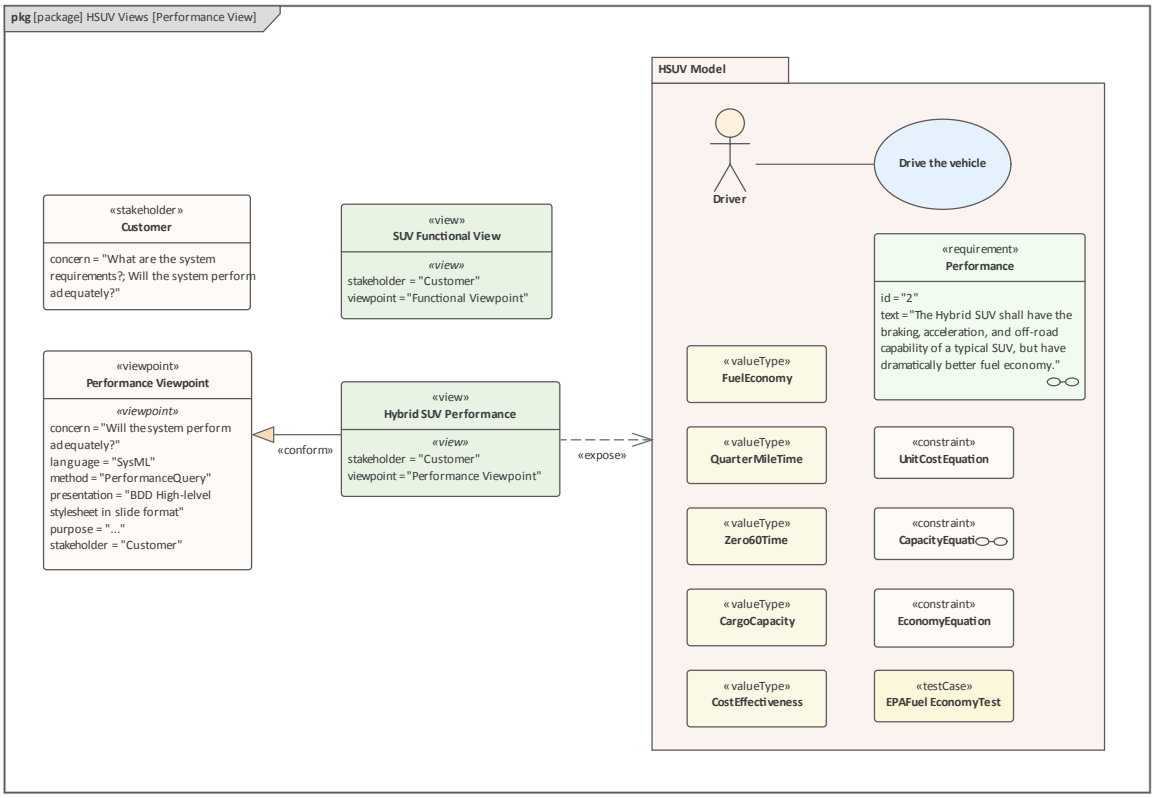
块定义图表-分析上下文

对系统进行建模的一个关键原因是能够对模型进行分析，这提供了一种经济高效的方式来深入了解构建的系统将如何在现场执行。对模型进行分析比构建原型更便宜、更方便。此块定义图定义了将用于在此示例中进行分析的各种模型元素。它描述了将用于分析的每个约束块和相关方程，以及它们之间的关键关系。图上有两种类型的元素- 块和约束。该图还显示了需求和测试用例之间的验证关系。



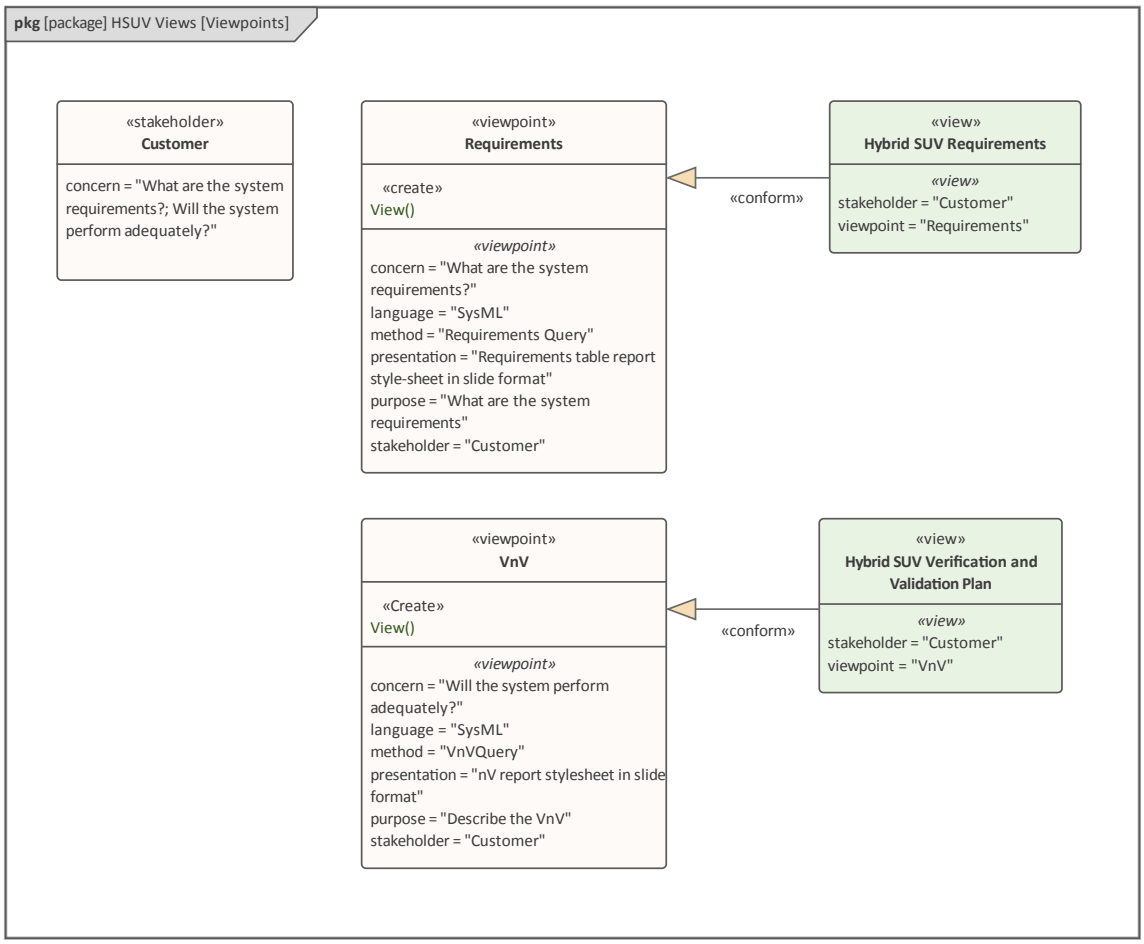
包图-性能视图定义

SysML 允许团队定义自己的观点；**在此示例中**，我们看到了用户定义的性能视图，以及填充视图特定性能视图的元素。性能视图本身可能包含许多描述其包含的元素的图表。我们可以在图中看到已经定义了许多视图，**包**括混合动力 SUV 性能视图和 SUV 功能视图。每个视图都有一个利益相关者定义和一个视图点。暴露关系已用于将性能视图与 SUV模型相关联，并且一致性关系表明性能视图与性能视图一致。



包图-观点定义

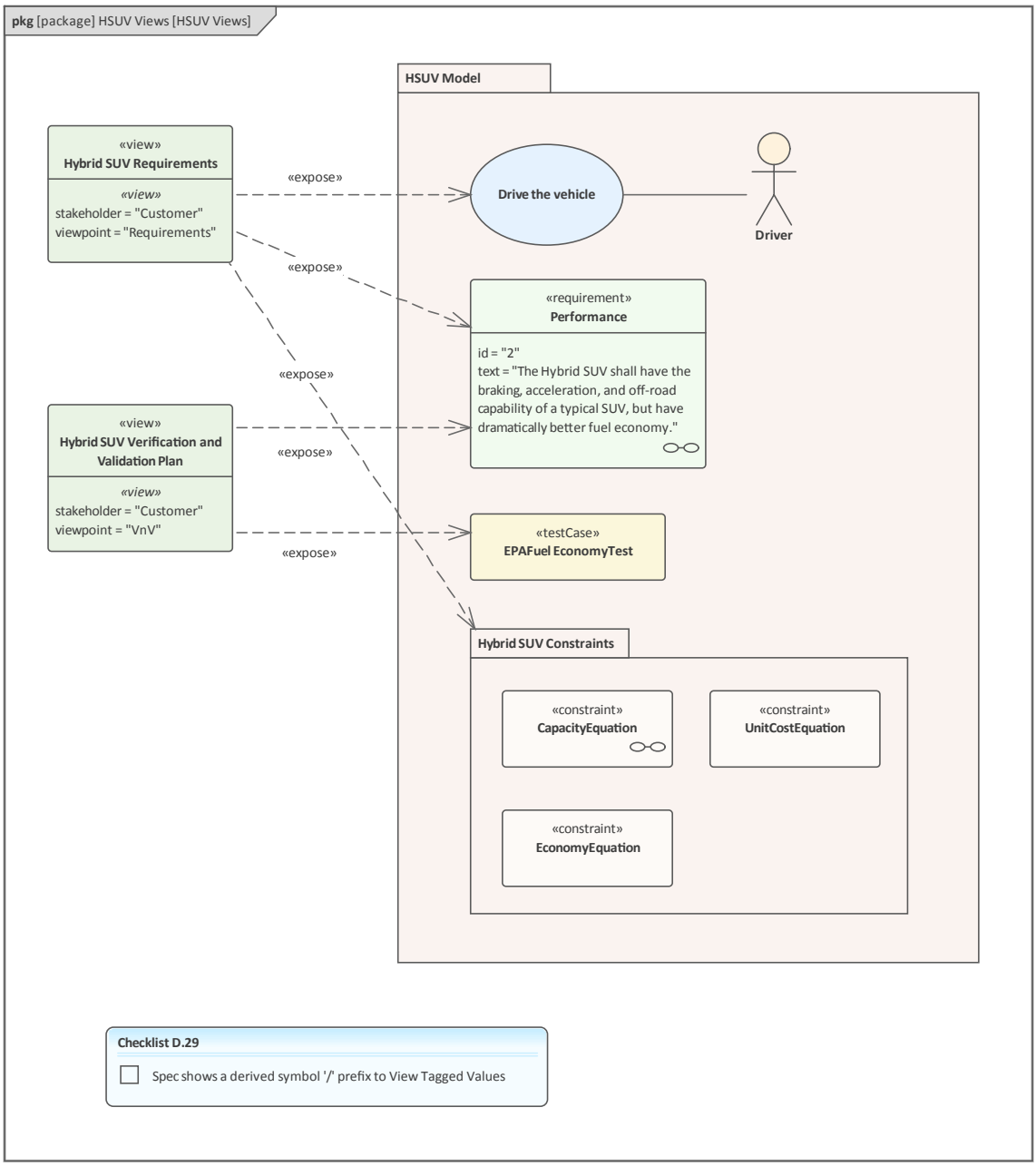
此包图显示了需求和验证与确认 (关系) 观点定义以及与利益相关者、关注点和观点的关系。利益相关者和观点通过以文本形式显示为关注属性值的评论共享相同的关注点。如果需要，注释可以用图形显示与利益相关者和观点的关系。注记，利益利益相关者属性的值是刻板印象的一个实例，而不是应用刻板印象的类。



包图-视图定义

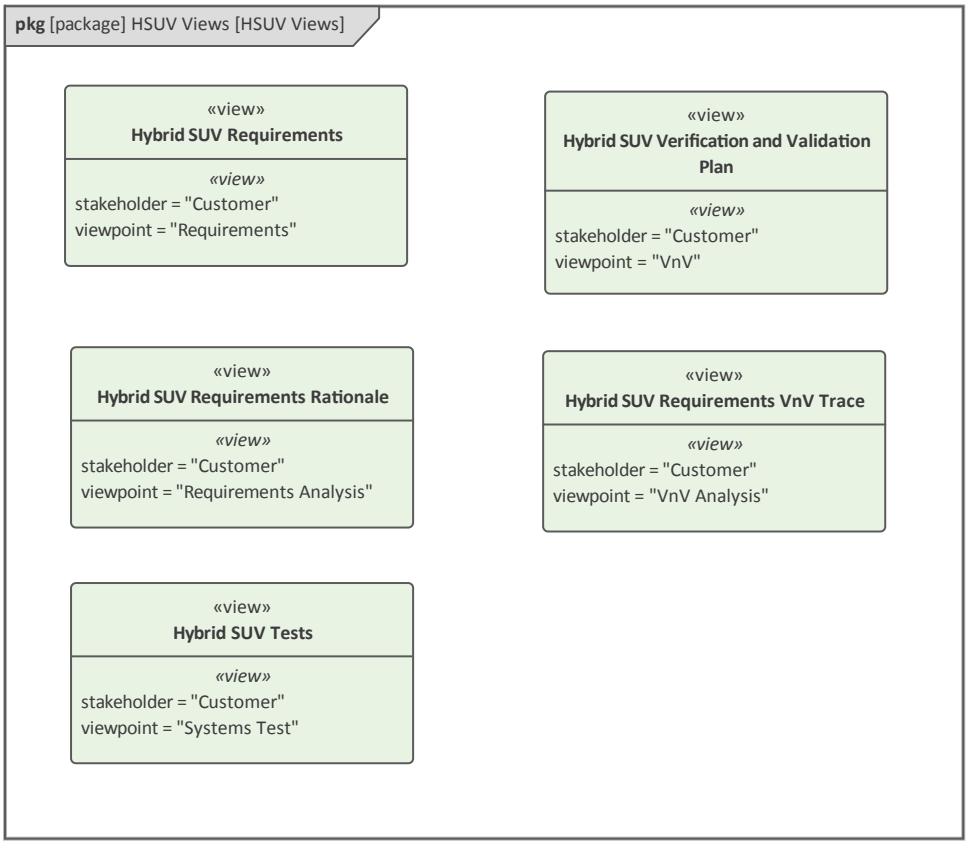
此图显示了 Expose关系的使用，它是一种图形设备，用于指示作为视图一部分的元素（包括包）。需求以及验证和验证视图具有指向模型中许多元素的外向暴露关系。这提供了一种有用的方式来指示视图中涉及的元素；例如，从图中可以看出，Hybrid SUV用需求视图暴露了Drive Vehicle用例、Performance需求和包含一组SUV约束的包。

使用一些特征丰富的可视化工具，还可以可视化给定元素参与的视图，例如驱动车辆用例可能出现在许多不同的视图中。



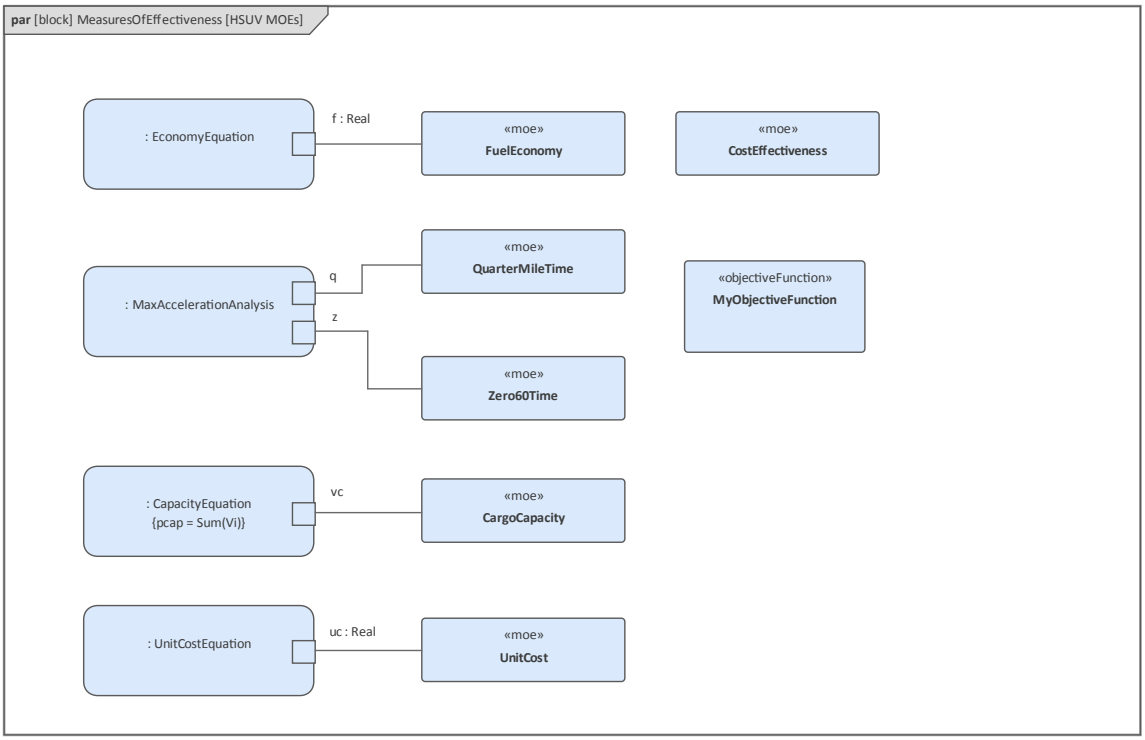
包图-视图层次结构

这个包图显示了如何将视图或任何其他具有相同原型的元素收集到一个包中并以视觉方式呈现。



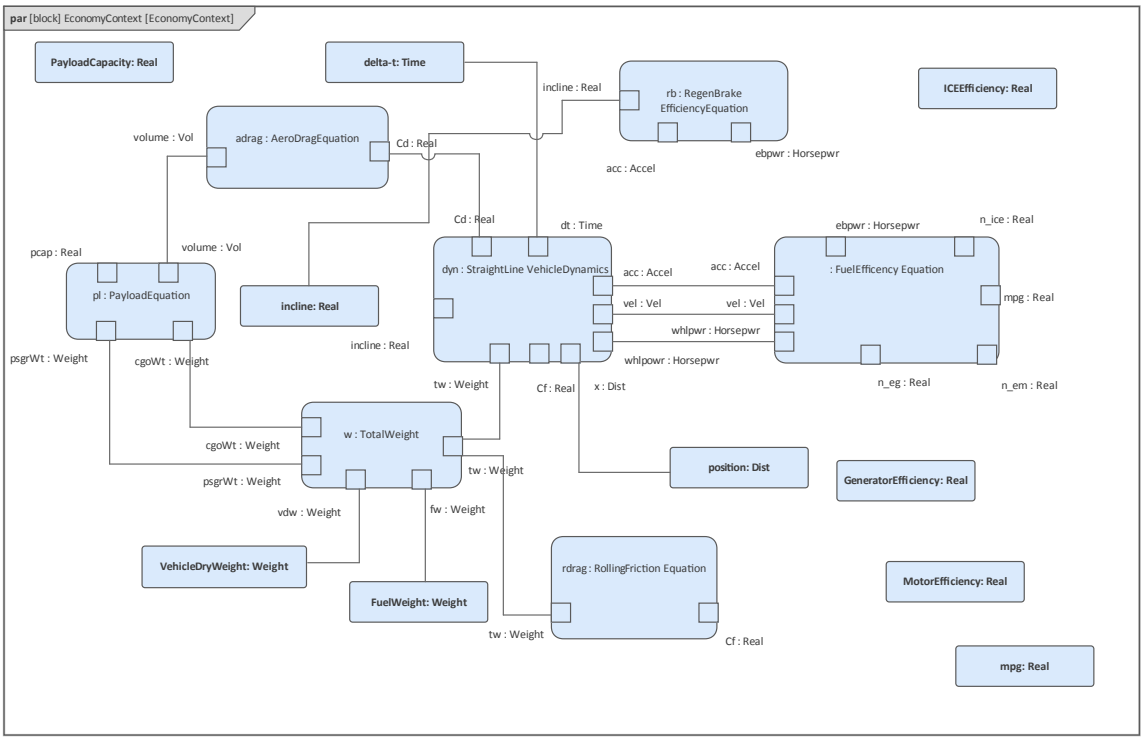
参数图表- 有效性测量

有效性度量是一种通过设置一组度量来评估解决方案的机制，这些度量将允许工程团队评估一个问题的两个或多个解决方案。这种技术通常称为贸易研究，计算两个或多个解决方案的有效性度量 (MOE)，并使用实用程序（目标）函数进行比较。 MOE 是用户定义的原型，不是 SysML 核心语言的正式组成部分；它依赖于允许扩展语言语法的构造型扩展机制。此参数图显示了如何评估 HSUV 的整体成本效益。它显示了 HSUV 设计的一种备选方案的特定 MOE，并且可以重复用于评估其他备选方案。



