



**ENTERPRISE ARCHITECT**

Série de Guides d'Utilisateur

# Guide de MBSE avec SysML

Author: Sparx Systems

Date: 7/11/2024

Version: 17.0

CRÉÉ AVEC  **ENTERPRISE  
ARCHITECT**

# Table des Matières

Guide de MBSE avec SysML	7
Une équation à quatre variables	9
La méthode ou le processus d'ingénierie	10
Modélisation comme discipline	13
Démarrage	15
Définir l'objectif d'un Modèle	17
Décider où Démarrer	19
Pièces de connexion du Modèle	20
Assurer la qualité d'un Modèle	21
Le Systems Modeling Language (SysML)	22
Enterprise Architect l'outil Modélisation	23
Plateforme de collaboration	26
Gestion de Projet Établi	27
Modèle Référentiel	28
Démarrage	29
Configuration d'une structure Modèle	31
Adaptation de l'application pour MBSE	32
Définir une perspective	33
Sélection d'un style visuel	35
Sélection d'un espace de travail	36
Définition des préférences	37
Importation de matériel existant	38
Création Diagrammes d'éléments et Relations	42
Visualisation des modèles	47
Synchronisation avec des données externes	49
Où nous allons	50
Découvrir les Diagrammes SysML	55
Aspects communs des Diagrammes	63
Interrompre lorsqu'une Variable Change de Valeur	73
Diagramme des exigences	76
Diagramme de cas d'utilisation	79
Diagramme Paquetage	81
Diagrammes d'activité	83
Diagramme Interne de Bloc	85
Diagramme Paramétriques	87
Diagramme de Séquence	89
Diagramme Statemachine	92
Présentation Langage Modélisation des Systèmes	94
Architecture du langage	96
Concepts grammaticaux clés	103
Modèles, Diagrammes , éléments et Vues	105
Collaborer en tant qu'équipe d'ingénierie	110
Référentiel partagé central	111
Cloud Computing	112
Discussions et chat	114
Ressources et calendriers Kanban	115
Révision de Modèles	116

Partager des ressources dans la Bibliothèque d'Équipe	119
Affichage des modèles sur les appareils mobiles	120
Modélisation de l' Avenir	121
Contrôle de Version et Lignes de base	125
Serveur d'actifs réutilisables	127
Utiliser Paquetages pour Structurer le Référentiel	130
The Function of Packages	132
Présentation Diagrammes Paquetage	133
Régimes d'organisation Paquetage	138
La fenêtre Navigateur	141
Accéder au Référentiel à l'aide de Modèle Vues	144
Définition et gestion des besoins	145
Exigences des citoyens de première classe	150
Présentation Diagrammes d'exigences	154
Développer Exigences	158
Élicitation	160
Sources des documents	161
Observations des utilisateurs	162
Séminaires des parties prenantes	163
Créer Exigences	166
Exigences externes et internes	167
Catégories d'exigences	168
Propriétés requises	170
Spécification	172
Rencontrez le Gestionnaire de Spécification	173
Analyse	176
Prioriser les Exigences	177
Validation	181
Visualiser Exigences	183
Diagrammes Exigences	184
Gestionnaire de Spécification	185
Navigateurs et Vues	186
Matrice de relations	188
Exigences Tableaux	189
Gérer Exigences	190
Traçage des Exigences	191
Suivi Exigences	194
Gérer Exigences changeantes	196
Analyse d'impact des changements	199
Exigence Volatilité	201
Réutilisation des exigences	203
Relations d'exigences	205
Ajout de raffinement à une exigence	208
Relation de confinement	209
Copier Exigences existantes	211
Dérivation d'une exigence à partir d'une autre	212
S'assurer qu'une exigence est satisfaite	213
Traçabilité des éléments Modèle	215
Vérifier la relation	216
Visualisation Relations entre exigences	218
Documenter Exigences	224

Glossaire du Projet	225
Spécification logicielle requise	226
Décrire les objectifs de l'utilisateur avec des cas d'utilisation	227
Exigences et cas d'utilisation	228
Présentation Diagrammes de cas d'utilisation	231
Découvrez le créateur de scénarios	235
Structurer un Modèle de cas d'utilisation	237
Générer Diagrammes de comportement	238
Rapport de cas d'utilisation	239
Utilisation de blocs pour Modèle la structure et les contraintes	241
Démarrage avec des blocs	246
Modélisation des contraintes sous forme de blocs	249
Présentation Diagrammes de définition Bloc	250
Les éléments structurels fondamentaux	256
Modélisation Fonctionnalités Structurelles	257
Modélisation Comportementale Fonctionnalités	263
Autres Relations Bloc	269
Points d'interaction Modélisation	274
Modélisation de la quantité à l'aide de types de valeur	278
Utilisation Propriétés et des pièces pour l'utilisation Modèle Bloc	280
Présentation des Diagrammes Fonctionnels Internes	281
Modélisation et assemblage des pièces	283
Modélisation Paramétriques Equations	286
Présentation Diagrammes Paramétriques	287
Systèmes d'équations utilisant des associations de parties	290
Mesures d'efficacité à l'aide de paramètres	292
Coordonner le comportement avec les activités	294
Actions les éléments fondamentaux du Comportementale	296
Présentation Diagrammes d'activité	299
Créer des hiérarchies d'activités	302
Spécification Séquence Action avec des flux de contrôle	303
Spécification du flux Item avec des flux Object	305
Entrées et sorties Modélisation avec paramètres et Pins	307
Visualisation des activités avec des simulations	310
Affectations et autres Relations	312
Modélisation du changement avec Statemachines	314
States et comportements	317
Présentation Diagrammes Statemachine	318
Déclencheurs et transitions	321
States et régions composites	325
Pseudo-États – La police de la circulation	328
Tableaux State - Une autre Vue	333
Visualisation et mise en œuvre avec des simulations	336
Les interactions comme Séquence de messages	338
Lignes de vie, messages et activations	339
Présentation du Diagramme de Séquence	344
Orchestration de messages avec des fragments	348
Visualisation avec des simulations	350
Simulation SysML dans Modelica et Simulink	351
Comment fonctionne Simulation SysML	353
Démarrage avec OpenModelica	355



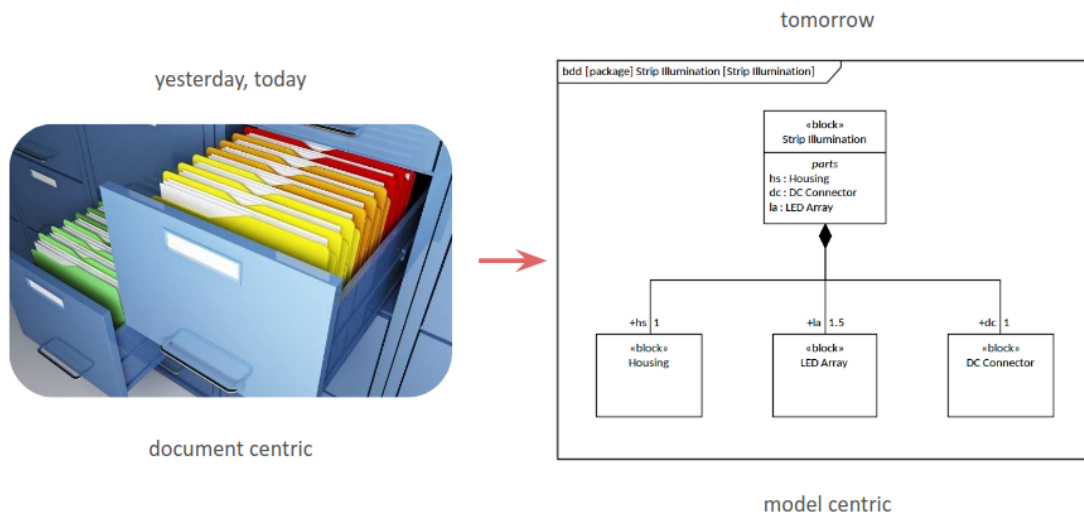
Créer des modèles pour Simulation .....	360
Exemple Modèle SysML .....	367
Présentation Paquetage (Structure de l'exemple Modèle ) .....	368
Diagramme Paquetage - Application du profil SysML .....	369
Diagramme Paquetage - Affichage de la structure Paquetage du Modèle .....	370
Définir le contexte (limites et cas d'utilisation) .....	373
Modèle de domaine opérationnel - Définition du contexte .....	374
Diagramme de cas d'utilisation - Cas d'utilisation de niveau supérieur .....	376
Diagramme de cas d'utilisation - Cas d'utilisation opérationnels .....	378
Élaboration du comportement ( Diagrammes Séquence et Statemachine ) .....	380
Diagramme de Séquence - Drive Black Box .....	381
Diagramme Statemachine - States opérationnels HSUV .....	385
Diagramme de Séquence - Démarrer le véhicule Black Box et White Box .....	387
Etablir Exigences ( Diagrammes et Tableaux Exigences ) .....	389
Diagramme des exigences - Hiérarchie des exigences HSUV .....	390
Diagramme d'exigences - Exigences dérivées .....	391
Diagramme d'exigences - Relations d'exigences d'accélération .....	392
Tableau - Exigences Tableau .....	393
Décomposer les pièces ( Diagrammes de définition Bloc , Diagrammes Bloc internes) .....	394
Interrompre lorsqu'une Variable Change de Valeur - Domaine Automobile .....	395
Interrompre lorsqu'une Variable Change de Valeur - Hybrid SUV .....	396
Diagramme Interne de Bloc - SUV Hybride .....	397
Interrompre lorsqu'une Variable Change de Valeur - Power Subsystem .....	398
Diagramme Interne de Bloc pour le Sous-Système de Puissance .....	399
Définition des ports et des flux .....	401
Interrompre lorsqu'une Variable Change de Valeur - ICE Flow Propriétés .....	402
Diagramme Interne de Bloc - CAN Bus .....	403
Interrompre lorsqu'une Variable Change de Valeur - Fuel Flow Propriétés .....	404
Diagramme Paramétriques - Débit De Carburant .....	406
Analyser les performances ( Diagrammes de contraintes, Diagrammes , Vues ) .....	407
Interrompre lorsqu'une Variable Change de Valeur - Contexte d'analyse .....	408
Diagramme Paquetage - Définition Vue performance .....	409
Diagramme Paquetage - Définition du point de vue .....	410
Diagramme Paquetage - Définition Vue .....	411
Diagramme Paquetage - Hiérarchie Vue .....	412
Diagramme Paramétriques - Mesures d'Efficacité .....	413
Diagramme Paramétriques - Economie .....	414
Diagramme Paramétriques - Dynamique .....	415
Diagramme de synchronisation (non normatif) - accélération 100 ch .....	416
Définition, décomposition et répartition des activités .....	417
Diagramme d'activité - Accélération (niveau supérieur) .....	418
Interrompre lorsqu'une Variable Change de Valeur - Accélération .....	419
Diagramme d'activité (EFFBD) - Accélération (détail) .....	420
Diagramme Interne de Bloc - Power Subsystem Comportementale and Flow Allocation .....	421
Diagramme Interne de Bloc : Valeurs spécifiques à la propriété - Test d'économie de carburant de l'EPA .....	422
Rencontrez les outils d' Ingénierie des Systèmes .....	423
Diagramme Exigences .....	424
Diagramme d'activité .....	427
Diagramme de cas d'utilisation .....	429
Générateur de scénarios .....	432