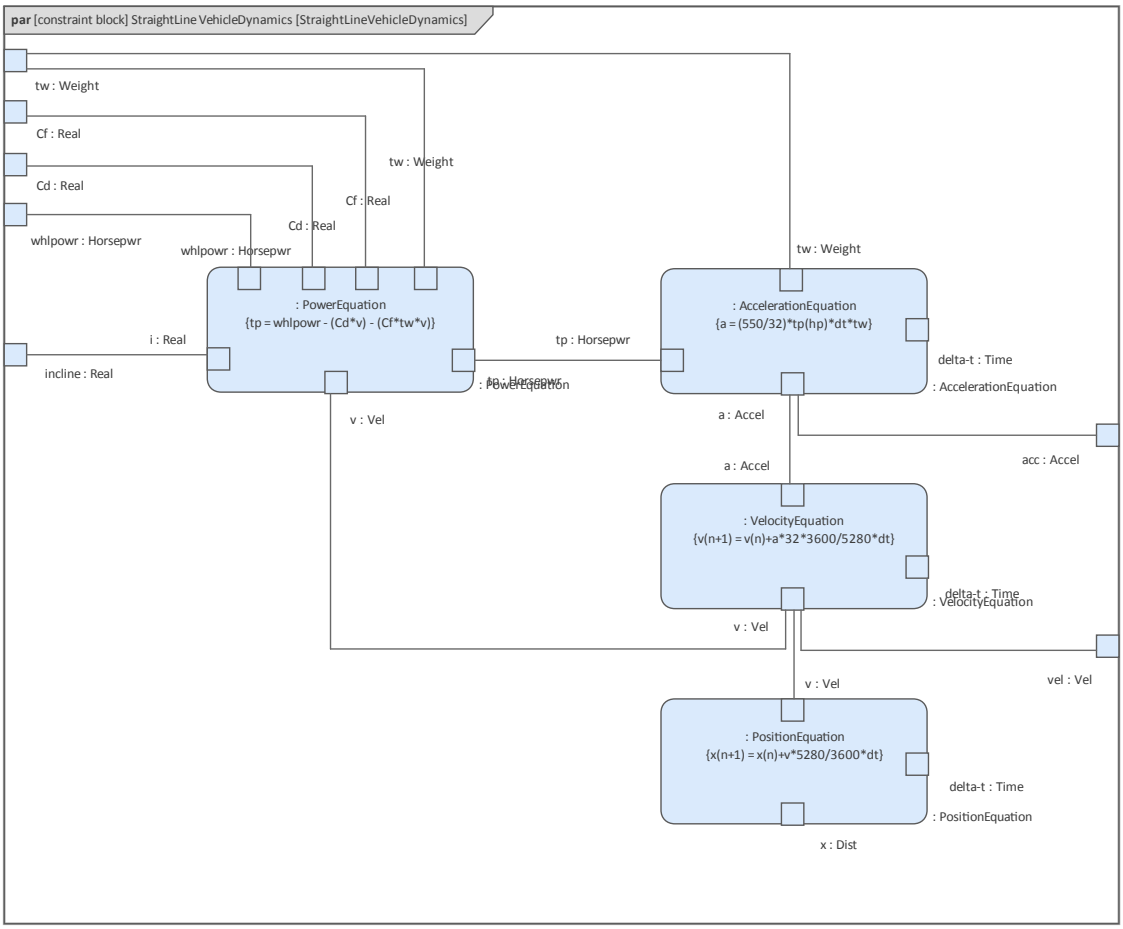
参数图表- 经济

此参数图用于模型燃油经济性，因为它已被定义为 SUV 的重要高级要求，因此需要进行评估以确保最终产品能够充分发挥作用。参数图包含约束属性（约束块的实例）。有许多方程有助于评估整体经济性，**包括：航空阻**力方程、再生制动效率方程、直线车辆动力学、滚动摩擦方程和燃油效率方程。约束属性包含通过绑定连接器连接到其他参数的边界安装参数。



参数图表- 动力学

在这个参数图中，约束块- 直线车辆动力学 - 从前面的例子中得到了扩展，以显示如何使用许多约束属性对其进行建模。直线车辆动力学约束由图框架表示，对整个方程有贡献的组成方程在图中建模为属性。约束属性所基于的每个约束都定义了一个约束方程，如图中的大括号 {} 所示；例如，约束块内的加速度方程定义为{a = (550/32)*tp(hp)*dt*tw}。捆绑连接器用于将一个方程中的参数（变量）与参数（变量）在另一个方程中。



（非规范）正时图 -图表加速度

Enterprise Architect具有使用其 OpenModelica 或 Simulink 集成生成参数图的强大功能。对物理系统建模的一大好处是能够分析系统在现实世界中的行为上下文，而无需构建昂贵的原型或必须对构建的系统本身进行测试。对控制系统运行方式的数学方程进行建模的能力，以及使用块定义和参数图创建这些模型模型提供了先兆。

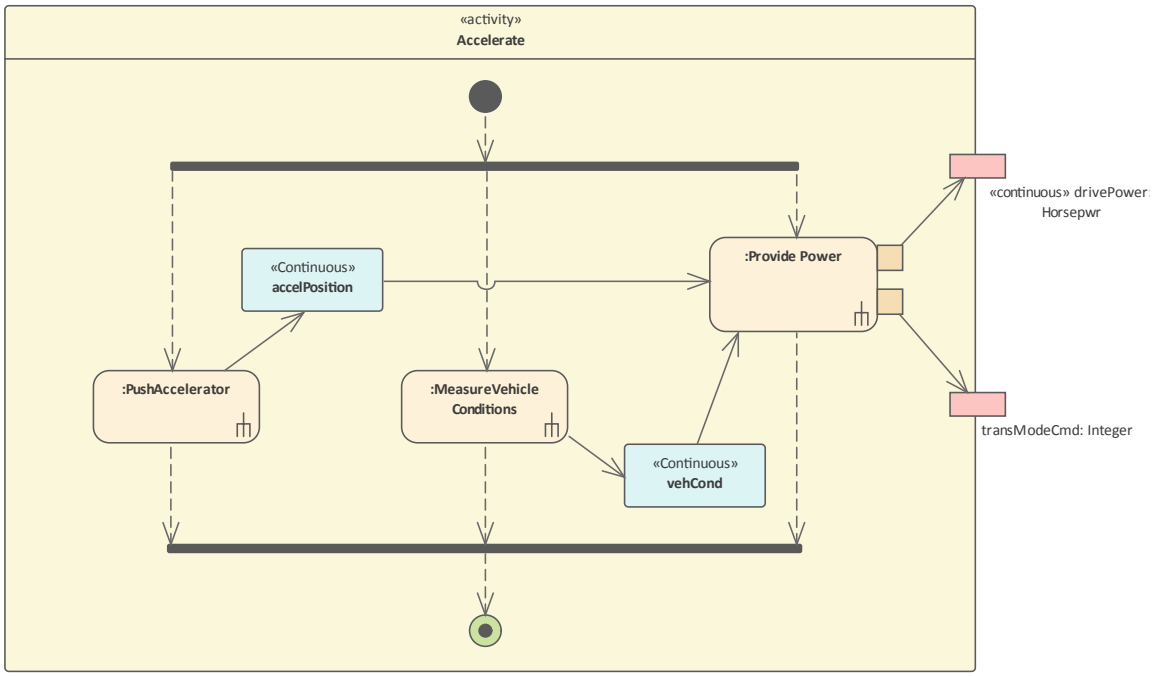
Enterprise Architect利用名为 OpenModelica 的开放工具的强大功能，该工具由 Modelica 语言支持，以生成运动方程的绘图和图形表示。

定义、分解和分配活动

本节主题中的示例使用描述 Hybrid SUV 行为方面的活动图，使用负责定义工作的行动，该工作最终由 Blocks 实例执行。还有一些内部块图展示了分配的表示方式。

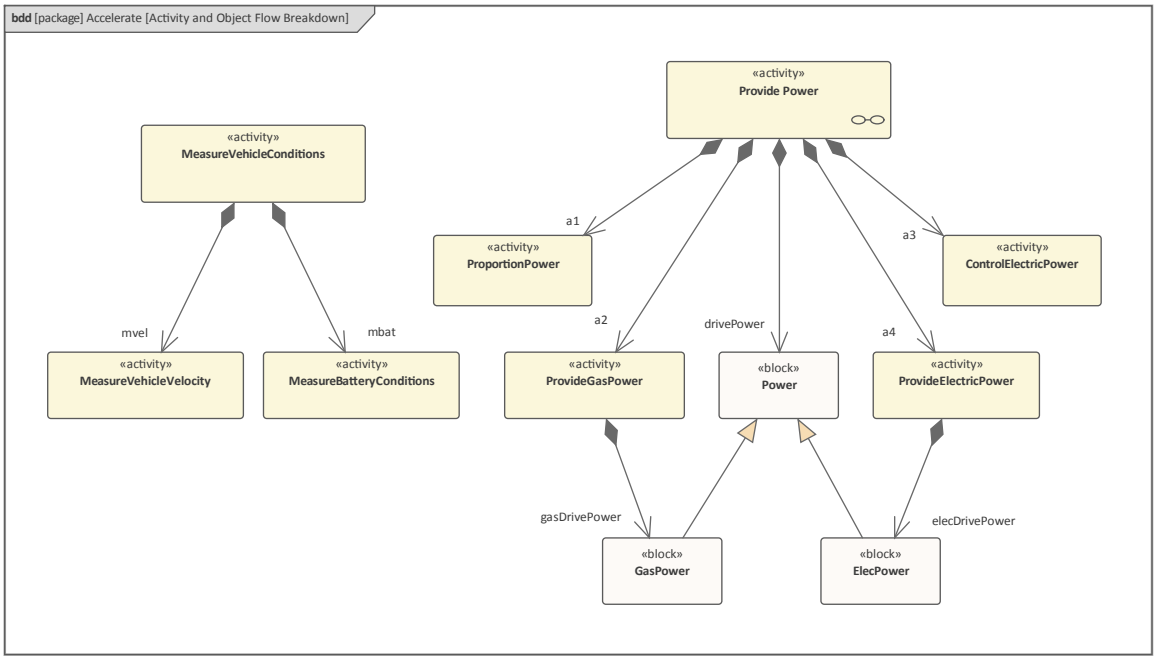
活动图表- 加速（顶层）

此活动图显示了代表 HSUV 加速度的活动的顶层行为。本例中系统工程师的意图是将此行为分配给电源子系统的各个部分。然而很快发现，所描绘的行为无法分配，必须进一步分解。图中动作之间的object节点上的构造型适用于动作调用的行为或操作的参数



块定义图表-加速度

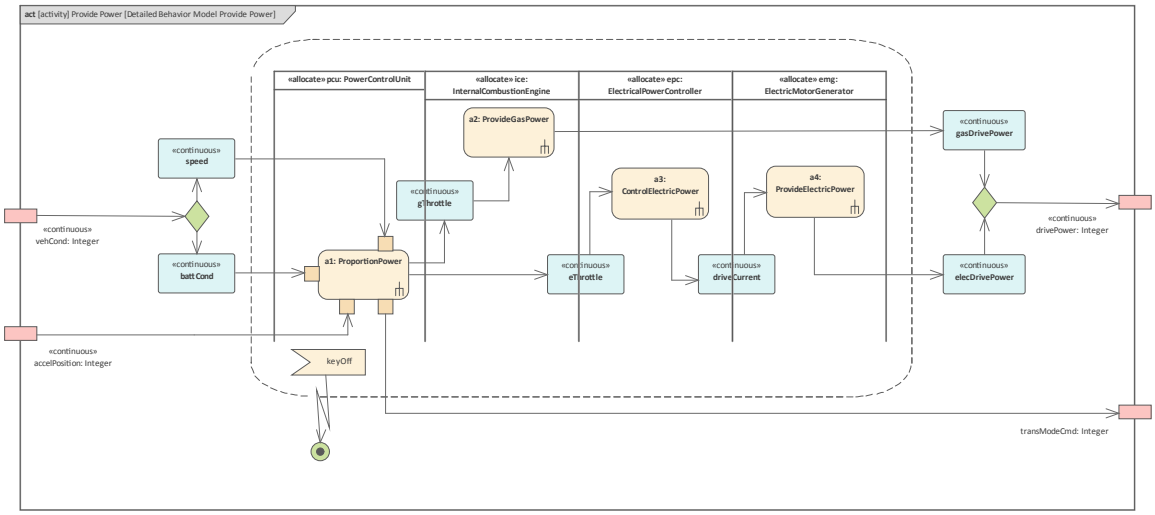
此块定义图显示了对上一主题图中的提供电源活动的分解。重要的是要注记这是一个功能分解，因此它定义了层次结构中活动之间的结构关系，因此应该在块定义图上建模。



活动图(图表) - 加速（细节）

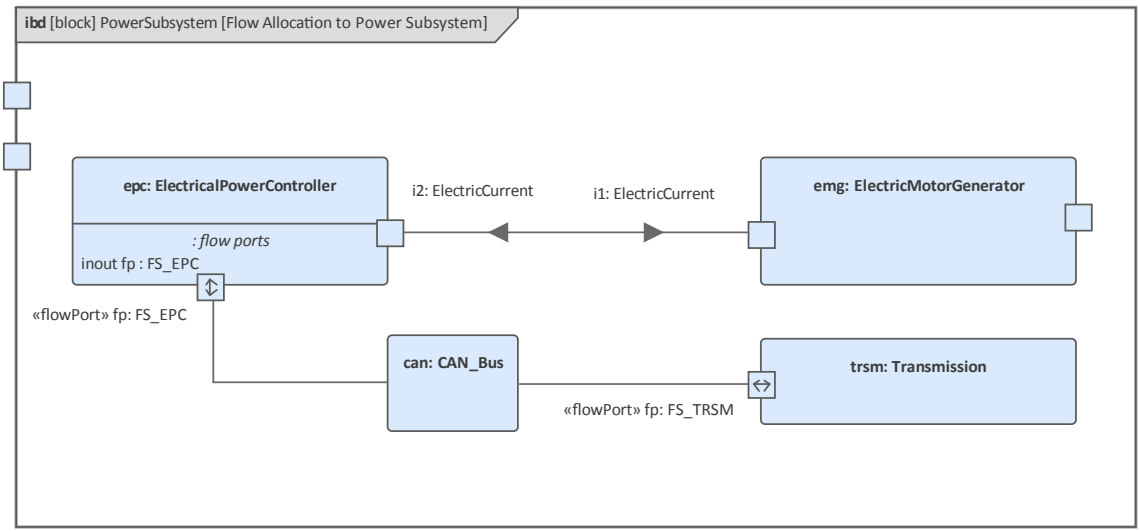
此活动图具有作为图框架的提供电源活动，其中包括从上一个图中调用分解的活动和物件节点的行动。它还使用分配活动分区，它们在图中垂直定向。这些分区包括控件、Internal Combustion Engine、Electric Power控制器和Electric Power Generator，用于显示系统的行动部分负责图中定义的动作还有一个分配标注，用于显式分配活动和object流到 Power Subsystem块中的部件。

建模工程师已将传入和传出object流用于 ProvidePower活动。这样做是为了区分电力产生的机械动力和燃气产生的机械动力的流动，并进一步了解被监测的特定车辆状况。



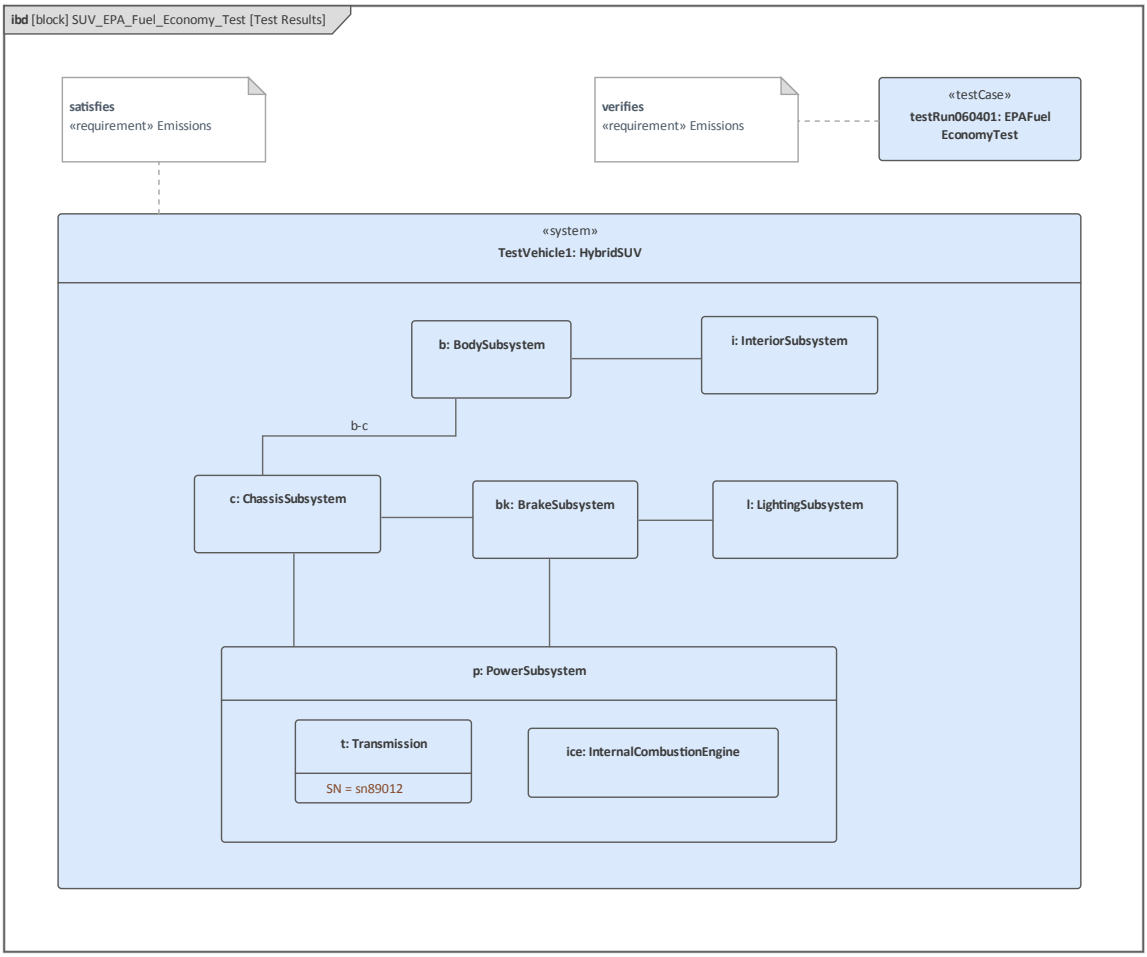
内部块图表- 电源子系统行为和流量分配

这个部分完成的内部块图详细说明了前面示例中显示的一些分配关系。在这里，我们看到作为属性添加到图表中的块是如何相互通信的，我们可以看到项目从一个块的实例到另一个实例的流动。具体来说，Electric Power控制器连接到 Electric Motor Generator，我们可以看到电流在两个属性之间流动。



内部块图：属性Specific图表- EPA Fuel Economy测试

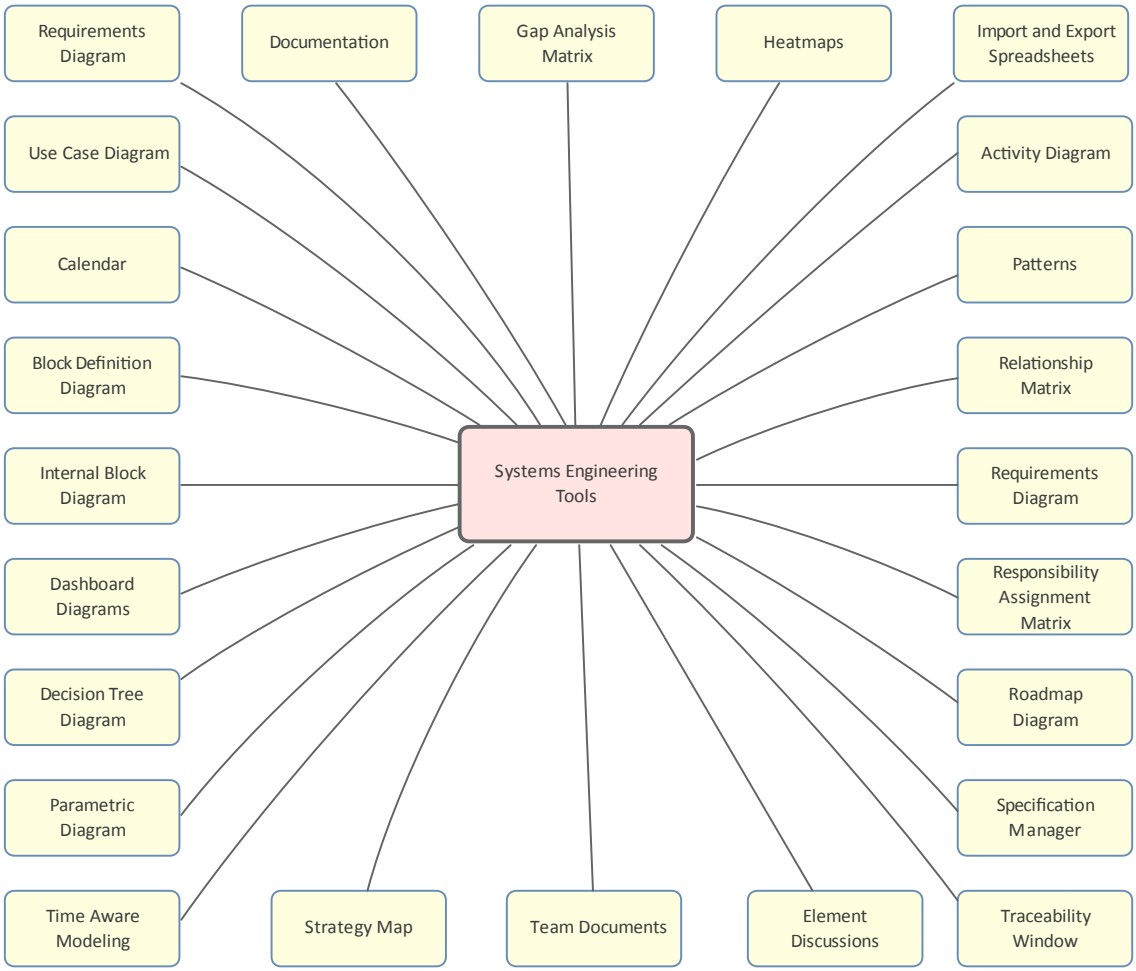
这个内部块图演示了测试案例可以在上下文表示的时间图中建模的方式。这些测试是针对可能已从生产线下线并具有特定 VIN（车辆识别号）的 Hybrid SUV 的特定实例进行的。已进行的测试是 EPA 燃油经济性测试。还可以指示特定相关部件的序列号，以识别部件实例可能存在的任何问题。



认识系统工程工具

Enterprise Architect是一个复杂而灵活的基于模型的系统平台，可用作存储库和管理工程项目的工具。它可以在整个生命周期中使用，从建立系统工程计划或实践，到规划、管理、开发和记录工程工作，再到使用设计和架构输出的实施项目的治理。该工具可用于任何流程和任何数量的表示语言，**包括** SysML、 UML 、 ArchiMate **或** BPMN。有多种功能和工具允许工程师使用他们喜欢的方法工作，例如文字处理器视图、电子表格视图、图表、关系矩阵以及一系列其他核心和扩展特征。

此思维导图显示了可用于设置和维护任意数量的基于模型的系统工程计划的关键系统工程工具的概况。虽然这**些是主要工具**，但个别团队或工程师会发现其他一系列有用的工具；这些可以通过用户指南进行探索。

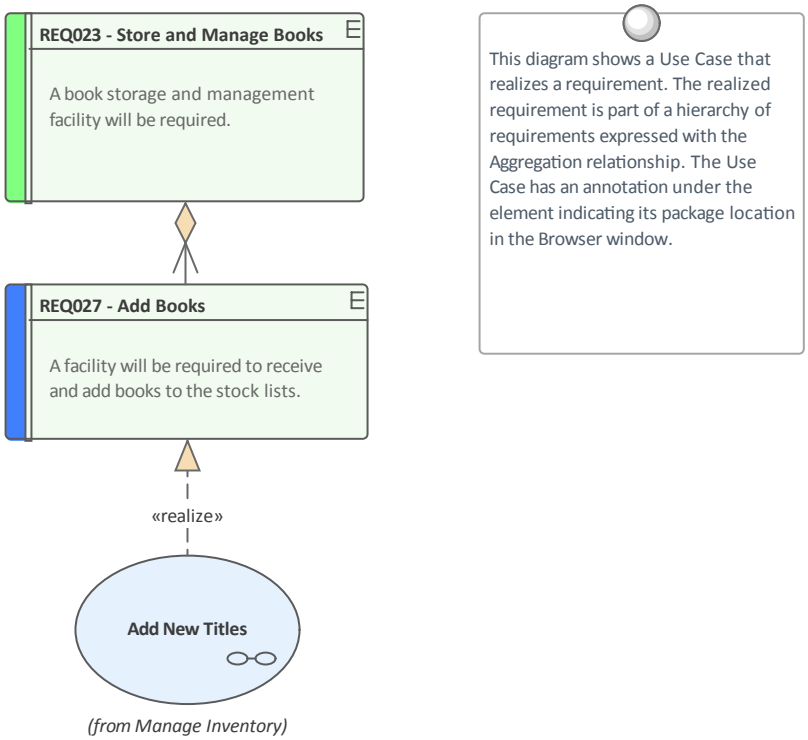


需求图表

了解需求图表

需求图表介绍

需求图直观地展示了需求如何相互关联以及与模型中的其他元素相关，**包括**业务驱动因素、约束、业务规则、使用案例、用户案例、设计组件等。该图**是**Enterprise Architect的扩展图类型之一。它提供了一个吸引人的需求图形表示，对于习惯于使用基于文本的工具的需求分析人员来说，这将是一个受欢迎的变化。



在哪里可以找到需求图表

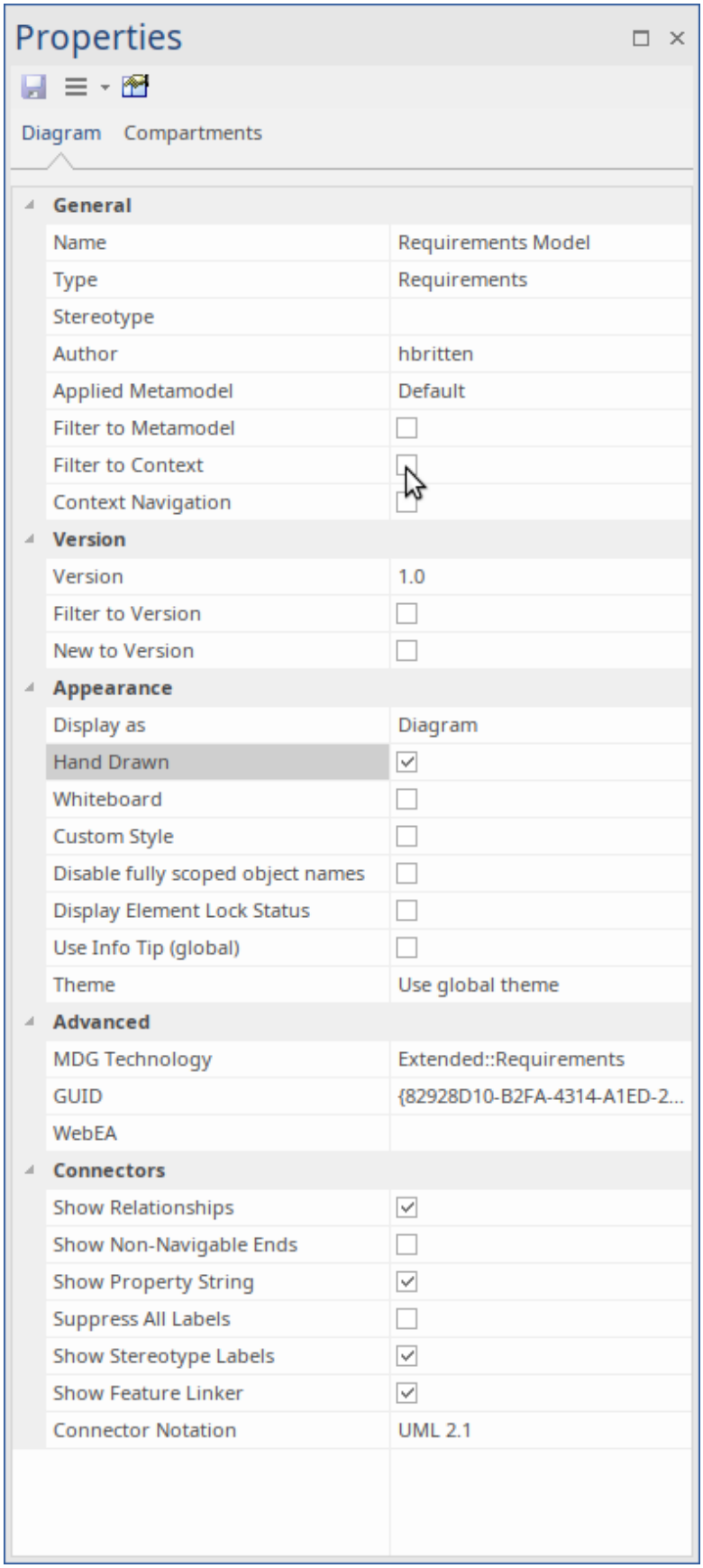
浏览器窗口 菜单： 添加图表：上下文菜单|**管理** |全部显示蓝图|扩展|**需求**

需求图表的用途

一种用法是显示需求如何在层次结构中连接在一起，**或者更重要的是**，显示需求如何与其他元素相关联。有经验的建模师会在规范管理器中定义和管理**需求**，然后用需求图来展示每个需求是如何与业务驱动因素因素等上游流程**要素**，以及使用案例、用户案例、用户案例等下游流程要素相关联的。体验设计和解决方案组件。

需求图表选项

可以更改图表的外观以适合观众，并且可以包含、隐藏或更改细节以确保图表满足其主要交流目标。有多种选择，从创建手绘风格的图表到过滤图表内**容**。

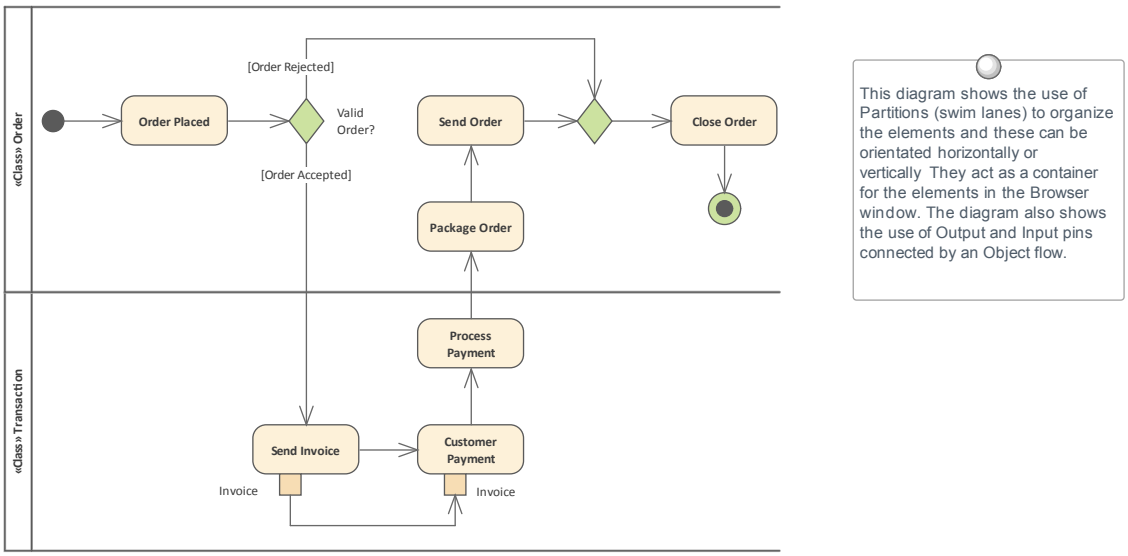


了解更多关于需求图表 Working In Diagrams

活动图表

活动图是Unified Modeling Language （ UML ）行为图之一，可用于模型过程或算法作为步骤的序列。它是其近亲流程图的更复杂的版本。活动图可用于模型业务流程，作为 BPMN业务流程流程图的UML**替代方案**；他们具有在浏览器窗口中创建活动层次结构的相同能力。

Activity Diagram showing the use of Partitions



可以给元素命名，并且可以将详细描述添加到注记中。通过用连接器（**控件流**）连接活动、决策和分叉，**序列**元素可以描述业务流程。可以通过浏览器窗口中的子流程功能图构建嵌套的活动图，并使用A流程层级来实现从价值链级别向下钻取流程。