Audit .....	434
Calendrier .....	436
Interrompre lorsqu'une Variable Change de Valeur .....	438
Diagramme Interne de Bloc .....	439
Diagrammes du tableau de bord .....	441
Diagramme d'arbre de Décision .....	443
Diagramme Statemachine .....	445
Documentation .....	447
Matrice d'Analyse des Écarts .....	449
Carte de Chaleur .....	451
Importer et exporter des feuilles de calcul .....	452
Diagramme Paramétriques .....	454
Motifs .....	455
Matrice de relations .....	457
Feuille de Route Diagramme .....	459
Gestionnaire de Spécification .....	461
Carte de Stratégie .....	463
Bibliothèque .....	465
Modélisation Consciente du Temps .....	467
Fenêtre de traçabilité .....	470
Chaîne de valeur .....	472

# Guide de MBSE avec SysML

Ingénierie Systèmes Modèles Basée (MBSE) est devenue une approche précieuse pour la gestion et l'ingénierie des systèmes complexes. Elle s'éloigne de l'approche centrée sur les documents et permet de développer et d'utiliser des modèles à des fins très diverses, notamment la spécification des exigences, la conception, les compromis, architecture, la vérification, la validation, les simulations et support, etc.

Les modèles agissent comme une « police d'assurance » contre les erreurs d'ingénierie catastrophiques et contribuent à réduire le coût élevé des échecs lors des phases de spécification, de conception, de test, de mise en œuvre et support.







En utilisant Enterprise Architect pour MBSE, vous ouvrez la porte à une toute nouvelle façon de penser et de travailler. Enterprise Architect est une plate-forme de collaboration rigoureuse qui permet à un large éventail de parties prenantes de partager des idées, des problèmes, des solutions et des implémentations, notamment :


- Clients
- Dirigeants
- Responsables de l'ingénierie
- Chefs d'équipe d'ingénierie
- Architectes et designers
- Ingénieurs Systèmes
- Ingénieurs en logiciel
- Testeurs
- Fournisseurs
- Intégrateurs de systèmes
- Personnel Support
- Les utilisateurs ou leurs substituts


Les modèles sont également disponibles en temps réel pour visualisation, contributions, révisions et discussions via une interface Navigateur sécurisée, utilisant la puissance de la collaboration pour créer des architectures et des conceptions robustes et bien formées.


## Discussions

 Guys what is the current status of this requirement, did the customer reply to all of our questions?    
2017-03-30 03:35:10 PM Project Manager

 Gareth Edwards from London replied to with the answers to question 1 to 8 but waiting on a reply regarding the others  
2017-03-30 03:38:02 PM Junior Developer

 Just got off the phone with Beth McSimmons in Scotland and she is emailing the answers to the remaining questions.  
2017-03-30 03:41:24 PM Senior Developer

 OK thanks  
2017-03-30 03:42:50 PM Project Manager

Post reply 

Dans ce guide, nous allons explorer de nombreuses fonctionnalités d' Enterprise Architect qui peuvent être exploitées pour amener un ingénieur individuel, une équipe, une organisation ou un secteur industriel entier au niveau de pratique et de performance requis à une époque dominée par l'innovation et des changements techniques sans précédent. Le guide fournira le contexte du Langage Modélisation des Systèmes et montrera comment les constructions du langage peuvent être créées dans Enterprise Architect . Il est destiné à donner aux nouveaux venus à la fois au langage et à l'outil une exposition à tout ce qui est possible dans le domaine de Ingénierie des Systèmes en utilisant cette plate-forme d'ingénierie et de collaboration. Selon les mots du célèbre physicien du XVIIe siècle, Sir Isaac Newton :

*« Si j'ai vu plus loin que les autres, c'est parce que je me suis tenu sur les épaules de géants. »*

Enterprise Architect fournit une plateforme où cette vision peut prendre forme, où, grâce aux fonctionnalités et facilités de collaboration telles que les simulations et l'automatisation, un ingénieur peut voir les opportunités, concevoir des solutions et concevoir l'avenir.

Enterprise Architect inclut fonctionnalités majeures pour les ingénieurs systèmes, les architectes techniques et autres personnes souhaitant coupler leur travail modélisation et de simulation dans Enterprise Architect avec MATLAB, Octave, OpenModelica et bien d'autres. Les classes « Solveur » et une Bibliothèque mathématique étendue dans le moteur JavaScript offrent une capacité Simulation considérablement étendue. Pour plus d'informations, consultez la rubrique d'aide [Mathematical Simulations](#) .

# Une équation à quatre variables

Le titre de ce guide implique qu'il n'y a que deux disciplines à apprendre :

- Ingénierie Systèmes Modèles Basée (MBSE) et
- Le Systems Modeling Language (SysML)

Il faut cependant prendre en compte quatre aspects avant d'adopter cette approche. Il s'agit d'une équation à quatre inconnues, dont chacune doit être résolue avant qu'une équipe puisse réussir un projet ou une initiative MBSE. Les variables de l'équation sont les suivantes :

- La méthode ou le processus d'ingénierie
- La discipline de modélisation
- Le Systems Modeling Language (SysML)
- Enterprise Architect , l'outil modélisation de choix

Il n'est pas impératif de maîtriser ces quatre aspects immédiatement, mais il est important qu'ils soient connus et travaillés et que l'équipe développe les compétences nécessaires pour les comprendre individuellement et comment ils sont liés les uns aux autres. Par exemple, comment créer un diagramme Exigences SysML à l'aide Enterprise Architect et quelles propriétés doivent être incluses, quand doit-il être réalisé et à quelles autres parties du modèle doit-il être lié.

*Approche MBSE = Processus + Langage + Modélisation + Outil*

Le guide abordera toutes ces questions et, à la fin, le lecteur ne sera pas confronté à des variables inconnues, mais l'équation aura des valeurs basées sur la formation , résolvant ainsi l'équation avec les quatre variables avec lesquelles nous avons commencé. À ce stade, le lecteur sera sur la bonne voie pour une pratique solide de la modélisation technique.

## La méthode ou le processus d'ingénierie

Le Langage Modélisation des Systèmes est agnostique par rapport aux processus et peut être utilisé avec n'importe quelle méthode ou processus. Ce point n'est parfois pas compris par les nouveaux venus dans le langage qui s'attendent à ce qu'il soit prescriptif et donne des directives claires sur les éléments, diagramme et modèles à créer et quand. Cette position agnostique offre une grande flexibilité et permet au langage d'être utilisé de manière applicable au processus et au problème sous-jacent ou au domaine de solution.

Les éléments, connecteurs, diagrammes et définitions de langage définis dans le cadre du langage Systems Modélisation ont tous été créés dans le but exprès de permettre aux ingénieurs de créer des modèles de :

- mission
- parties prenantes
- exigences
- mesures d'efficacité
- aspects structurels et comportementaux d'un système tels que les composants qui mettent en œuvre les exigences

Le processus utilisé par une équipe pour créer, gérer et diffuser les artefacts est complètement arbitraire et doit être défini au niveau de l'organisation ou de l'équipe.

L'ingénierie système nécessite généralement une approche collaborative ou multidisciplinaire dans laquelle les équipes travaillent ensemble pour produire un résultat qui répond aux besoins des parties prenantes. Tout processus comporte deux aspects importants :

- Un processus de gestion - qui régit les parties prenantes, les risques, le calendrier, le budget et la qualité
- Un processus technique - qui gère architecture, l'analyse, la conception, l'intégration et les tests

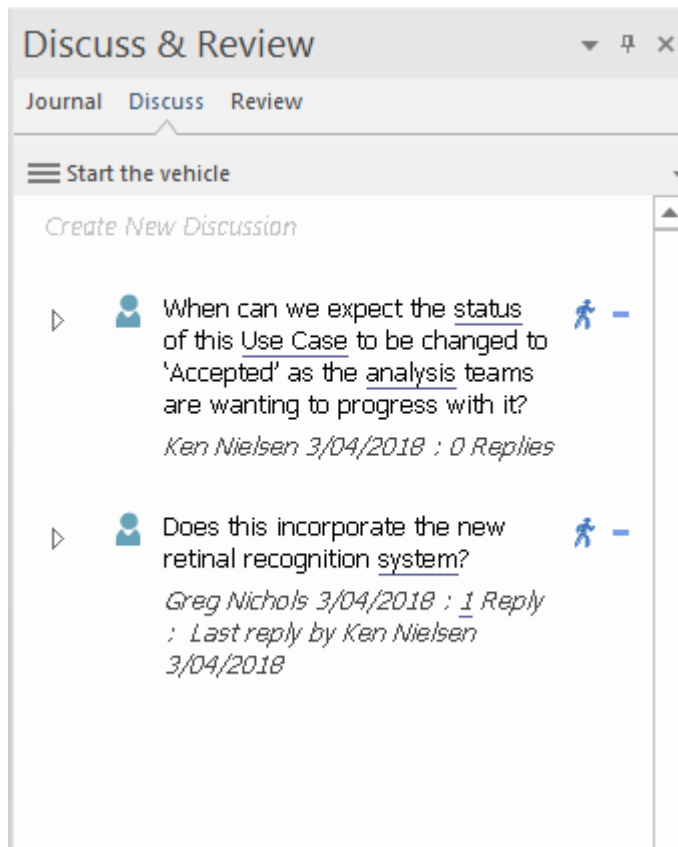
Les deux processus nécessitent toutefois clairement des points de contact pour garantir que la mission globale et les buts et objectifs du projet sont respectés.

Enterprise Architect vous permet d'utiliser n'importe quel type de processus, qu'il soit défini formellement, intégré à une norme ou élaboré en interne. Il existe également facilités au sein d' Enterprise Architect qui vous permettent de définir, publier et partager un processus sur mesure.

### Une équipe bien soutenue

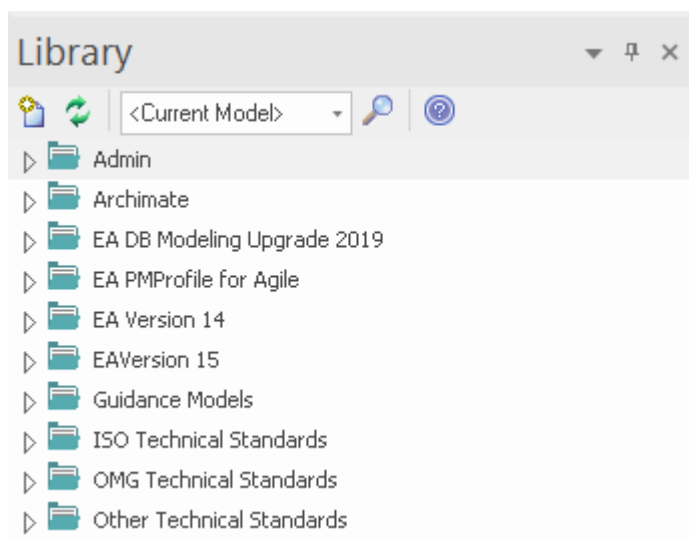
Enterprise Architect propose une large gamme d'outils qui aideront les équipes à collaborer, indépendamment de leur situation géographique ou de la distance qui les sépare. Le produit a été conçu dès le départ comme une plateforme collaborative, permettant aux parties prenantes, qu'elles soient techniques ou non, de travailler ensemble dans une structure collaborative et intégrée.

Le référentiel peut être basé sur le Cloud et les utilisateurs peuvent se connecter en toute sécurité depuis n'importe quel endroit du globe, créant ainsi une équipe virtuelle. Ceci est important pour un certain nombre de projets où l'expertise n'est pas disponible localement ou lorsque le projet lui-même est global. Les utilisateurs et les équipes peuvent utiliser les fonctionnalités de collaboration telles que Discussions, Chats, Révisions et Mail de Modèle pour travailler ensemble. Le résultat sera architecture et une conception collaboratives qui ne sont pas le résultat du travail d'un seul ingénieur, mais le résultat de plusieurs esprits, et le travail sera plus que la somme de ses parties.

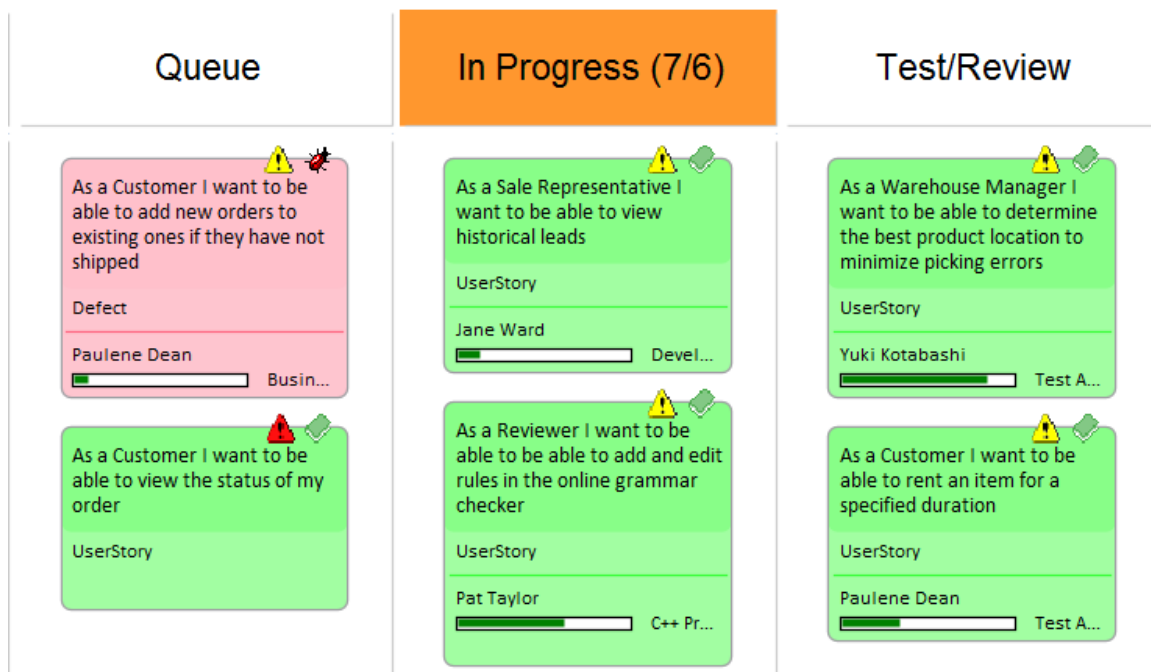


Ces outils sont efficaces car ils peuvent être utilisés pour annoter des modèles, des éléments et diagrammes, permettant aux utilisateurs de travailler ensemble comme s'ils travaillaient en collaboration sur un tableau blanc dans la même pièce.

Bibliothèque de Modèle est un autre outil de collaboration pratique qui permet d'inclure tout type de fichier dans le référentiel ou de le répertorier avec un lien hypertexte et/ou une référence URL vers son emplacement externe sur un site Web. Des documents tels que des normes, des spécifications, des lignes directrices, guidage, des exemples, des mentors et d'autres documents peuvent tous être catalogués dans la Bibliothèque d'Équipe.



Il existe une large gamme d'autres outils qui peuvent être utilisés pour faciliter le travail en équipe, notamment le Gestionnaire d'images, les Calendriers, la Publication, Kanban, fonctionnalités Gestion de Projet et bien d'autres encore. Cet exemple montre un diagramme Kanban qui peut être utilisé pour visualiser ce sur quoi travaille une équipe Agile qui développe des composants physiques ou logiciels d'un système. Pour plus d'informations, consultez la rubrique d'aide [The Modeling Team](#).