了解活动图表

在哪里可以找到活动图表

功能区：设计>图表>添加图表> UML >行为> UML行为>活动

浏览器window Toolbar : New图表> UML > Behavioral > UML Behavioral >活动

浏览器窗口上下文菜单 |添加图表... > UML Behavioral >活动

活动图表的用途

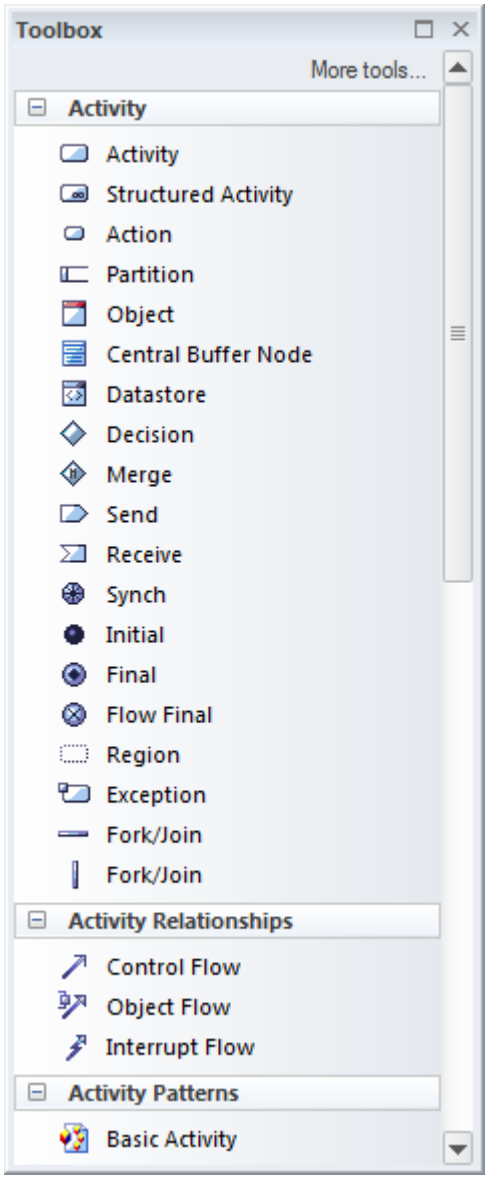
活动图可用于对具有一系列步骤的任何业务或技术活动或概念进行模型。这包括业务和技术流程以及计算机算法。这些步骤由显示控件顺序的控制关系连接。决策和合并可用于通过活动模型进行选择和进一步控制流。可以添加分叉和汇合来拆分和重新组合控制流和添加的对象，以显示数据的供应和消费方式。

活动图的图表

活动图表可以在不同的形式层次上绘制，从用于表示简单业务流程的基本流程图图表的图表到可用于模型复杂系统的复杂的基于动作的图表。有一个工**具箱**，其中包含用于创建模型的一系列元素、关系和模式。

活动图（像任何图一样）可以被视为一个元素列表，这使得使用元素属性更**容易**。

显示图表时也可以使用图表过滤器，以引起对图表部分的注意，并且可以通过更改图表的属性以手绘或白板样式呈现图表。



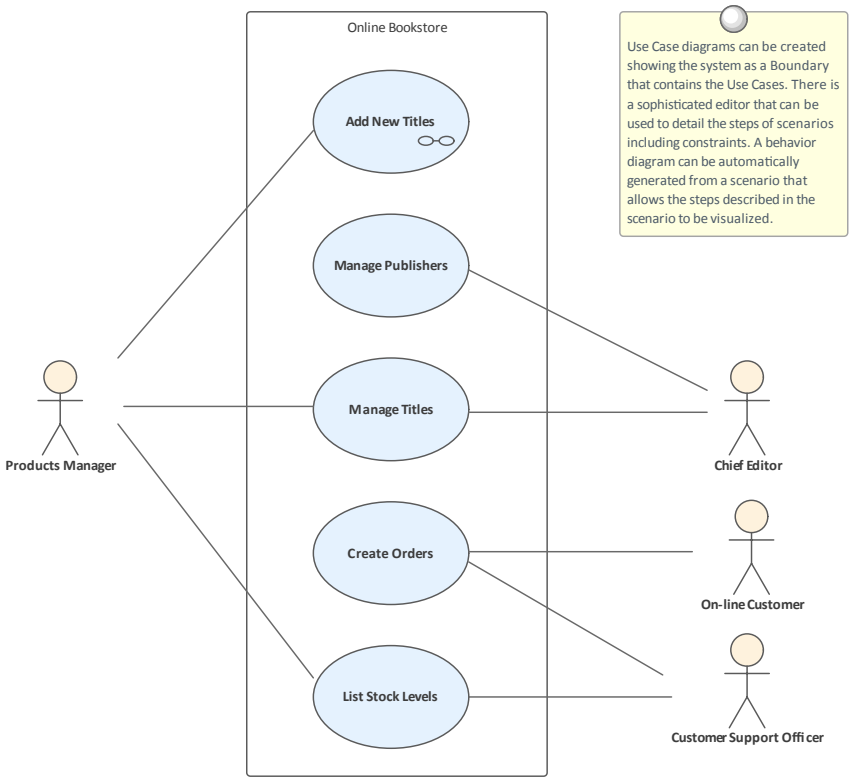
了解更多关于活动图表 Activity Diagram

用例图表

了解用例图

介绍使用用例图表

用例图是Unified Modeling Language (UML) 行为图之一，可用于描述与正在建模的系统交互的用户和其他系统的目标。它们用于描述系统、子系统或实体的功能需求，并展示了如何使用系统的简单但令人信服的画面。



它们通常与更高层次的业务和利益相关者需求结合使用，并且经常辅以一组**非功能性需求**。

在哪里可以找到用例图表

功能区：设计>图表>添加图表> UML行为>**用例**

浏览器window Toolbar : New图表> UML Behavioral >**用例**

浏览器窗口上下文菜单 |添加图表... > UML Behavioral >**用例**

用例图表用途

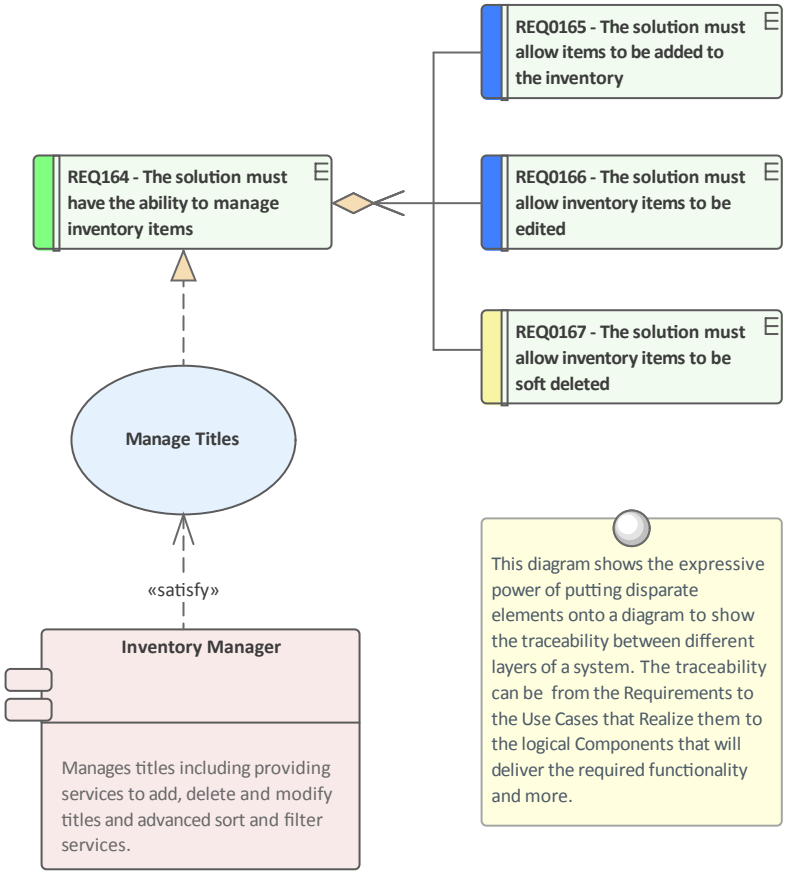
用例图用于描述用户或其他系统希望通过与系统交互来实现的目标。他们总是从 Actor 的角度描述目标，用例的细节将更准确地描述目标。

使用案例通常会作为定义测试案例的基础。

使用图表的用例

可以创建任意数量的使用用例图来表示系统的不同部分或使用案例包。图表保持简单，或者它们可以通过应用一些额外的连接器来实现，例如包含、扩展和结构概括关系。

可以包括用于命名讨论中A系统、边界或实体的系统； Actors **位于外部，而**边界位于内部使用



用例图可用于显示用例如何与系统中的其他元素相关联，包括需求等上游元素和组件等下游元素。

用例图（对于任何图）可以被视为一个元素列表，这使得使用元素的属性更容易。

显示图表时也可以使用图表过滤器和图表图层，以引起对图表部分的注意，并且可以通过更改图表的属性以手绘或白板样式呈现图表。

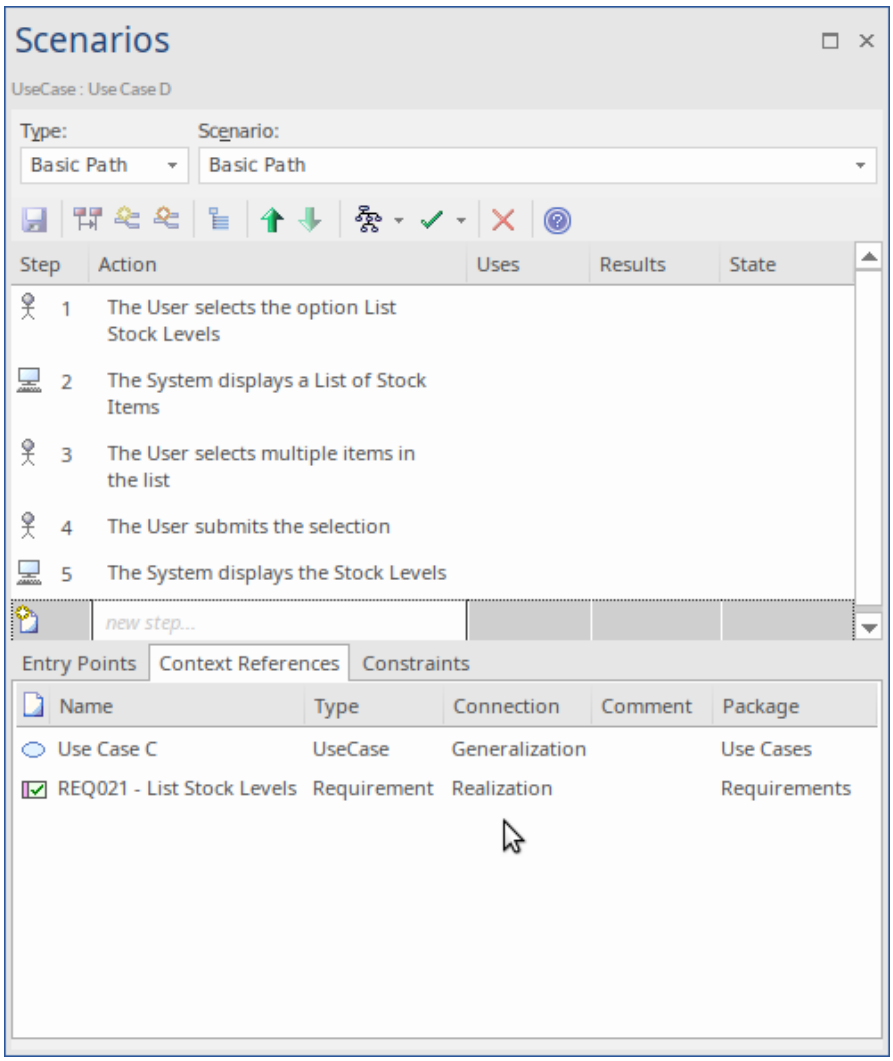
了解更多关于用例图表 Use Case Diagram

场景生成器

了解场景构建器

场景生成器简介

场景构建器用于定义一个用例的细节，包括定义详细描述、创建一个或多个场景以及定义前置条件、后置条件和其他约束。可以记录一个用例的详细步骤并将其链接到模型中的其他元素，然后这些可以生成为一个图表，提供该用例及其场景的可视化表示。图表和文本可以同步，然后可以将各个步骤跟**踪到其他元素**，例如将实现使用示例中指定的需求的用例。



在哪里可以找到场景生成器

>开始>应用> 编辑 > 职责 > 结构化场景

设计>**元素**>编辑>职责>结构化场景

元素上下文菜单：属性|职责 > 场景 |右键|**加新**：结构编辑器

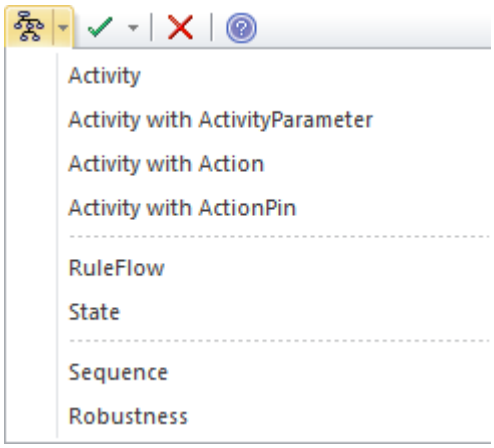
场景构建器的用途

定义一个用例的细节以及它的场景和约束，可以用来代替传统的基于文本文档的方法来定义使用案例。这确保了使用案例及其场景和约束的用例图和文字细节都包含在同一模型中并且可以追溯。如果出于合同或流程原因需要文档格式的使用案例，则可以使用内置文档引擎从模型中自动生成用例报告。

场景生成器的选项

Scenario Builder 可以被视为选项卡式或停靠窗口或元素的属性窗口。一个用

例的步骤，包括它的场景，可以从生成图表工具栏图标自动生成许多不同的图表类型。



了解更多关于 **Scenario Builder**

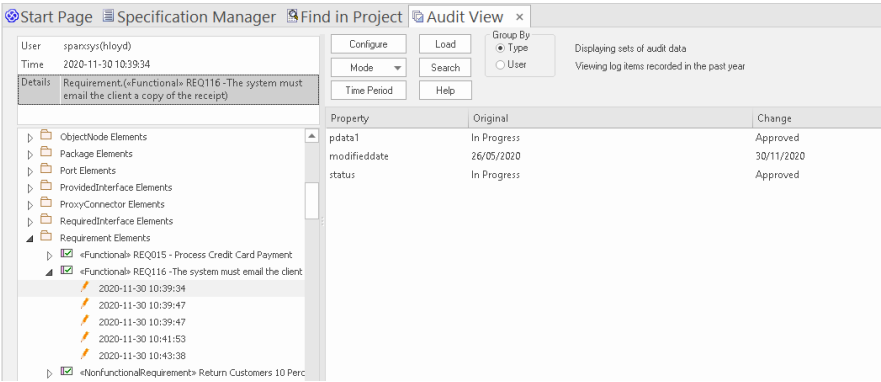
Scenarios

审计

了解审计

介绍审计

特征审计可以跟踪需求的更改，包括更改的内容、更改的时间和由谁更改。默认情况下，审计是禁用的，必须在记录需求更改之前启用。一旦启用，它就是一个被动工具，可以静默记录元素的更改。它不能代替版本控制状态或基线，也不能与这些工具相比，不能用于返回以前的质量控制。更改都是由使用状态模型审计的。



在哪里可以找到审计

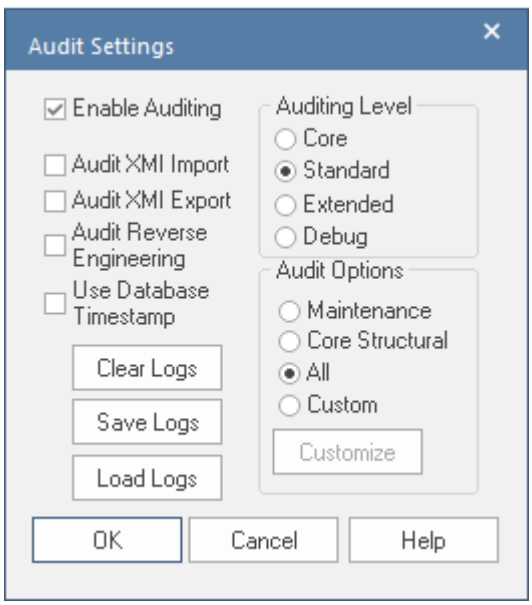
功能区：设置>模型>审计

使用审计

审计可用于跟踪模型中的更改内容、更改人员以及更改时间。有多种模式，存储库管理员可以使用这些设置来指定审计中记录的内容。虽然基线可用于显示模型和快照在某个时间点之间的差异，但审计工具会记录每个单独的变化；但是，它不能用于恢复到以前的状态。

审计选项

配置审计的设置范围很广，从启用或禁用确定哪些元素具有审计跟踪和记录的细节级别的设置开始。可以从存储库中导出审计日志以提高性能。



了解更多关于审计

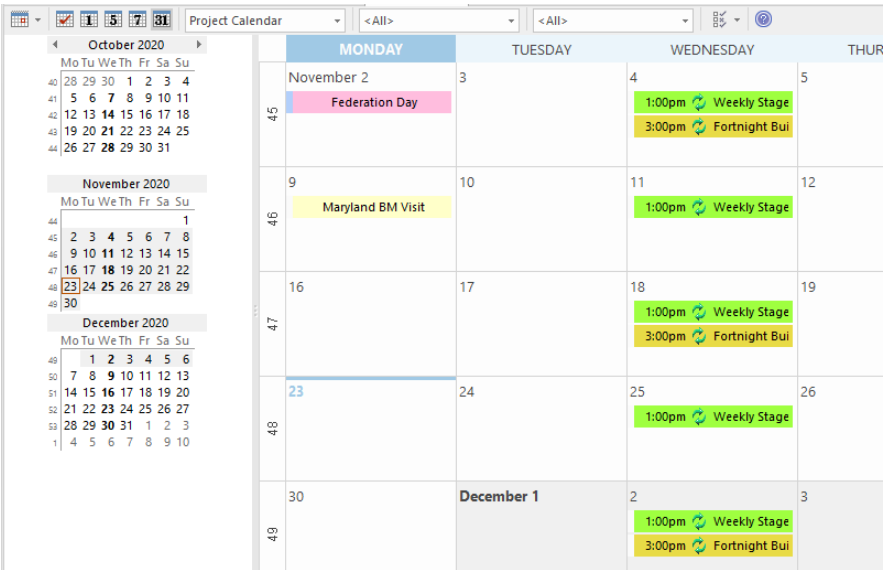
Auditing

日历

了解日历

介绍日历

日历是一种功能齐全的机制，用于记录计划中的重要事件并显示资源分配等其他信息。有日、周和月视图，显示可以设置为显示日历条目、项目任务和资源分配。当资源被分配时——例如分析一组需求——用户可以从日历钻取到浏览器窗口中的需求位置。



还有完全可配置的事件类型、类别和颜色。业务分析师的工作，包括各种活动，**包括**：研讨会和会议、聚焦小组、协作游戏、评论、观察。所有这些事件都可以在日历中方便地记录和管理。当资源已分配给元素和任务已分配给个人时，这些可以显示在日历中。