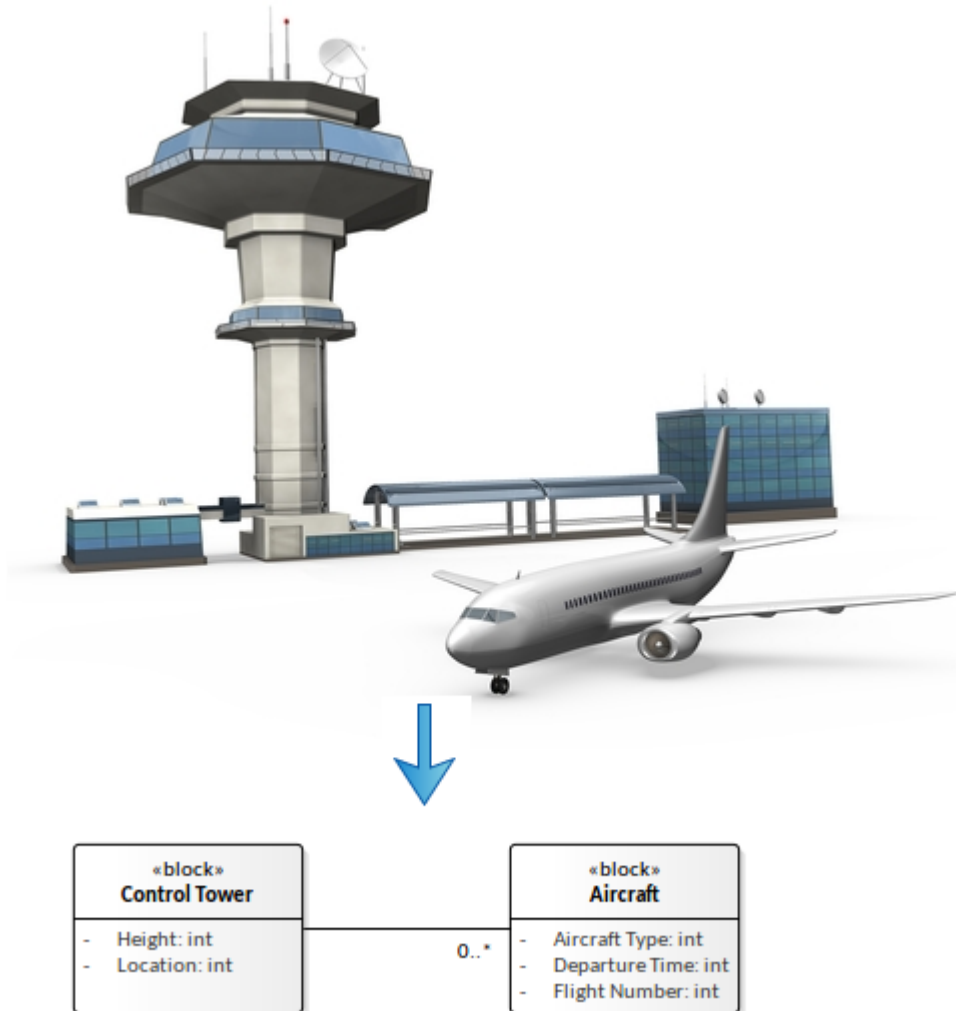


# Modélisation comme discipline

La plupart des gens, y compris les ingénieurs système, trouvent généralement plus facile d'écrire une longue description d'un sujet plutôt qu'un résumé succinct et concis – c'est analogue au défi de modélisation.

*La question n'est pas tant de savoir quoi inclure, mais plutôt quoi omettre.*

L'un des avantages de l'ingénierie système basée sur des modèles est précisément celui-ci : elle encourage les ingénieurs à créer des modèles descriptifs, clairs et concis. Les longues phrases (et parfois décousues) des processus basés sur des documents sont remplacées par diagrammes clairs et concis qui décrivent sans ambiguïté les exigences, la structure et le comportement du système.



Certains décrivent modélisation comme une discipline hermétique et la décrivent comme l'un des « arts obscurs » pratiqués par des ingénieurs alchimistes vêtus de robes violettes. Cela explique pourquoi modélisation est rarement enseignée en tant que matière dans nos universités et qu'il n'existe pas non plus de littérature abondante sur le sujet, ce qui fait qu'elle apparaît comme un art mystérieux plutôt que comme ce qu'elle est réellement : une science qui peut s'apprendre.

Il existe plusieurs types de modèles différents, notamment :

- Modèles à Échelle
- Modèles physiques
- Modèles abstraits

Dans ce guide, nous nous intéressons principalement aux modèles abstraits, car ce sont les modèles que nous allons

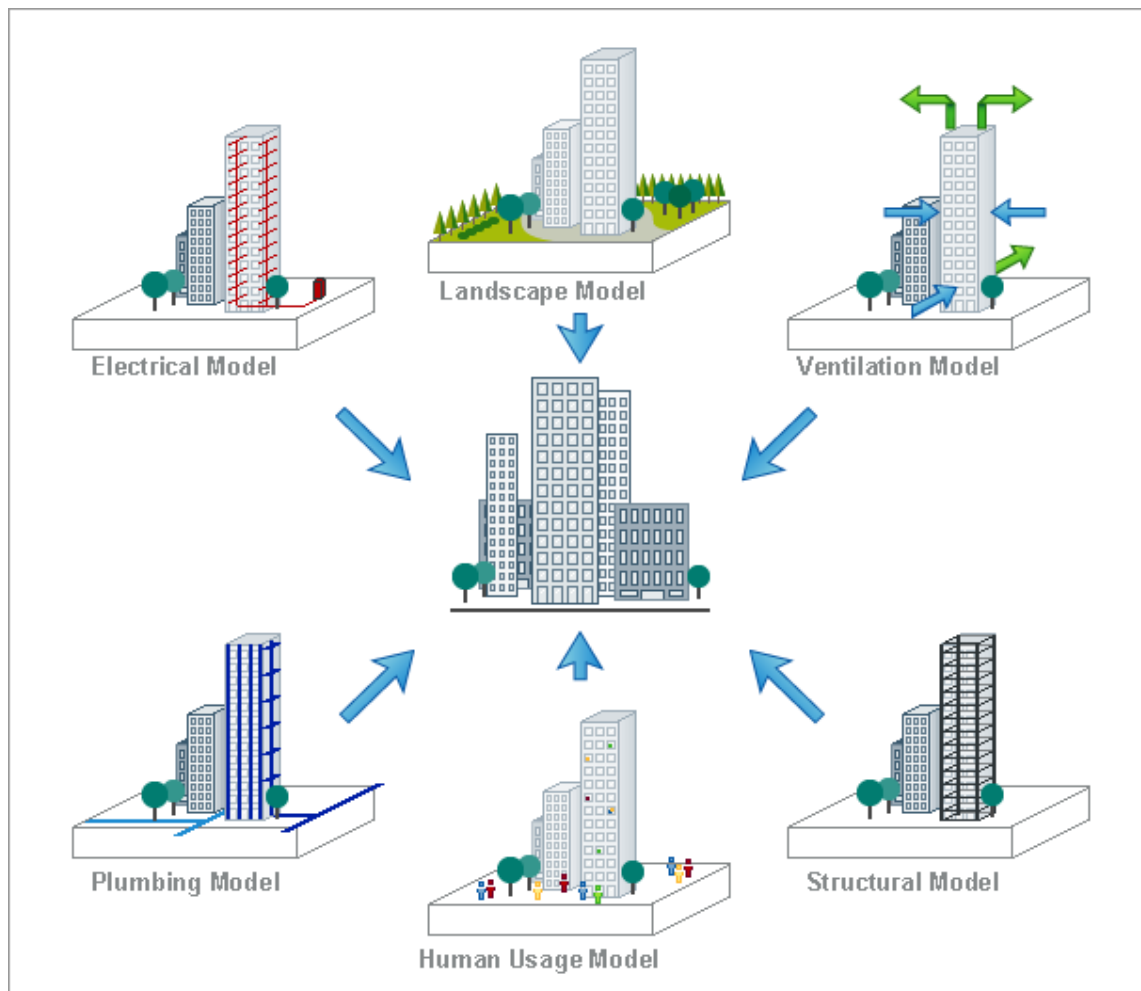
généralement créer à l'aide Enterprise Architect et du Langage Modélisation des Systèmes .

Ces modèles sont - comme leur nom l'indique - des abstractions de la réalité qui cherchent à mettre en évidence les aspects les plus importants d'une entité, d'un sous-système ou d'un système, tout en laissant de côté les éléments qui ne sont pas importants ou qui ne sont pas pertinents de ce point de vue. Pour plus d'informations, consultez le Rubrique d'aide sur [Models](#) .

## Démarrage

Un modèle abstrait est une représentation d'un élément du monde réel qui nous aide à le comprendre sans avoir besoin de le visualiser. En général, un modèle est beaucoup plus petit et constitue une vue simplifiée d'un système ou de l'une de ses parties. Un modèle peut également être créé en se concentrant sur un seul aspect ou une seule facette d'un système, par exemple le système de communication ou de navigation d'un avion.