在哪里可以找到日历

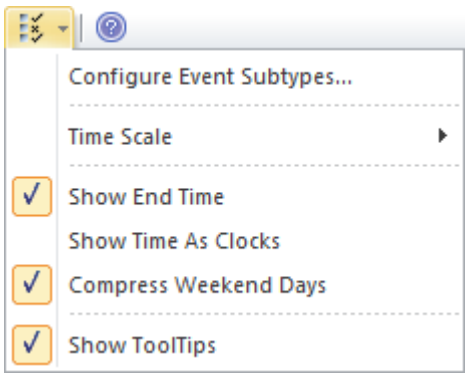
功能区：开始>协作>日历

日历的用途

日历可用于安排和查看会议、里程碑、评论、研讨会等活动。它可用于查看资源对存储库中元素的分配，例如谁在分析一组需求。它还可用于查看项目任务。分析人员可以方便地单击浏览器窗口或项目任务中的元素。

日历选项

日历有许多选项，包括创建周期性事件的能力。有一个选项工具栏图标，**允**许配置日历外观的各个方面。



了解更多关于日历

The Model Calendar

块定义图表

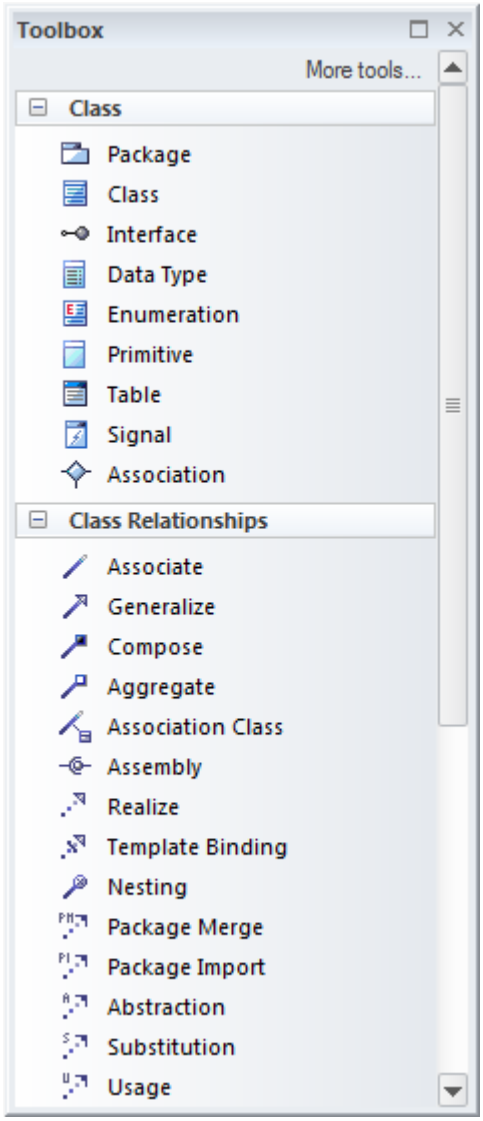
了解块定义图表

介绍块定义图表	块定义图是系统建模语言（SysML）结构图之一，可用于模型广泛的事物。它是用于对业务和技术领域中的实体进行一般的通用图表，包括术语和概念、业务规则以及 XML 和数据库模式中的功能。
在哪里可以找到块定义图表	功能区：设计>图表> 添加图表:块系统工程1类型图表> 浏览器window Toolbar : New图表:块系统工程1类型图表> 浏览器窗口上下文菜单|添加图表：块系统工程1类型图表>
块定义图表的用途	每当需要系统的逻辑或结构表示时，都可以使用块定义图。它适用于对业务和工程概念进行建模。它是对系统或子系统或其组件之一的结构进行建模的基本图。
块定义图表的选项	块定义（像任何图表一样）可以被视为一个元素列表，这使得使用元素的属性更容易。 显示图表时也可以使用图表过滤器来吸引对图表部分的注意，并且可以通过更改图表的属性以手绘或白板样式呈现图表。
了解更多关于块定义图表	Using Blocks to Model Structure and Constraints

内部块图表

了解内部块图表

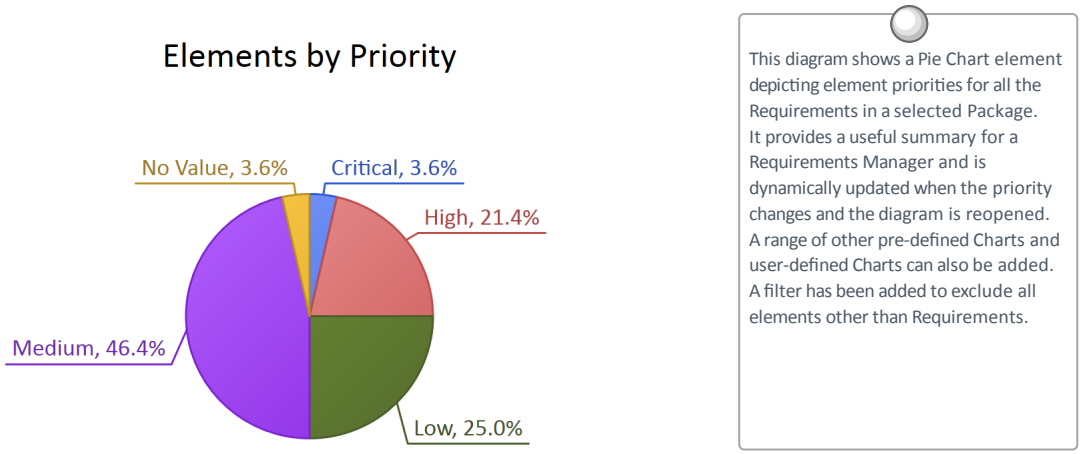
内部块图表	内部块图与块定义图表使用，但它通常用于显示块的内部结构，包括其部分以及它们如何协同工作以提供块指定或已分配给它的行为.
在哪里可以找到内部块图表	功能区：设计>图表> 添加图表:图表系统工程1类型内部块required type> 浏览器window Toolbar : New图表:图表系统工程1类型内部块<必需类型> 浏览器窗口上下文菜单|添加图表类型系统工程1图表内部块必需类型>
内部块图表的用途	内部块图用于模型块的内部结构，包括其组成部分以及这些关系之间的关系。
内部块图表的选项	内部块图（像任何图一样）可以被视为一个元素列表，这使得使用元素的属性更容易。 显示图表时也可以使用图表过滤器来吸引对图表部分的注意，并且可以通过更改图表的属性以手绘或白板样式呈现图表。



了解更多关于内部块模块的图表 Using Properties and Parts to Model Block Usage

仪表板图表

仪表板图允许您创建高质量的图表和图形，以视觉上引人注目的方式显示存储库信息。此图是在**Sparx** Sparx Systems Enterprise Architect中创建仪表板图的示例；它说明了图表中需求优先级的比率。



Enterprise Architect提供了一个预先配置的图表和图表的工具箱页面，但您可以自由地创建和保存任意数量的图表，从存储库中的任何位置获取数据。图表提供了有价值的汇总信息，**有助于需求管理。可以使用大量可用的**图表和报告元素轻松跟踪和记录高级报告和项目状态，这些图表和报告元素与模型内容和状态紧密相关。

了解仪表板图表

哪里可以找到仪表板图表

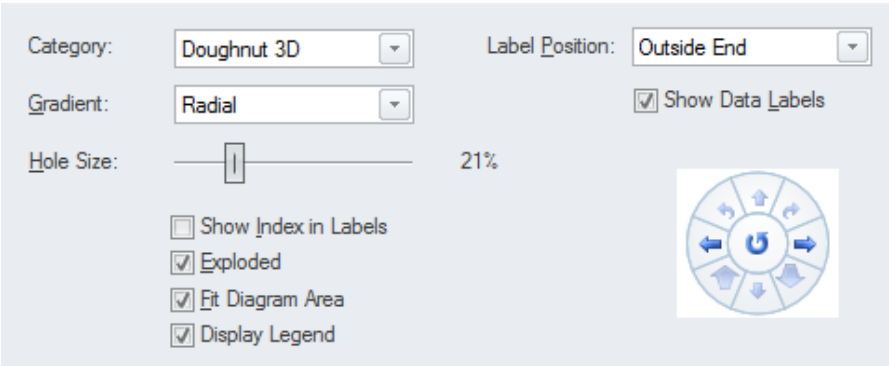
浏览器窗口 |右键单击包|**添加** 图表管理 |全部显示蓝图|扩展|仪表板

仪表板图表用途

仪表板图提供了丰富且易于理解的信息视图——例如系统特定版本中需求的状态——可以在模型中打开或方便地直接复制到管理或项目团队演示文稿中。它们对于规划迭代（**例如敏捷冲刺**）**很有用**，可以查看实施团队的需求准备情况；**例如**，查看有多少百分比的需求已被批准并具有高优先级。

仪表板图表选项

工具箱方式进行配置，包括更改源、应用过滤器或修改图表的图表，如图所示，可从图表的属性窗口使用' 外观"部分。



了解更多图表仪表板

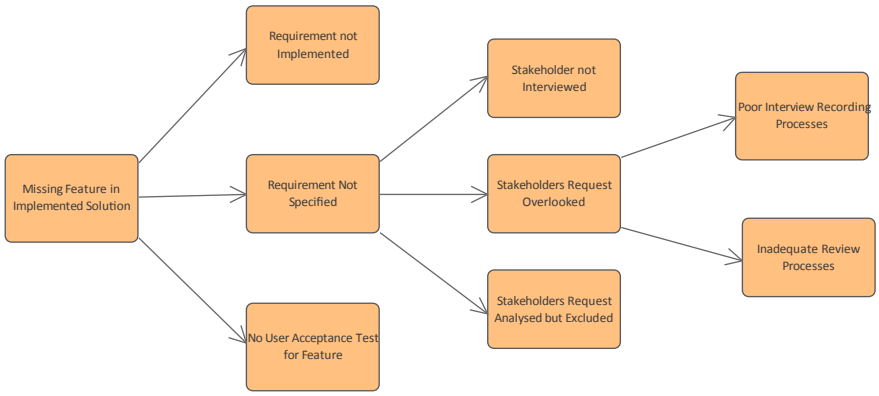
- Standard Charts

决策图表

了解决策图表

介绍决策图表

决策是一种以图形方式表示多个选项的有效方法，并提供了一种机制来调查选择这些选项的可能结果和好处。他们还可以帮助分析师形成与每个可能的行动方案相关的风险和收益的平衡图景。它们是决策表的近亲，但具有图形化的好处。 Enterprise Architect有一个专门构建的图表，允许对复杂的决策进行建模和显示，包括概率和不确定性。



在哪里可以找到决策图表

功能区：设计>图表>添加图表>**策略建模**>决策

浏览器window Toolbar : New图表>**策略建模**>决策

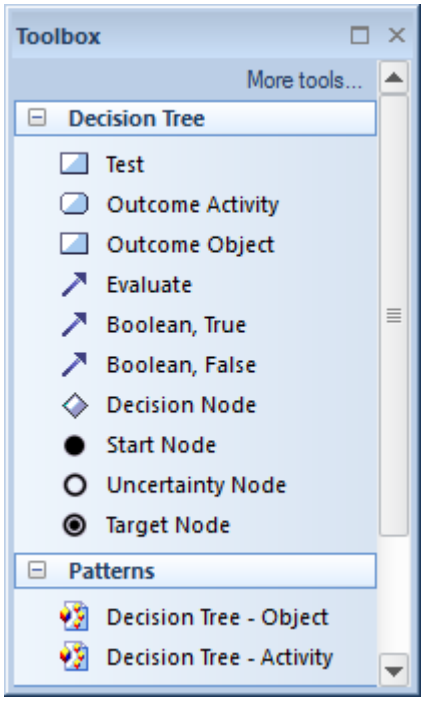
浏览器窗口上下文菜单 |添加图表... >**策略建模**>决策

用途决策的图表

决策可用于帮助决策过程，特别是当决策涉及具有不同发生可能性的一组复杂条件时。它们可用于战略或运营决策分析，并有助于确定决策的基础，特别是当采取的行动必须基于形式分析或产生代价高昂的后果时。决策A用于为更喜欢查看图表而不是库表和文档的利益相关者呈现决策决策表的图形图片。

决策图表选项

决策可以用不同级别的形式绘制，从具有一系列决策的简单树到导致结果的更正式的树，这些树涉及分配概率值的不确定性或具有输入参数的公式表达式。 “决策”工具箱页面包含一系列可使用的元素，以及可用于创建图表的两种模式，为分析师提供起点。



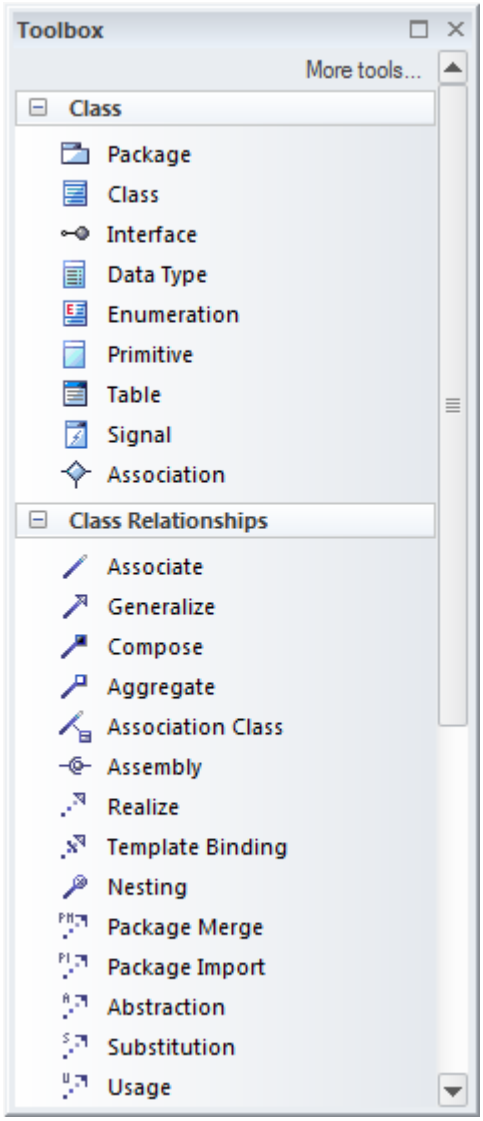
显示图表时也可以使用图表过滤器来吸引对图表部分的注意，并且可以通过更改图表的属性以手绘或白板样式呈现图表。

了解更多关于决策图表 Decision Tree

状态机图表

了解状态机图表

状态机图表介绍	状态机图是系统建模语言（SysML）行为图之一，可用于模型广泛的事物。它是用于对业务和技术领域中的实体进行一般的通用图表，包括术语和概念、业务规则以及 XML 和数据库模式中的功能。
在哪里可以找到状态机图表	功能区：设计>图表> 添加图表:图表系统工程1类型状态机required type> 浏览器窗口工具栏：图表系统工程1类型状态机<必需类型> 浏览器窗口上下文菜单|添加图表：图表系统工程1类型状态机<必需类型>
状态机图表的用途	只要需要系统的逻辑或结构表示，就可以使用类图。它适用于对业务和技术概念进行建模，并可用于模型信息和结构，例如 XML 和数据库模式。
状态机图表选项	类图（像任何图一样）可以被视为一个元素列表，这使得使用元素的属性更容易。 显示图表时也可以使用图表过滤器来吸引对图表部分的注意，并且可以通过更改图表的属性以手绘或白板样式呈现图表。



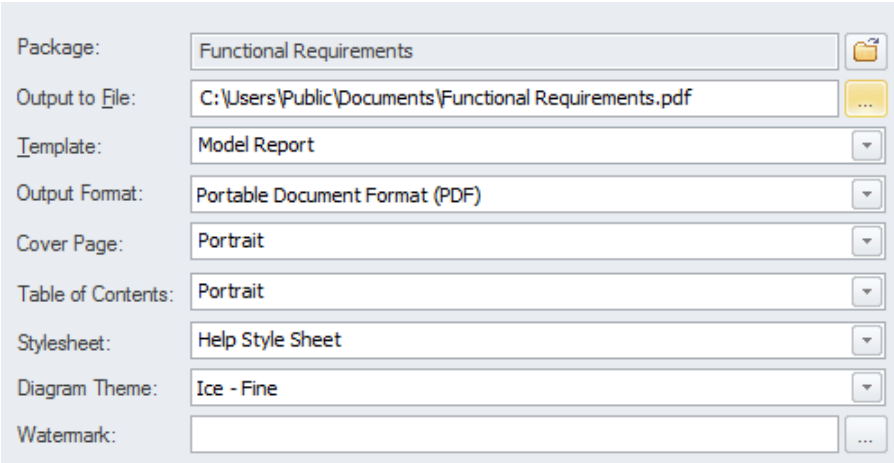
了解更多关于状态机图表 StateMachines

文档

了解文档

介绍文档

文档特征可用于直接从模型自动生成范围广泛的文档。这些可以是基于文档的，例如 PDF 和 Docx 格式或基于 HTML。灵活的模板可用于对生成的文档进行完全定制，包括公司logo、内容库表库表、元素信息库表和图表。也可以通过词汇表和搜索窗口等多种工具创建临时报告。



在哪里可以找到文档

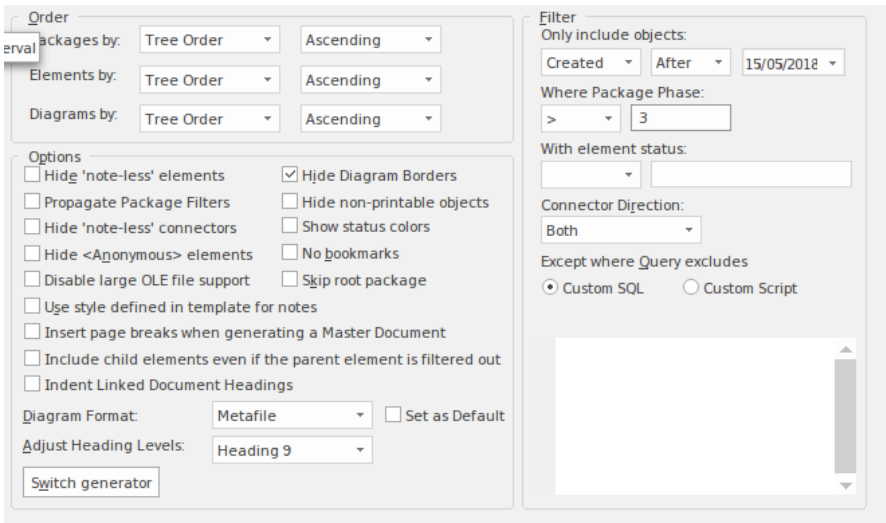
功能区：发布 >模型报告 > 报告生成器

使用文档

建模师、分析师、架构师、项目经理和其他人可以使用该功能生成范围广泛的基于文档的出版物和报告，例如系统需求规范、用例报告、数据字典、解决方案架构描述等。它还可以用于临时报告以创建报告，例如最不稳定的需求列表。还可以发布 HTML 文档，以允许无权访问Enterprise Architect的利益相关者从 Intranet 站点查看模型，这些模型可以放置在文件系统上，而无需网络服务器。