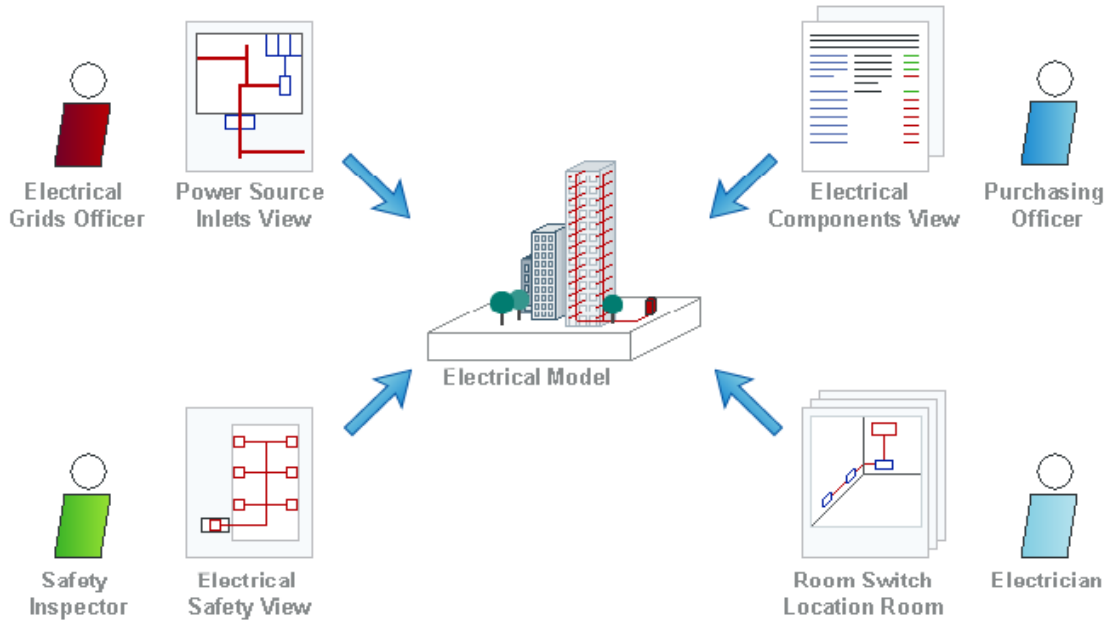
Un bâtiment est une structure complexe qui comporte un certain nombre de systèmes différents, notamment structurels, électriques, de ventilation, de plomberie, d'aménagement paysager, etc.



En construisant un certain nombre de modèles, nous sommes en mesure de créer une vue simplifiée de chacun des sous-systèmes, ce qui facilite la compréhension de cet aspect du bâtiment. Les modèles eux-mêmes doivent également être résolus les uns par rapport aux autres. Par exemple, il est essentiel que le système d'alimentation représenté dans le modèle électrique fournisse de l'électricité à l'équipement de climatisation modélisé dans le système de ventilation. Le modèle d'utilisation humaine doit être résolu par rapport au modèle paysager pour garantir que les jardins et l'aménagement paysager répondent aux besoins récréatifs des occupants et de leurs visiteurs.

Un modèle est généralement visualisé par un certain nombre d'intervenants différents qui ont généralement des rôles très différents par rapport à la partie du système modélisée. Pour garantir que le modèle est utile à un intervenant particulier, des vues peuvent être créées représentant ce qui est vu lorsque l'on regarde le modèle depuis un point de vue particulier.

Le modèle électrique du bâtiment est utile à un certain nombre d'intervenants différents, qui ont tous un point de vue différent sur le système, notamment le responsable du réseau électrique, l'inspecteur de sécurité, l'électricien et le responsable des achats.



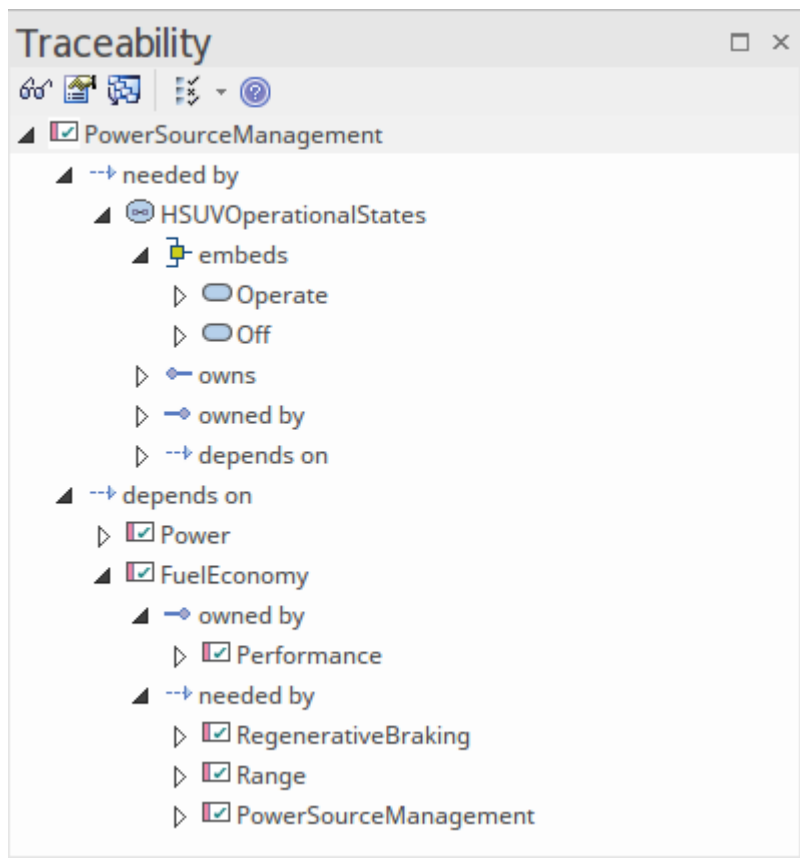
## Définir l'objectif d'un Modèle

Le passage à l'ingénierie système basée sur Modèle comporte un certain nombre de défis et de pièges pour les équipes qui sont plus habituées à travailler avec des méthodes centrées sur les documents. Le piège le plus courant est probablement de commencer modélisation sans avoir une compréhension ou une définition claire de l'objectif des modèles.

En comparaison avec les approches modélisation centrées sur les documents, il est plus difficile de définir l'objectif d'un modèle que celui d'une suite de documents. Le modèle est bien plus utile et efficace qu'un document et peut être utilisé pour effectuer des tâches inimaginables avec un système basé sur des documents. Voici quelques-uns des avantages de l'approche basée sur les modèles :

- Les contrôles de cohérence peuvent être facilement appliqués
- Des vues alternatives peuvent être facilement créées et maintenues cohérentes
- Si une documentation est requise, elle peut être générée automatiquement
- Les modèles sont imbriqués et cohérents
- L'impact du changement peut être visualisé et automatisé
- Les modèles peuvent être maintenus conformes à un métamodèle sous-jacent
- Les modèles peuvent être versionnés et référencés
- La traçabilité Exigences peut être facilement gérée
- Les modèles peuvent être facilement manipulés et modifiés
- Le modèle peut être utilisé pour générer du code et des normes
- Les modèles peuvent être simulés pour produire des visualisations riches
- Les modèles peuvent être transformés d'un niveau d'abstraction à un autre
- Certaines parties du modèle peuvent être facilement réutilisées, ce qui crée une efficacité

Ce diagramme montre comment la traçabilité peut être visualisée et gérée dans l'outil, vous permettant de visualiser la manière dont les parties des modèles s'imbriquent et comment les éléments forment un graphique de connexions, vous aidant à décrire et à comprendre votre modèle.



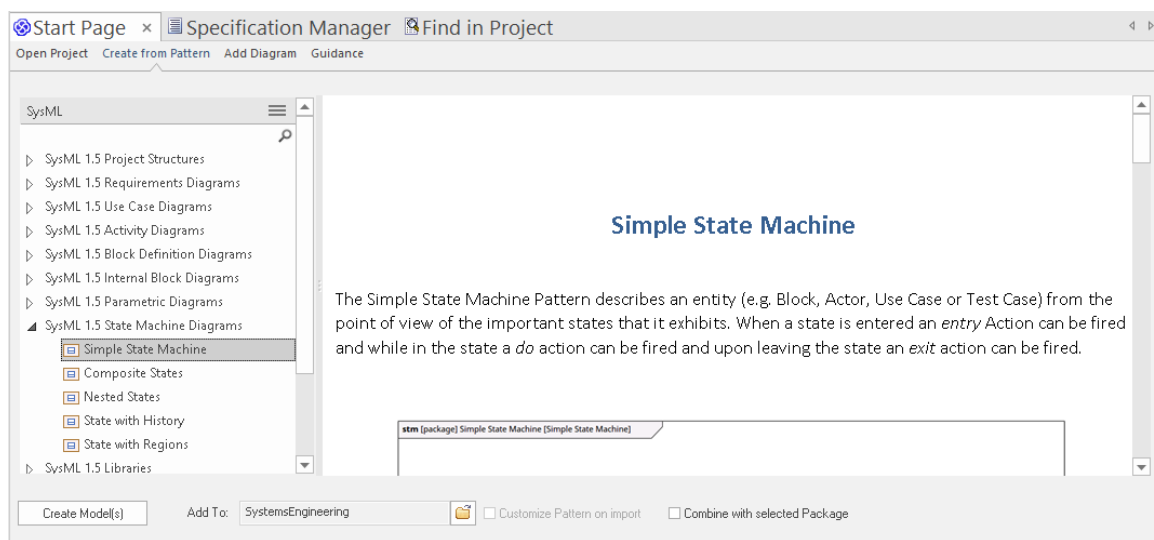
Enterprise Architect exploite la puissance de SysML, ainsi qu'un large ensemble d'outils conçus à l'intention des responsables d'ingénierie système, des ingénieurs système et d'autres parties prenantes, offrant des moyens simples mais efficaces de tirer parti de l'approche Ingénierie des Systèmes basée sur les modèles.

L'adoption d'une approche basée sur des modèles présente d'autres avantages considérables, notamment la garantie que les projets et les programmes de travail sont exécutés avec rigueur, productivité et efficacité à l'aide d'un outil qui encourage l'excellence et la collaboration. Pour plus d'informations, consultez la rubrique d'aide [Why Enterprise Architect?](#).

## Décider où Démarrer

Le processus de modélisation peut être assez intimidant pour les ingénieurs qui débutent dans Ingénierie Systèmes Modèles Basée . Plus que toute autre chose, il semble que la question de savoir par où commencer modélisation soit l'équivalent pour l'ingénieur de l'inertie de la « toile vierge » de l'artiste.

Enterprise Architect apporte une solution bienvenue à ce problème, en fournissant une série de motifs qui peuvent être utilisés pour créer des points de départ pour une initiative ou un projet, y compris tous les types diagramme SysML avec un certain nombre de motifs pour chaque type. Pour plus d'informations, consultez la rubrique d'aide [The Model Builder](#) ,

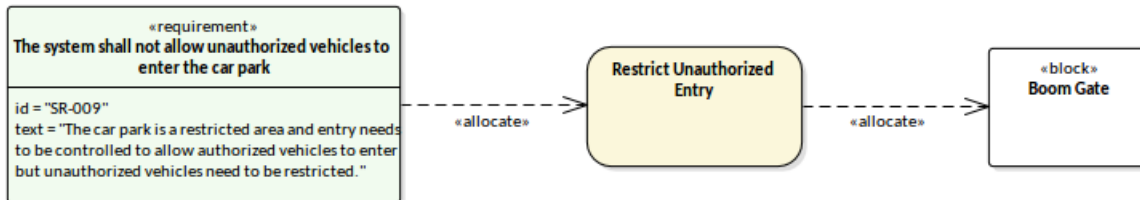


Les manuels décrivent généralement une série d'étapes qui doivent être exécutées dans un ordre prescrit, mais dans la pratique, ces recettes ne fonctionnent pas car les projets sont sensiblement plus complexes que les méthodes génériques décrites dans les livres, et les dépendances complexes entre projets et ressources signifient que les tâches ne peuvent pas être exécutées dans un ordre prescrit.

Le point de départ sera généralement déterminé par la méthode ou le processus d'ingénierie utilisé pour le projet, qui peut être une cascade, un processus itératif, une combinaison des deux ou un autre type de processus. Quel que soit le type de processus utilisé, une compréhension claire de la mission est souvent un bon point de départ, et définir les parties prenantes concernées ainsi que leurs préoccupations et leurs exigences est souvent une bonne étape suivante.

## Pièces de connexion du Modèle

Le Langage Modélisation des Systèmes encourage les ingénieurs à créer une série de modèles qui, pour le novice ou le nouveau venu dans l'ingénierie des systèmes basée sur les modèles, semble fragmenter la vision du système. En réalité, le langage SysML décrit un réseau de modèles, chacun répondant à des préoccupations particulières mais reliés entre eux pour décrire le système et ses parties dans leur ensemble.



Dans cette illustration, nous voyons un diagramme Enterprise Architect convaincant qui décrit les connexions entre les parties du modèle à l'aide d'éléments et de connecteurs spécialisés, à savoir une exigence, une activité et un Bloc utilisant la relation Allocate. Les éléments peuvent être réutilisés dans un nombre quelconque de diagrammes, et la modification de leurs propriétés à un seul endroit les mettra à jour dans tous les contextes. Diagrammes peuvent être créés rapidement et facilement à l'aide d'un certain nombre de fonctionnalités, et ils peuvent être visualisés de diverses manières, telles que des listes, tableaux et des feuilles de calcul. Les diagrammes peuvent être filtrés et les éléments peuvent être remplacés par des icônes graphiques pour susciter davantage d'intérêt auprès d'un public non technique. Pour plus d'informations, consultez la rubrique d'aide [Model Diagrams](#).

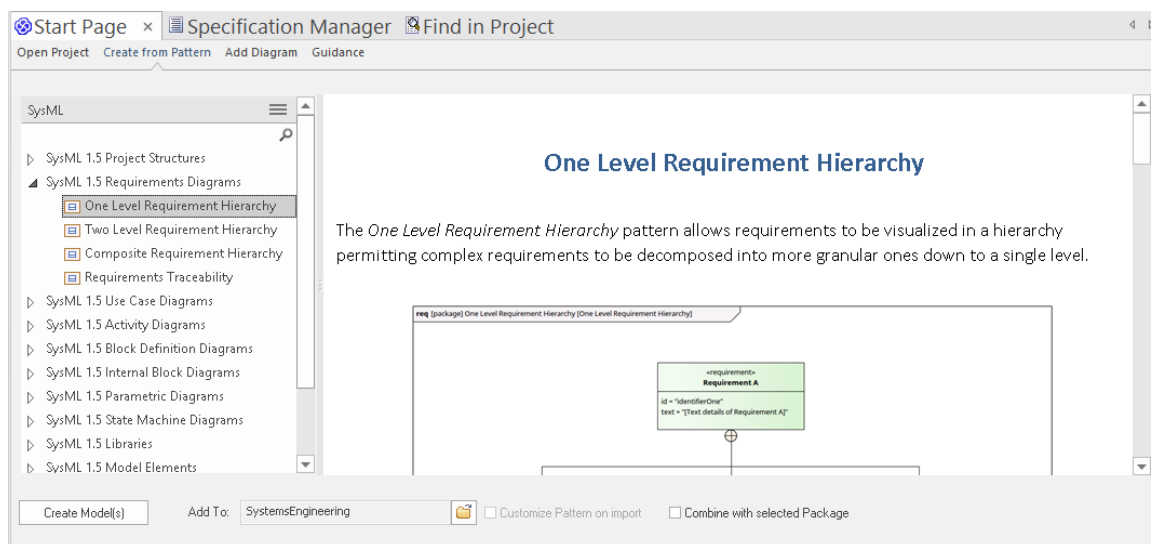
C'est là la véritable force de l'approche Ingénierie Systèmes Modèles Basée, car elle permet de visualiser un système de multiples façons, depuis une vue complète et de haut niveau jusqu'à plusieurs couches de décomposition ou de hiérarchie. Chaque niveau est connecté, les lacunes ou ruptures dans les modèles peuvent être facilement identifiées et des remèdes peuvent être trouvés.



# Assurer la qualité d'un Modèle

La qualité d'un modèle se reflétera finalement dans la qualité du système qu'il représente. Enterprise Architect a été conçu pour fournir une plate-forme de création et de gestion de modèles collaboratifs de haute qualité. Il existe un certain nombre de fonctionnalités qui aident le modélisateur à atteindre le niveau de qualité requis, notamment facilités telles que :

- *Métamodèle* - qui peut être défini par un utilisateur pour créer efficacement une grammaire pour le modèle, garantissant que les utilisateurs créent des phrases modélisation « conformes » ; pour plus d'informations, consultez la rubrique d'aide [Developing Profiles](#)
- *Validation Modèle* - qui permet de vérifier la conformité du modèle avec le métamodèle sous-jacent
- *Discussions et chat* - qui permettent aux modélisateurs de travailler en collaboration sur un problème ou une solution
- *Révisions* - qui permettent aux experts internes ou externes de visualiser et de critiquer les modèles
- *Modèle Motifs* - qui fournissent Paquetages de modèles créés par des experts pour fournir un point de départ pour modélisation
- *Recherches* - qui aident à trouver des problèmes particuliers dans le modèle



Ce diagramme montre les Modèle Motifs en action, où les modélisateurs novices et expérimentés peuvent, d'une simple pression sur un bouton, créer des modèles et diagrammes bien formés en utilisant un corpus productif de modèles de bonnes pratiques de l'industrie - tous conformes à SysML. Il y a également une explication détaillée et une discussion sur la façon d'utiliser le motif, où obtenir de l'aide supplémentaire et plus encore. Pour plus d'informations, consultez la rubrique d'aide [The Model Builder](#) .

# Le Systems Modeling Language (SysML)

Le Systems Modeling Language (SysML) a été défini dans le but de représenter les artefacts des problèmes et solutions Ingénierie des Systèmes ou des programmes de travail de manière cohérente, efficace et robuste.