文档选项

可以设置几个选项来定制生成的文档中包含的信息，包括元素和图表的排序以及隐藏某些元素。也可以应用过滤器和单词替换以及其他选项。



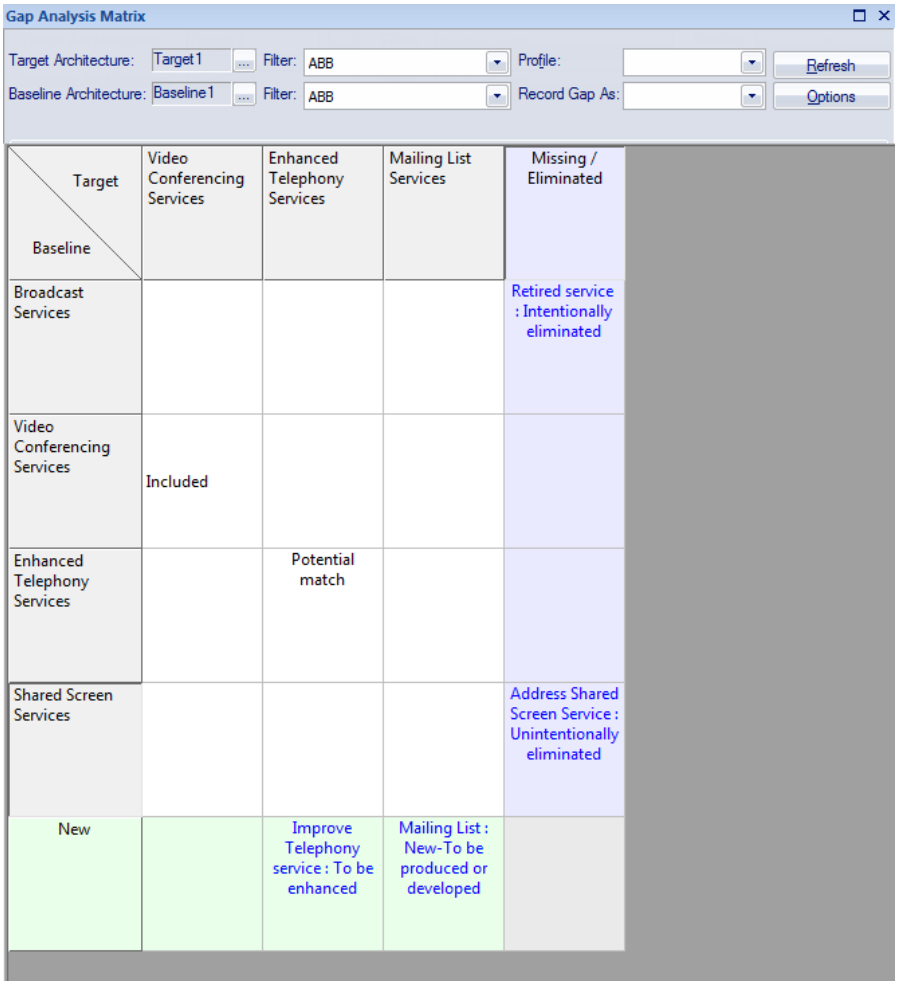
了解更多about Documentation	Model Publishing

差距分析矩阵

了解差距分析矩阵

引入差距分析矩阵

差距分析矩阵是一种专门的关系矩阵，用于记录企业某些部分的两个版本之间存在的差距。可以记录两个不同版本的架构之间的差距，或者两个版本的能力或员工能力之间的差距，或者两个版本的信息或数据之间的差距。该工具的结构类似于具有列和行的电子表格。构成基线（起点）的元素被列为行，而构成目标（终点）的元素被列为列。有一列用于记录丢失或消除的元素，有一列用于记录新元素。在基线元素和目标元素的交叉处，可以添加注记来描述两个元素之间关系的任何细节。



哪里可以找到差距分析矩阵

功能区：设计>包>间隙分析

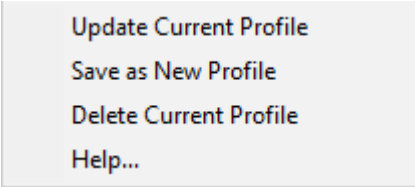
差距分析矩阵的用途

差距分析矩阵既可用于业务分析，也可用于技术分析。它是一种用于记录企业某些部分的不同版本之间比较一般的通用工具。在业务分析中，它可用于分析员工能力、数据和信息、业务流程、业务线等，并在企业当前和未来状态下进行比较。在企业架构中，矩阵可用于记录基线、过渡和目标架构之间的差距，比较能力、架构和解决方案组件等。

差距分析矩阵的选项

差距分析矩阵可以配置为显示存储库的不同部分。一旦为目标基线器选择了间隙的包并确定了合适的包，就可以选择元素类型的元素类型。为间隙选择

的元素将限制可用元素以表示矩阵中单元中 缺失"或 新"元素的间隙。 选项"菜单中有许多可用选项，包括能够更新、删除和保存差距分析矩阵作为**配置文件**，给它一个名称以便以后可以调用它。



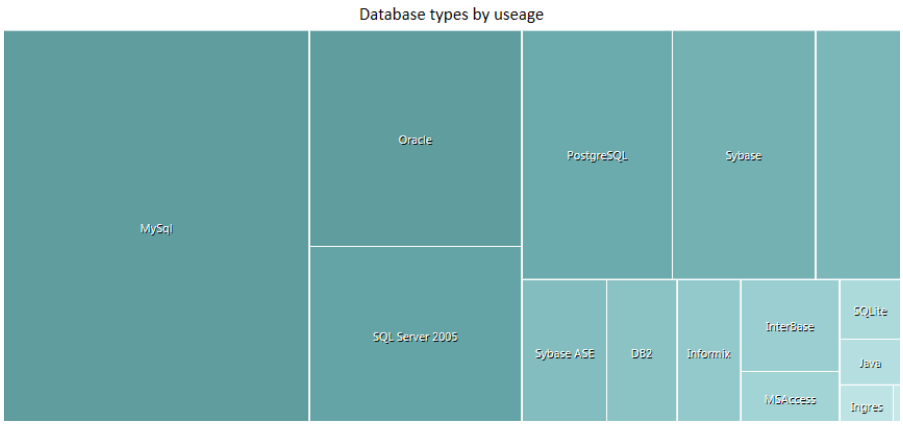
了解更多关于差距分析矩阵 Gap Analysis Matrix

热图

了解热图

介绍热图

热图是A图表，可用于在二维中可视化数据。它使用矩形的颜色来表示数据的维度，使用矩形的相对大小来表示另一个维度。它们通常用于为战略或战术决策创建引人注目的数据表示。它们可以用于从战略架构到技术架构的任何级别的存储库。



在哪里可以找到热图

双击图表元素图表源包

热图的用途

热图通常用于为战略或战术决策创建引人注目的数据表示。它们可以与需求一起使用来指示一组需求的状态，如果指标可用，则可以指示每个需求的估计实施成本。它们可以与应用程序或技术清单一起使用，以显示技术的流行程度。例如，哪些应用程序是用特定语言开发的或在特定运行上运行的。

热图选项

作为在 包"选项卡上的字段中指定热图参数的替代方法，您可以选择 自定义 SQL "选项卡并使用#

创建自定义热图#

.您仍然在 类型"字段中指定图表类型，但其他对话框字段显示为灰色。

了解更多关于热图

Heat Maps

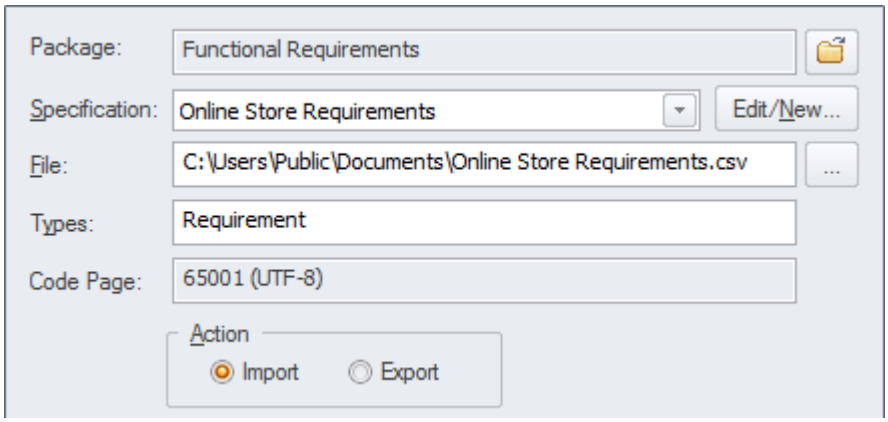
导入和导出电子表格

导入和导出电子表格

导入和导出电子表格

此功能是将电子表格或文字处理器表中定义的需求导入Enterprise Architect的有用机制。一旦进入Enterprise Architect ，需求就可以被管理和跟踪到诸如业务驱动程序和场景和组件之类的元素。或者，可以将Enterprise Architect中的需求导出到电子表格中，以便将其提供给第三方或用于某种类型的数值或统计分析。电子表格中的字段与Enterprise Architect中的类似属性之间的映射完全可以使用规范进行配置。

如需更详细的信息交换，Microsoft Office 的MDG Link（可从Sparx Systems获得）提供了在处理复杂需求时有用的附加功能和集成点。



在哪里可以找到导入和导出电子表格

功能区：发布 >模型交换 > CSV

使用导入和导出电子表格

此特征可用于从 CSV 文件导入或导出需求。在安装Enterprise Architect等工具之前，分析师可能已经在他们最喜欢的文字处理器中使用电子表格或表来记录需求；这些可以使用 CSV 导入功能方便地导入。或者，有时必须将需求提供给第三方，他们通常会在电子表格文件中指定他们想要它们；这可以使用导出功能来实现。

导入和导出电子表格的选项

导入和导出功能是完全可配置的，并具有用户定义的规范，以方便将电子表格字段映射到Enterprise Architect中的需求属性。该功能还包括在需求的标记值中导入和导出字段的能力。

了解更多关于导入和导出电子表格

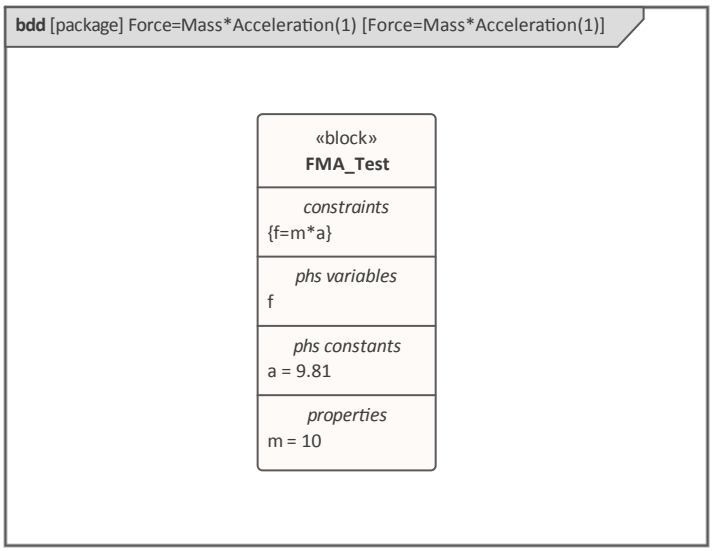
CSV Import and Export

参数图表

了解参数化图表

介绍参数图表

参数图是系统建模语言（SysML）结构图之一，可用于模型系统的数学方程。有一个助手可以帮助将方程转换为包括属性在内的建模元素。一旦对方程进行建模，它们就可以用于使用 OpenModelica 集成创建模拟。



在哪里可以找到参数图表

功能区：设计>图表> 添加图表:图表=系统工程> SysML, Select From = SysML 1 .5,类型= Parametric

浏览器窗口工具栏：图表=系统工程> SysML，Select From = SysML 1 .5，类型= Parametric

浏览器窗口上下文菜单|添加图表：图表=系统工程> SysML，Select From = SysML 1 .5，类型= Parametric

参数图表的用途

每当您需要可视化或模拟方程组时，都可以使用参数图。

参数图表的选项

参数图有许多选项，包括简单方程或方程组的建模。

了解更多关于参数化图表

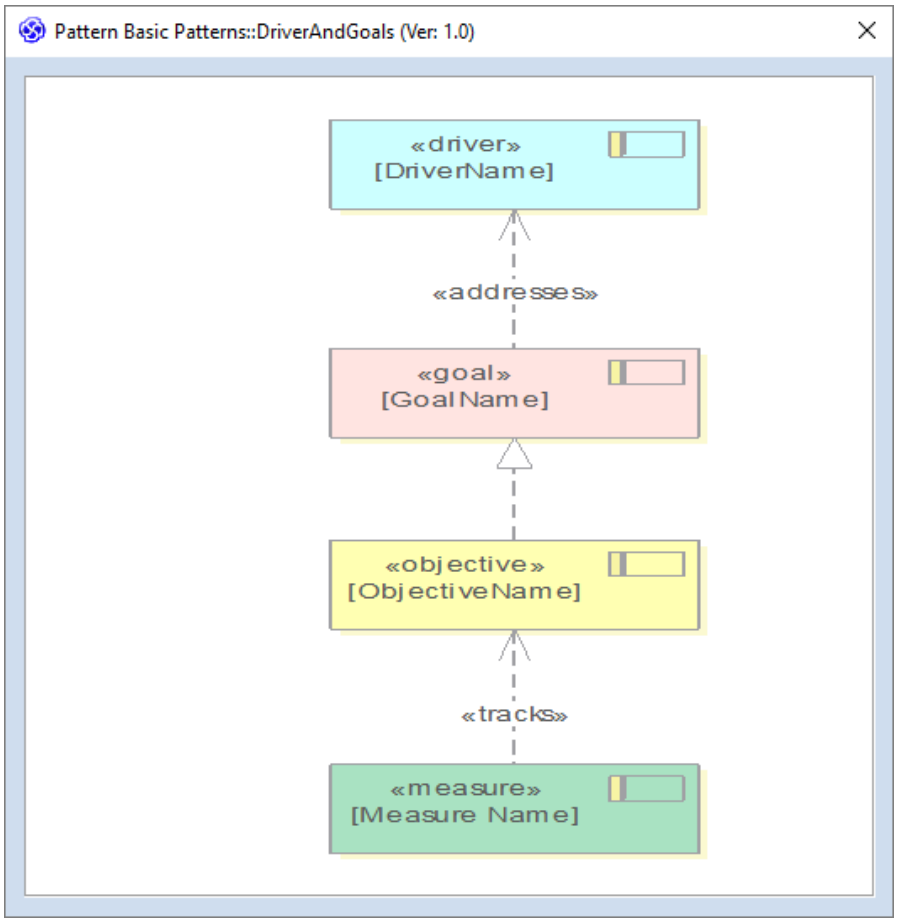
- Parametric Diagrams
- Class Diagram

模式

认识模式

介绍模式

模式是针对给定架构上下文中常见问题A一般可重用设计解决方案。模式不是已解决的设计，而是如何解决问题的模板。这个概念起源于建筑架构世界，最初发表在克里斯托弗·亚历山大（Christopher Alexander）的一本名为“设计模式”的书中。然　它们被应用于软件行业，并被软件工程领域广泛用于解决经常出现的软件工程问题，尽管从表面上看问题的性质似乎完全不同。



在哪里可以找到模式

创建模式：

特定>技术> 发布技术> 发布图表模式

使用模式：

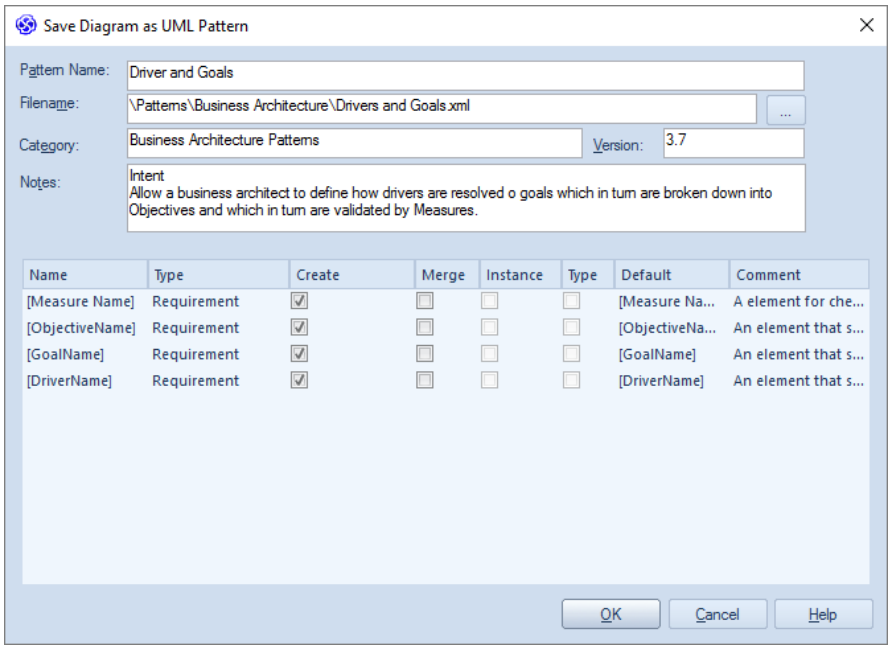
开始>应用程序> **共享**>资源>**模式**<模式组>> 右键单击模式名称> 将模式添加到图表

模式的用途

模式可以应用于从业务到技术架构的各种情况，但始终用于将通用解决方案应用于表面上可能看起来完全不同的任何数量的问题或上下文。 Enterprise Architect为挖掘模式提供了一种有用的机制，这意味着任何图表都可以作为模式发布，然后在相同或不同的上下文。模式及其用法的一个示例可能是阐明驱动因素、目标、目标和措施之间关系的模式。现有图表可以作为模式发布，然后任何业务架构都可以通过简单地将其拖到空图表上来重用该模式。

模式选项

模式最常见于浏览器窗口的 资源"选项卡，但有时也内置于技术中并从工具箱页面提供。发布模式时有许多可用选项，包括描述模式整体细节的能力，以及为构成模式的每个元素添加注记的能力。



关系矩阵

认识关系矩阵

介绍关系矩阵

关系矩阵提供了一个可视化的矩阵样式视图，便于分析需求相互之间以及模型中其他元素的关联方式。可用于查看Stakeholders与需求的关系、使用案例与需求或功能需求的关系、Capabilities与业务驱动因素业务的关系、哪些Component实现了一组需求等。可以快速定义任意数量的矩阵，然后将其保存以供车间查看，或包含在从模型自动生成的文档中或导出到电子表格文件中。创建矩阵时，可以通过将需求放在矩阵的一个轴上，将连接的元素放在另一个轴上来查看连接，然后矩阵的单元将指示关系的方向。

Target / Source	REQ011 - Manage User Accounts	REQ012 - Provide Online Sales	REQ013 - Manage Deliveries	REQ014 - ShoppingBasket	REQ015 - Process Credit Card Payment	REQ016 - Add Users	REQ017 - Remove User	REQ018 - Report on User Account	REQ019 - Manage Inventory	REQ020 - Receive Books	REQ021 - List Stock Levels	REQ022 - Order Books
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

在哪里可以找到关系矩阵

在浏览器窗口中，单击一个包并选择：

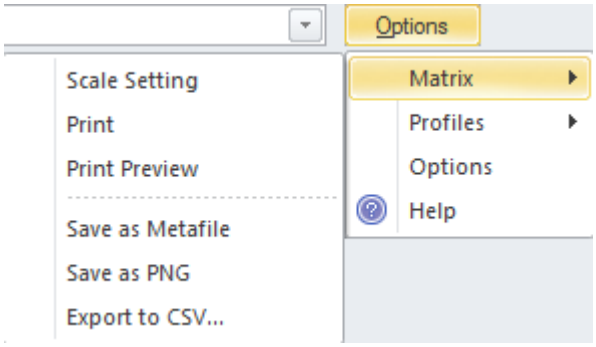
- '资源' 标签 |矩阵Profiles |右键单击配置文件 |打开矩阵配置文件或
- 开始功能区 >**全部窗口**>设计> **工具** >包矩阵

关系矩阵的用途

在视觉上引人注目的矩阵中显示两个包中元素之间存在的关系- 例如哪些需求由哪些使用案例实现。它有助于分析缺失的元素或关系；**例如**，确定哪些需求没有被任何用例实现，或者哪些组件没有相应的需求或使用案例。它在与可能不熟悉在跟踪图中看到需求的业务利益相关者的研讨会中特别有用。

关系矩阵的选项

可以为关系矩阵设置一系列选项，包括将其保存到浏览器窗口的 资源”选项卡或保存为 CSV 格式以便在电子表格中打开。关系矩阵的外观也可以通过对元素进行排序、显示轮廓编号视图和隐藏包名来进行更改。这些项目可从关系矩阵上的选项按钮获得。



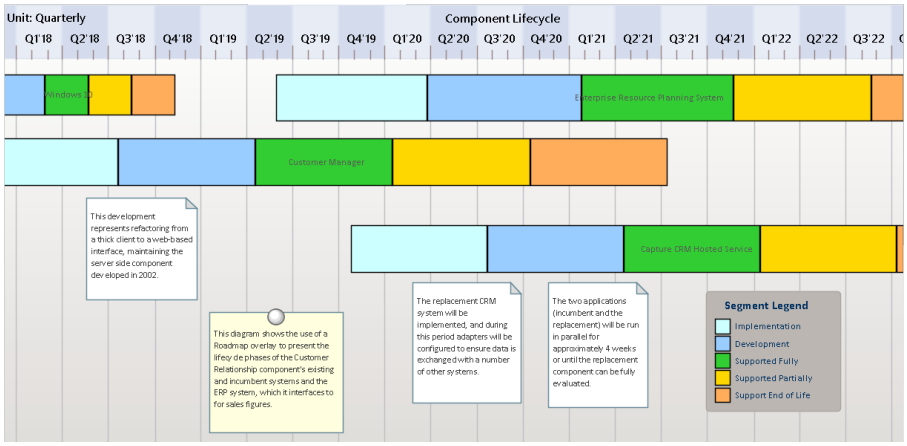
了解更多关于关系矩阵 Relationship Matrix

路线图图表

了解路线图图表

介绍路线图图表

路线图图是一种叠加层，可应用于任何图表，以描述元素中的重要阶段以及它们如何随时间变化。



图表上可以出现的元素类型没有限制，任何图表都可以定义路线图覆盖。元素生命周期中重要的用户定义阶段由彩色条表示，可以设置为显示持续时间。颜色和阶段可以使用图表图例进行配置，它会自动将它们应用于图表中的元素。它们在描述能力和应用程序路线图的企业架构图中特别有用。

在哪里可以找到路线图图表

功能区：**布局**>图表>路线图

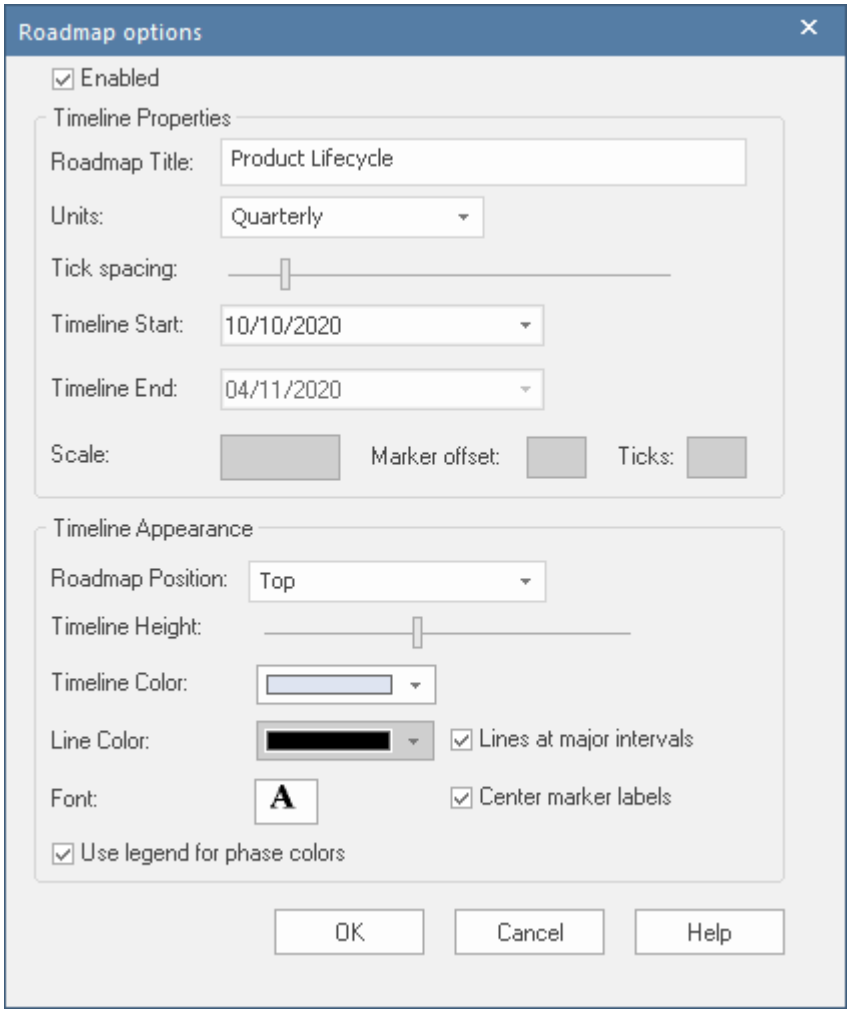
图表上下文菜单：路线图

路线图图表的用途

路线图在企业架构中有广泛的用途，可用于显示系统工程的应用程序和功能路线图，用于显示低级组件的时序。

路线图图表的选项

路线图覆盖有一系列选项来确定时间线的属性，例如时间标尺的比例、单位、它们的位置，以及时间线的外观，包括字体和颜色。时间线的高度和位置也可以配置为适合图表和显示。



图表图例可以配置为定义元素生命周期中的阶段，设置彩色带的规范等等。路线图段可以显示或隐藏在单个元素上，以防特定段可能不适用于某个元素或更多图表上的元素。

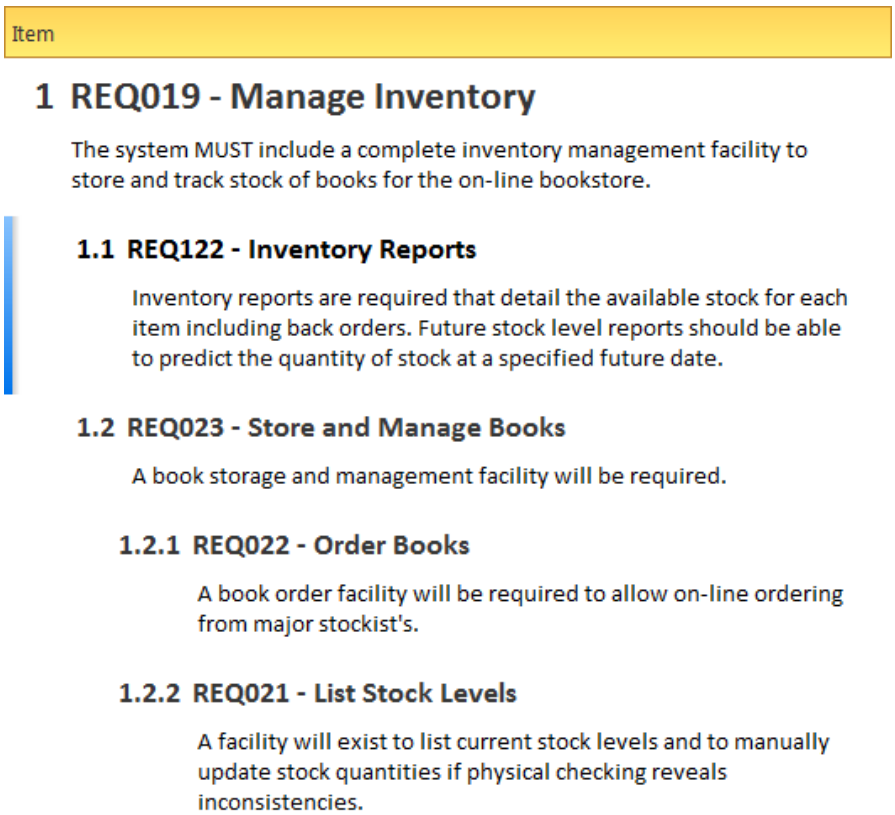
了解更多关于路线图图表 Roadmap Diagrams

规范管理器

了解规范管理器

规范管理器介绍

规范管理器是处理需求的中心工具；它提供了一个类似于文字处理器或电子表格工具的界面，用于输入、维护和查看需求。可以使用名称和详细描述创建新的需求，并且可以从下拉列表中添加状态和优先级等属性。现有的需求可以方便地查看和管理，在规范管理器中更改它们将在存储库中的所有其他位置更改它们，例如图表和窗口。对于那些更习惯于使用文本而不是图表以及习惯于在文字处理器或电子表格中工作的分析师来说，它是一个完美的工具。它的另一个优点是需求是模型的一部分，并且可以追溯到其他元素，包括驱动业务驱动因素、利益相关者和解决方案组件。



在哪里可以找到规范管理器

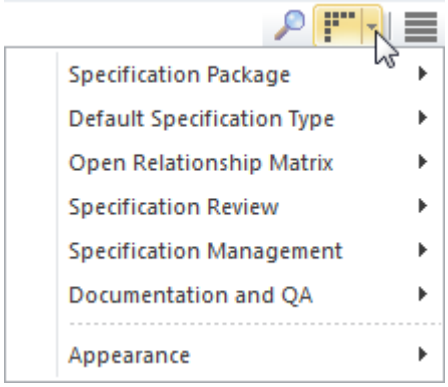
浏览器窗口 |右键包|规范管理器

规范管理器的用途

在类似于在文字处理器或电子表格中工作的基于文本的工具中创建、查看和维护需求。可以将详细信息添加到需求中，需求可以从下拉列表中添加属性。当需求在规范管理器中发生变化时，变化会方便地反映在浏览器窗口和所有其他窗口中。

规范管理器的选项

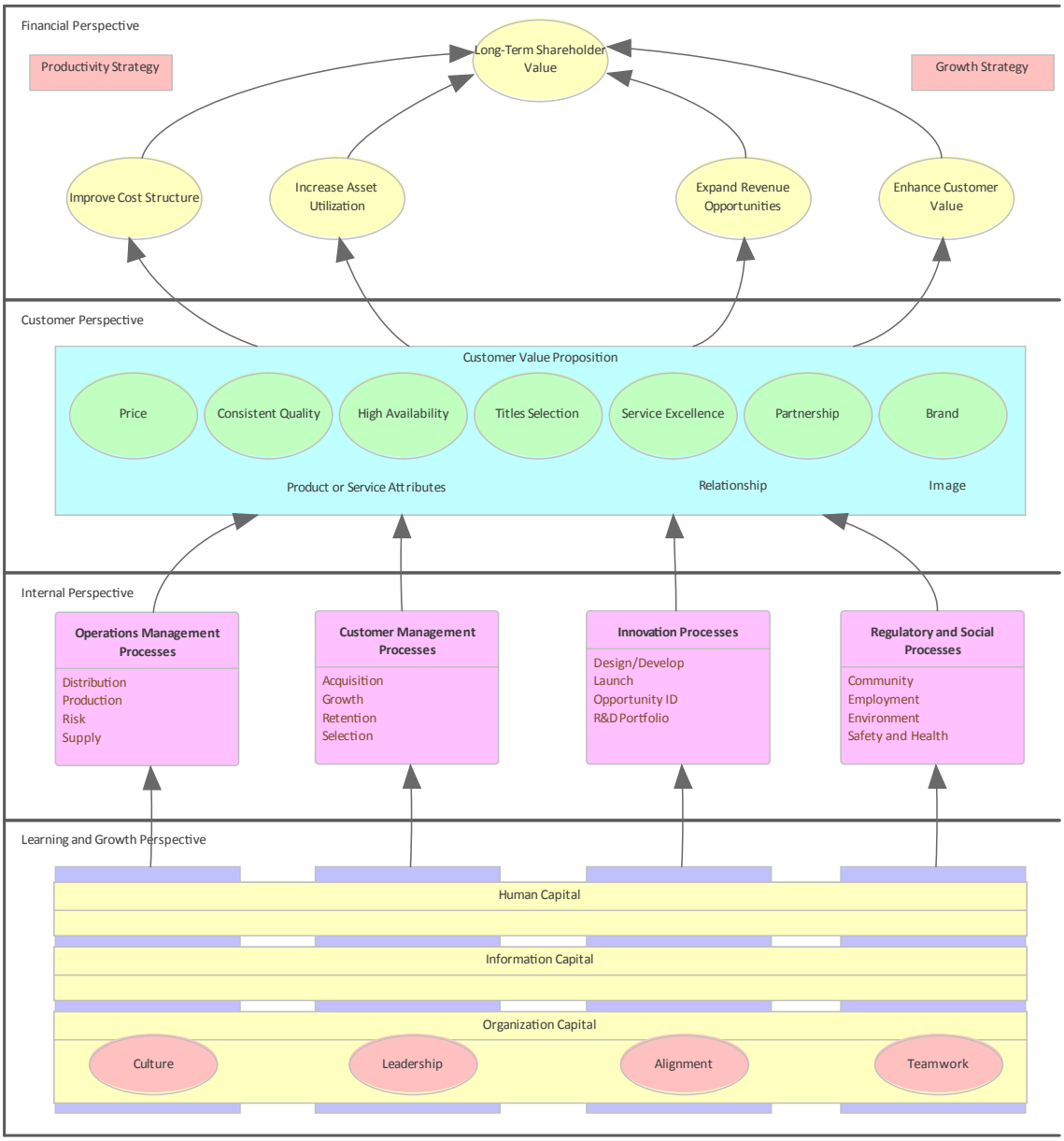
选项菜单中有多种选项可供选择，以定制您使用规范管理器的方式。其中包括级别（分层）编号、自动命名、拼写选择、文档、导入和导出需求、访问各种相关工具等等。



了解更多关于规范管理器 The Specification Manager

策略地图

A策略地图是用于描述对组织或业务团队重要的主要战略目标的图表。该图显示了四个重要的观点，它们是提供战略定义的重要问题。定义的视角是： 财务”、 客户”、 内部业务流程”和 学习 成长”。该图表用作一种沟通工具，以确保对战略达成共识，聚焦组织工作并协助评估进度。



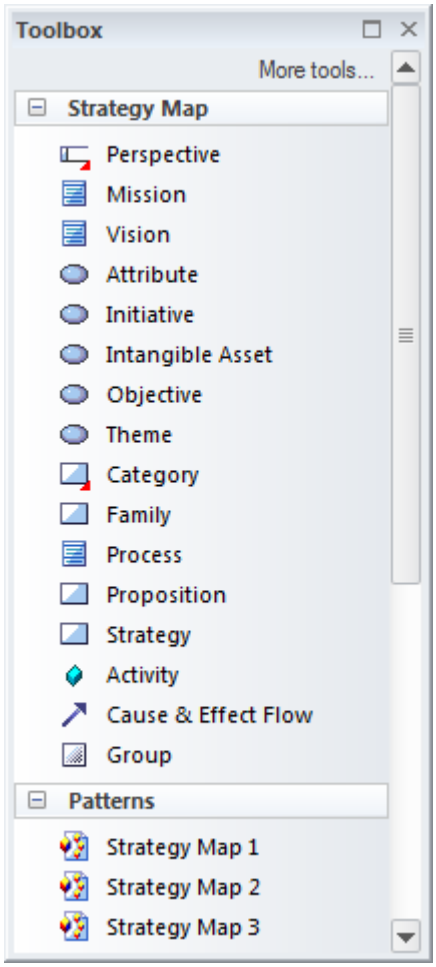
了解策略地图

哪里可以找到策略地图	功能区：设计>图表>添加图表>**策略建模**>策略地图 工具窗口工具栏：新图表图标 >策略建模浏览器模型 >策略地图 浏览器窗口上下文菜单 |添加图表... >**策略建模**>策略地图
用途策略地图的用途	战略策略地图用于模型组织或管理团队打算实现的关键战略目标。四个视角中的每一个元素都可以链接到存储库中的其他元素，以显示它们如何在业