SysML est conçu pour fournir des constructions simples mais efficaces pour modélisation d'une large gamme de problèmes et de solutions d'ingénierie des systèmes. Il peut être utilisé à diverses fins, mais il est particulièrement efficace pour spécifier les exigences, la structure, le comportement, les allocations et les contraintes sur les propriétés du système afin support l'analyse technique, y compris l'analyse paramétrique et la simulation. SysML peut être utilisé avec de nombreux processus et méthodes tels que structurés, orientés objet, itératifs, en cascade et bien d'autres.

Le langage a été conçu et amélioré pendant plus de dix ans pour être adapté à modélisation de systèmes d'une complexité toujours croissante. Ces changements ont vu un langage relativement compact et concis devenir plus large et plus diversifié ; néanmoins, la majorité des projets d'ingénierie système peuvent toujours être modélisés avec une petite partie du langage, que nous pourrions appeler « Core SysML ». Pour plus d'informations, consultez la rubrique d'aide [Modeling Systems in Enterprise Architect](#) .

## Enterprise Architect l'outil Modélisation

Enterprise Architect est à la fois un Modèle Référentiel et une Plateforme de Collaboration, ce qui en fait un outil efficace pour les projets Ingénierie Systèmes Modèles Basée . Il permet aux membres de l'équipe - y compris les sponsors de projet, les responsables d'ingénierie, les clients et les ingénieurs - de collaborer sur des projets dans un environnement rigoureux et productif. Grâce à WebEA et Prolaborate , la collaboration peut se poursuivre sur des appareils mobiles tels que les téléphones portables, les tablettes et les ordinateurs portables.

A l'ère de l'information et de l'innovation, un outil doit faire bien plus que stocker des informations ou permettre aux utilisateurs de visualiser diagrammes et des modèles. Enterprise Architect a relevé ce défi et propulsé son offre Ingénierie des Systèmes à un autre niveau, avec des outils tels que :

- *Générateur de scénarios* , qui crée automatiquement diagrammes d'activité à partir des étapes de cas d'utilisation et génère des cas Test à partir de scénarios ; pour plus d'informations, consultez la rubrique d'aide [Scenario Builder](#)

The screenshot displays the Scenario Builder interface in Enterprise Architect. The top menu bar includes 'General', 'Requirements', 'Constraints', 'Scenarios', 'Files', 'Links', and 'Templates'. The 'Scenarios' tab is active, showing a scenario named 'In-house Account Processing' with a 'Type' of 'Basic Path'. Below this, there are tabs for 'Description' and 'Structured Specification'. A toolbar with various icons is visible. A table lists the scenario steps:

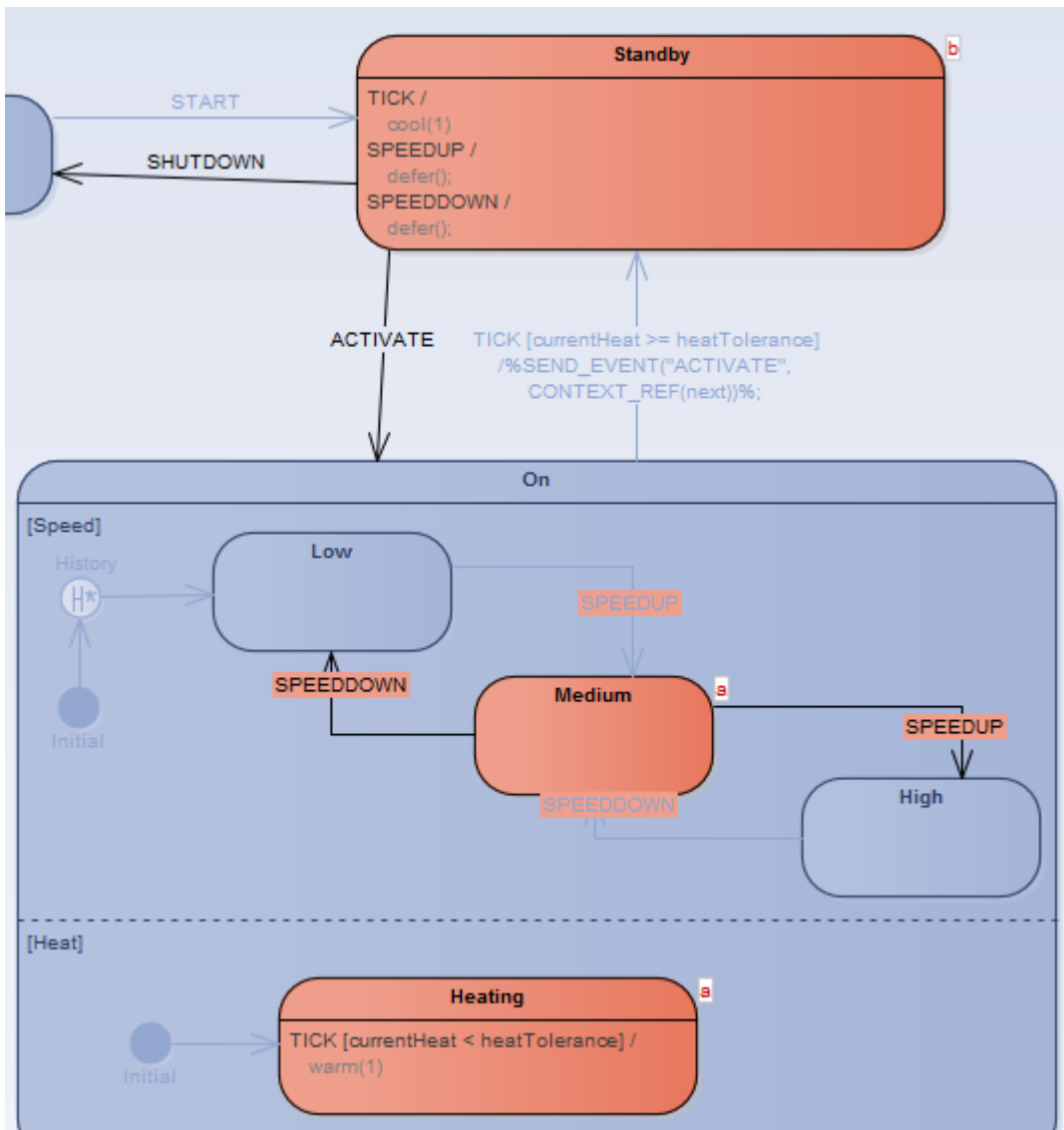
Step	Action	Uses	Results	State
1	Customer inserts ATM-card into ATM	ATM-Card	Initiate Transaction	START

Below the table, a diagram shows a 'Start' node leading to an activity node labeled 'Customer inserts ATM-card into ATM'. To the right, a 'Tagged Values' window is open for the activity, showing the following data:

Property	Value
result	<memo>*
state	<memo>*
step_guid	{F95E88CD-5B44-4faa-8C7D-FF66CC1ABE0...}
trigger	User
uses	<memo>*

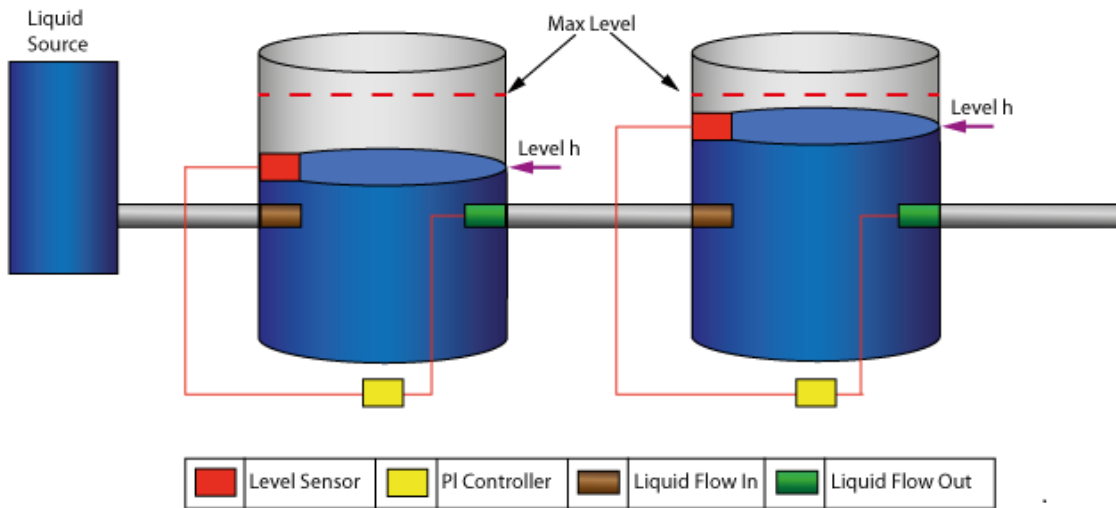
The 'result' property is highlighted, and its value is 'Initiate Transaction'.