务、应用程序或技术水平。

策略地图选项

可以A自动创建元素和图表的模式来创建策略地图，该模式可以用作策略地图的起点。共有三种模式可用，从在每个透视图中使用单个元素的非常简单的表达式到在每个透视图中使用多个元素的完整表达式。工具箱A了一系列附加元素和关系来扩展使用该模式创建的基本地图。



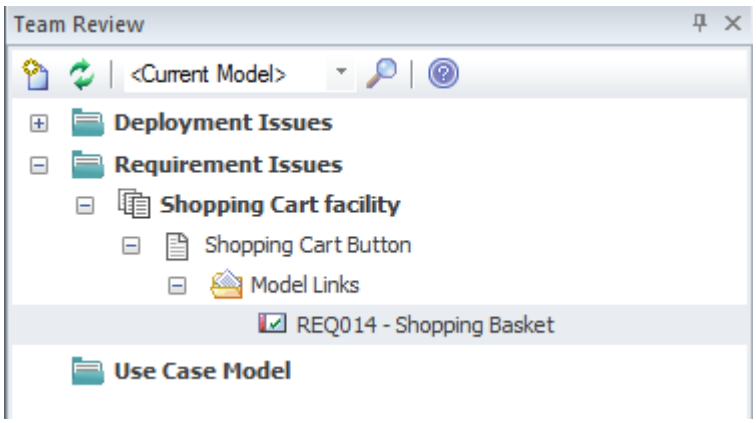
了解更多关于策略地图 Strategy Maps

图书馆

认识图书馆

介绍图书馆

库”窗口为开发人员、建模人员、客户和利益相关者提供了一个机会，可以对正在进行的工作或里程碑或项目的完成发表评论并提供反馈。



图书馆的用途

库特征可用于从任意数量的角度进行模型审查，包括演练、正式模型审查或临时审查。

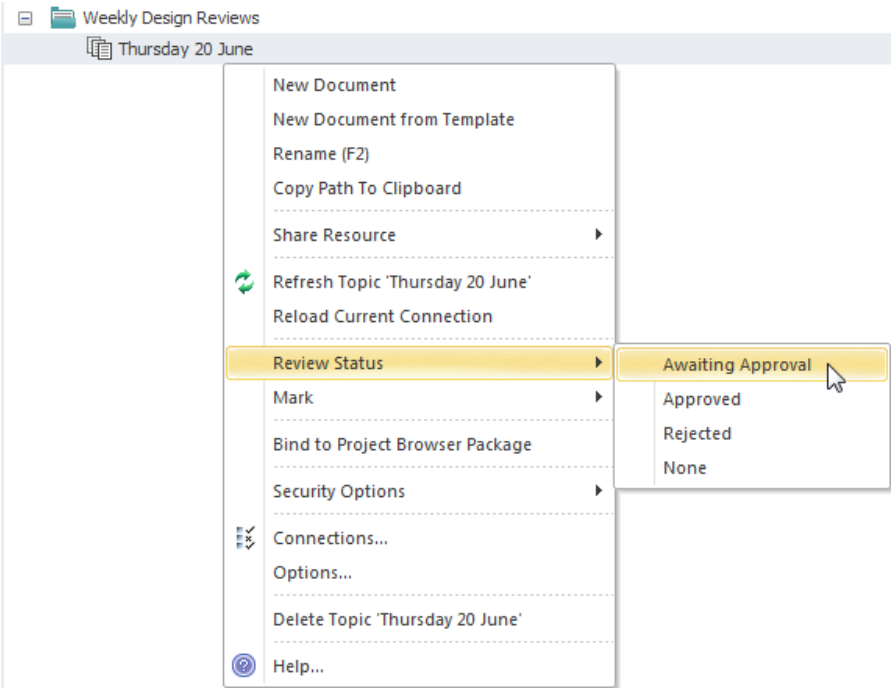
在哪里可以找到图书馆

发布或查看元素的讨论

功能区：开始>协作>**模型**库

图书馆的选项

有多种设置可用于配置库，可从类别和主题上下文菜单中获得，包括设置类别或主题的状态和其他选项。图表、元素和元素特征可以方便地从浏览器中拖动窗口来创建模型链接，团队成员可以使用这些链接直接从库窗口超链接到浏览器窗口中的这些项目。



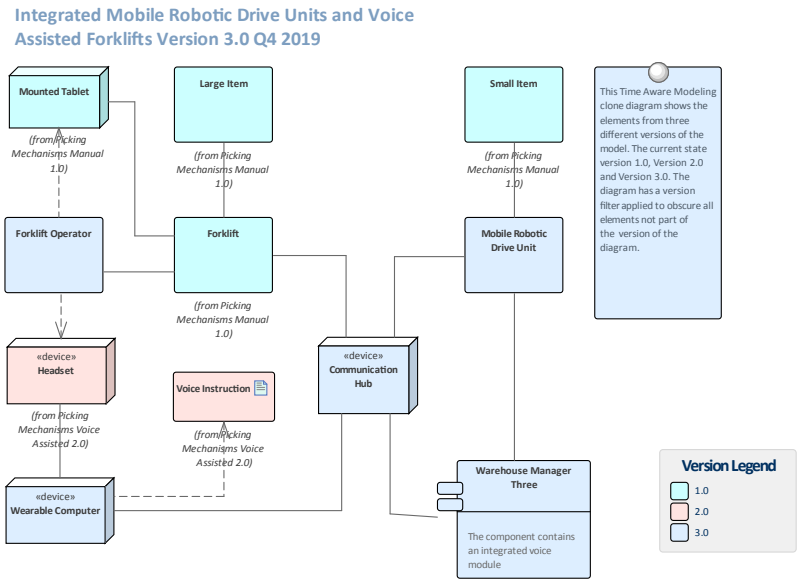
了解更多关于团队图书馆 The Model Library

时间感知建模

认识时间感知建模

介绍时间感知建模

时间感知建模模型特征允许分析师、架构师和其他人通过提供促进迁移的工具来创建他们的模型的增量版本，图表和时间包允许通过多个转换或版本创建。基线（'As-is'，当前状态）模型不受影响，并且可以构建任意数量的目标（'To-Be'，未来状态）模型以用于可视化和分析。当需要有效地表示和比较许多未来选项以进行“假设”分析时，它特别有用。时间是架构中最重要的维度之一，因为它是发生所有变化的基础。架构师意识到它的重要性，并且传统上创建了具有重复元素的模型；时间感知建模允许克隆现有元素、图表和包。



哪里可以找到时间感知建模

克隆包结构为新版本

功能区：设计>包>管理>克隆结构为新版本

浏览器window版本上下文菜单：Clone Structure as New

克隆图表作为新版本

功能版本功能区：设计>图表>管理>克隆为新

将元素克隆为新版本

功能区：设计>元素>管理>克隆元素为新版本

时间感知建模的用途

时间感知建模模型和技术模型可以应用在广泛的场景中；它提供了一种显示模型所代表的实体如何随时间变化的方法。基线（'As-Is'，当前状态）模型可以不受影响，而可以构建任意数量的目标（'To-Be'，未来状态）模型以反映基线模型随时间的可能演变。分析师或架构师会在许多领域发现此工具很有用。例如：

- 架构师使用模型来记录当前状态，然后继续执行更具挑战性和更有价值的任务，即定义未来状态的外观，通常是在一系列过渡中
- 当获得业务分析师，通常会遇到挑战，描述新的基础产品的不同版本应

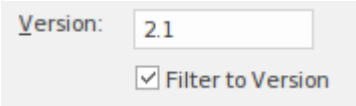
该如何为这些不同的客户群配置，从而导致同一产品的不同版本

- 业务战略家通常规定能力模型将如何寻找未来的组织，从而产生至少两个版本的能力模型
- 在合并和收购过程中，业务架构师负责描述收购完成后企业的可能状态
- 工程师需要开发更好、更高效的解决方案来应对未来的挑战，因此需要定义更新版本的解决方案
- 测试人员在设计和运行测试用例时需要了解产品的不同版本
- 基础架构工程师需要定义未来的环境以响应性能或安全问题，创建多个版本的服务器、设备甚至整个功能

所有情况都需要将时间纳入模型中，以便对其进行推理和明确。 Enterprise Architect的时间感知建模功能可用于这些情况，以确保将时间作为一等公民包含在模型中。时间不是以绝对或相对的方式测量或建模的，而是通过表示任意数量的未来状态或版本形式的差异。

时间感知建模选项

时间感知建模模型和元素特征允许建模者包、图表。大多数模型都不是微不足道的， Enterprise Architect提供了广泛的工具来帮助可视化模型以及它们如何随时间变化。可追溯性窗口对于查看时间感知模型中的元素与存储库的其他部分之间的连接特别有用。 A非常有用的特征是能够根据版本对图表应用过滤器，从而掩盖不属于特定版本的元素。



了解更多关于时间感知建模

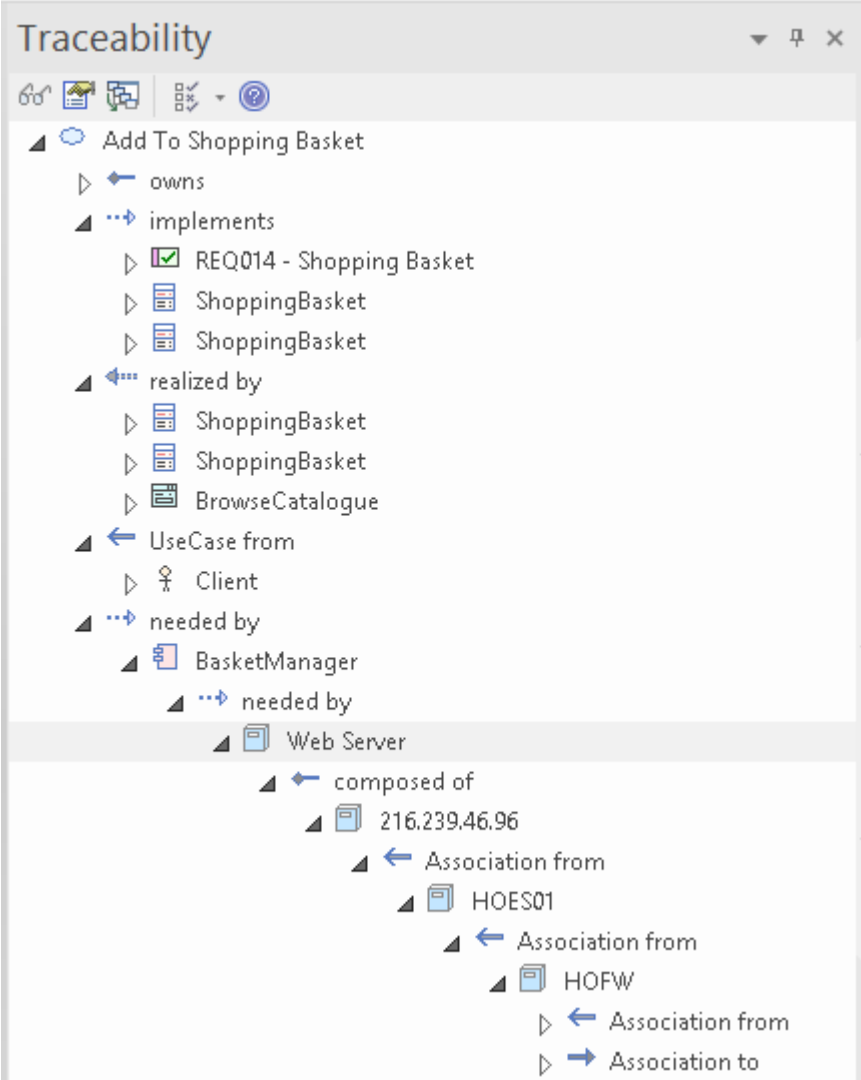
Time Aware Models

可追溯性窗口

了解可追溯性窗口

介绍可追溯性窗口

可追溯性窗口提供了元素连接的层次视图，允许追溯性作为模型中遍历的元素进行可视化和查询。这个工具特别有用，因为建模者经常会选择隐藏图表关系，但是通过在图表中选择一个元素并在可追溯性窗口中查看它的连接，它的所有关系都会被显示出来。



在哪里可以找到可追溯性窗口

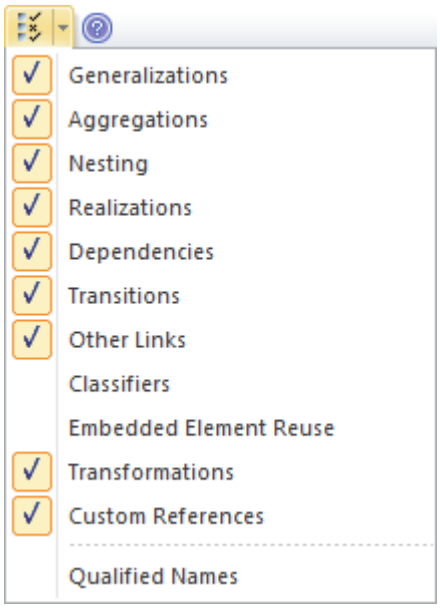
开始>应用>设计可追溯性

用途窗口的可追溯性

元素关系的可追溯性窗口提供了与存储库中其他元素的连接方式的分层视图，以及每个元素的类型。此窗口提供了所有关系的完成列表，这些关系无法通过查看浏览器窗口中的元素看到，也可能不会出现在任何图表中。这对于管理需求业务和跟踪需求与驱动因素因素等上游流程元素和组件等下游流程元素的关系非常有用。这是一个有用的工具，使模型新手能够快速了解哪些是重要且连接良好的元素。在你删除一个元素之前，你应该使用你现有的模型可追溯性窗口来确保该元素的现有关系。

可追溯性窗口的选项

有一系列选项将可追溯性限制为指定的连接器类型；可以设置这些选项来改变窗口中显示的内容。这些选项可从窗口顶部的工具栏中获得。



了解更多关于可追溯性窗口

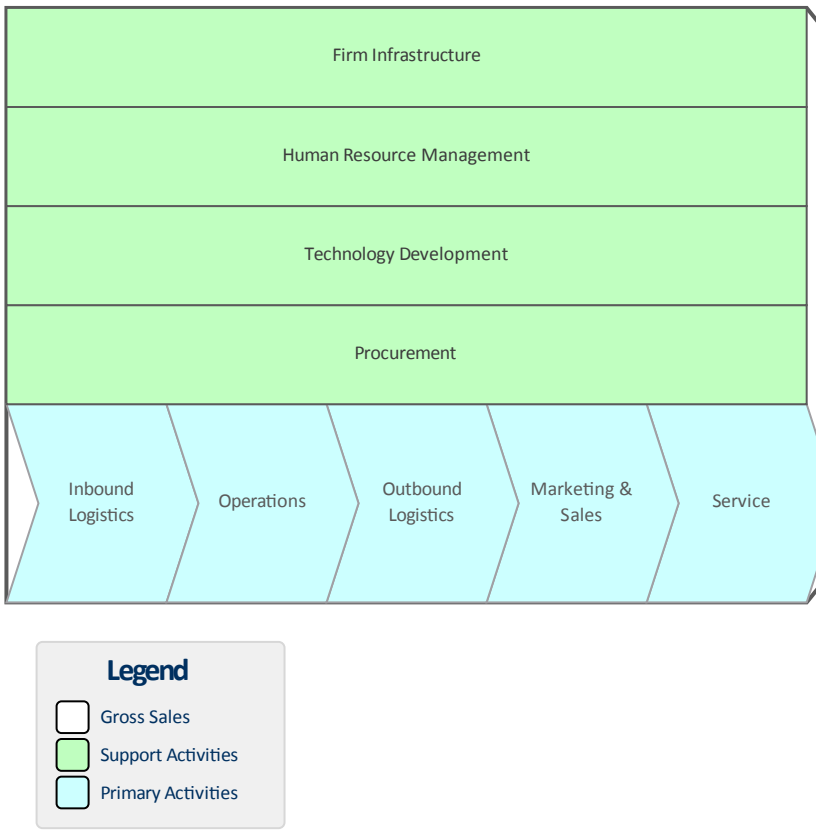
The Traceability Window

价值链

了解价值链

引入价值链

价值链是一个战略图，它允许对组织中的主要和次要活动进行建模。该图可以从一个模式创建，该模式添加并连接链中的五个主要活动以及支撑它们的四个支持活动。



总销售额元素充当主要活动和支持活动元素的框架。

在战略业务单位级别工作的分析员经常被要求模型业务单位为为其客户提供价值而执行的活动。价值链是创建组织执行活动序列的战略表示的首选工具。

在哪里找到价值链

功能区：设计>图表>添加图表>**策略建模**>价值链

浏览器window Toolbar : New图表>策略建模浏览> 价值链

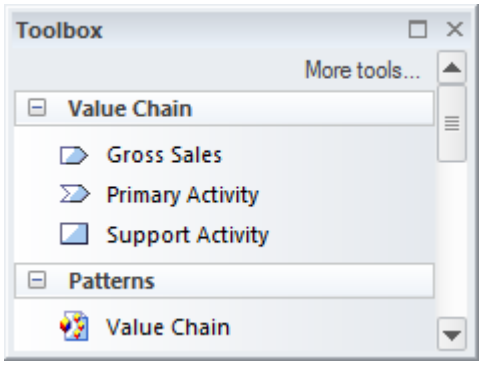
浏览器窗口上下文菜单 |添加图表... >**策略建模**> 价值链

价值链的用途

价值链是协助战略规划的重要工具，可以让人们了解整个序列（或链）。它还允许将链条分解为其组成活动，从而确定并可能改进对成本、资源和价值的评估。

价值链的选择

每个主要活动和支持活动都可以链接到模型中的其他元素，包括链接文档和定义基准的元素。



价值链图（像任何图一样）可以被视为一个元素列表，它使得使用元素的属性更容易。

在展示图表时，也可以使用图表过滤器来吸引对图表部分的注意。

了解更多关于价值链 Value Chains