- *Statemachines Exécutables* , qui permettent de générer automatiquement du code de programmation à partir de Statemachines ; pour plus d'informations, consultez la rubrique d'aide [Executable StateMachines](#)

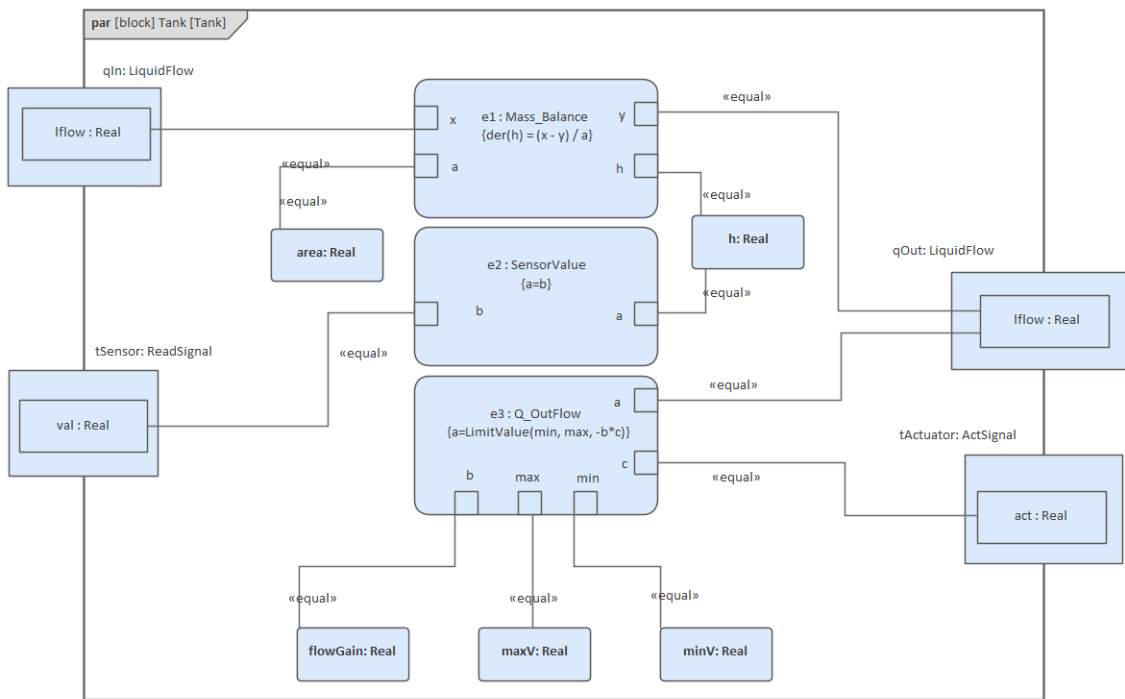


- *Simulations Paramétriques* utilisant OpenModelica ou Simulink, donnant vie aux modèles et permettant de visualiser et d'analyser des problèmes complexes et souvent insolubles pour support l'analyse des compromis et les recherches d'ingénierie ; pour plus d'informations, consultez l'aide [Simulation](#) sujet

Dans cet exemple, les relations entre les facteurs contrôlant un fluide s'écoulant entre deux réservoirs sont définies :

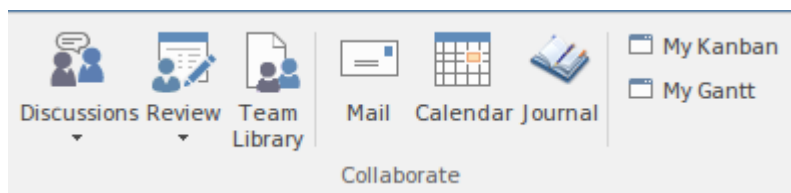


Les modèles peuvent ensuite être simulés à l'aide des fonctionnalités avancées de simulation d'OpenModelica.



## Plateforme de collaboration

L'ère de l'information s'est transformée en ère de l'innovation presque pendant que nous dormions, et il est maintenant plus que jamais impératif que les équipes travaillent ensemble de manière nouvelle et cohérente. Le partage de documents et de fichiers sur disque et le travail avec diagrammes statiques sont maintenant des choses que nous nous attendons à voir dans les musées. Des solutions réactives, robustes et innovantes ne peuvent être obtenues que par des équipes travaillant avec des outils exceptionnels qui permettent non seulement de construire des modèles et de faciliter la collaboration, mais aussi d'effectuer le travail. Enterprise Architect est une boîte à outils multifonctions qui permet aux équipes de collaborer, en réunissant les meilleurs esprits et les mains les plus expérimentées d'un large éventail de disciplines interdépendantes. Les personnes qui contribuent à la mise à jour et à la visualisation des modèles peuvent être dispersées géographiquement, opérer dans des fuseaux horaires différents, provenir de différentes organisations ou même parler des langues naturelles différentes.

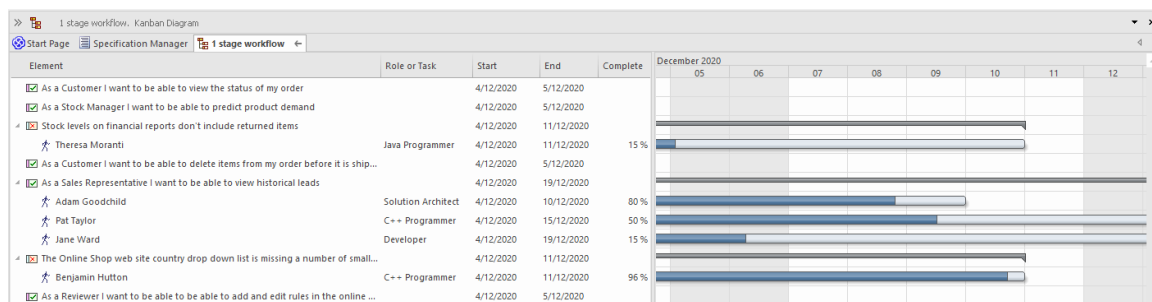


Cette image montre certaines des fonctionnalités de collaboration utiles disponibles à partir du ruban Démarrer . Les discussions et Révisions sont également disponibles à partir de WebEA et Prolaborate , ce qui permet au personnel modélisation et non-modélisation de collaborer, ce qui donne lieu à des solutions plus robustes et plus adaptées. Pour plus d'informations, consultez la rubrique d'aide [Teams & Collaboration](#) .

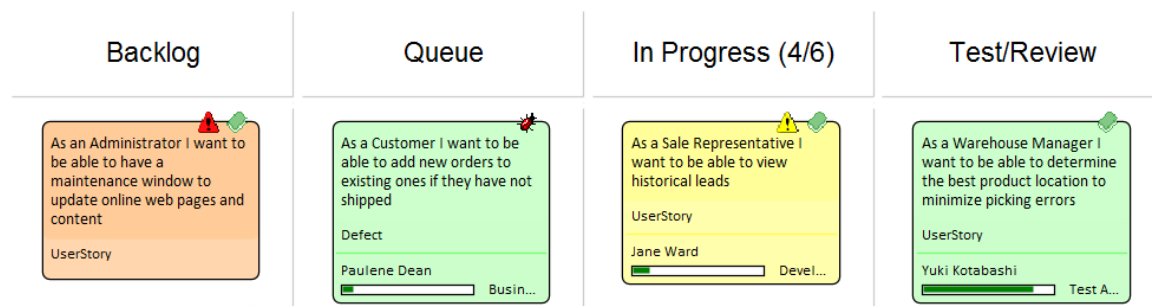
# Gestion de Projet Établi

Enterprise Architect propose une large gamme d'outils pour gérer un projet Ingénierie Systèmes Modèles Basée . Il peut ainsi servir d'atelier de gestion de projet pouvant être utilisé pour gérer un projet d'ingénierie. L'ensemble du cycle de vie des systèmes peut être modélisé dans l'outil, depuis la conceptualisation des besoins métier jusqu'à la conception, la mise en œuvre, l'utilisation, support et enfin la mise hors service ou la mise au rebut du système.

Il existe des diagrammes de Gantt, des calendriers, des bibliothèques Modèle , des registres de risques, de défauts, de tâches, d'efforts et de mesures, pour n'en citer que quelques-uns. Feuilles de Route sont une autre fonctionnalité précieuse qui permet à un chef de projet de visualiser l'évolution d'un projet au fil du temps, d'un état actuel à un certain nombre d'états de transition ou futurs.



Une équipe peut également travailler de manière cohérente en utilisant les tableaux Kanban intégrés qui permettent de visualiser des éléments tels que Exigences , les récits d'utilisateurs, Défauts et les modifications, etc., pendant qu'ils sont activement traités. L'allocation des ressources et Propriétés telles que la priorité et le statut peuvent être visualisées via les éléments du tableau et les limites de dépassement sont affichées.



Cette technique éprouvée, qui trouve son origine dans le secteur automobile japonais, a été mise en œuvre dans Enterprise Architect de manière à améliorer considérablement la productivité de votre équipe et la gestion de vos projets. Pour plus d'informations, consultez la [Project Build & Deploy](#) Rubrique d'aide.

## Modèle Référentiel

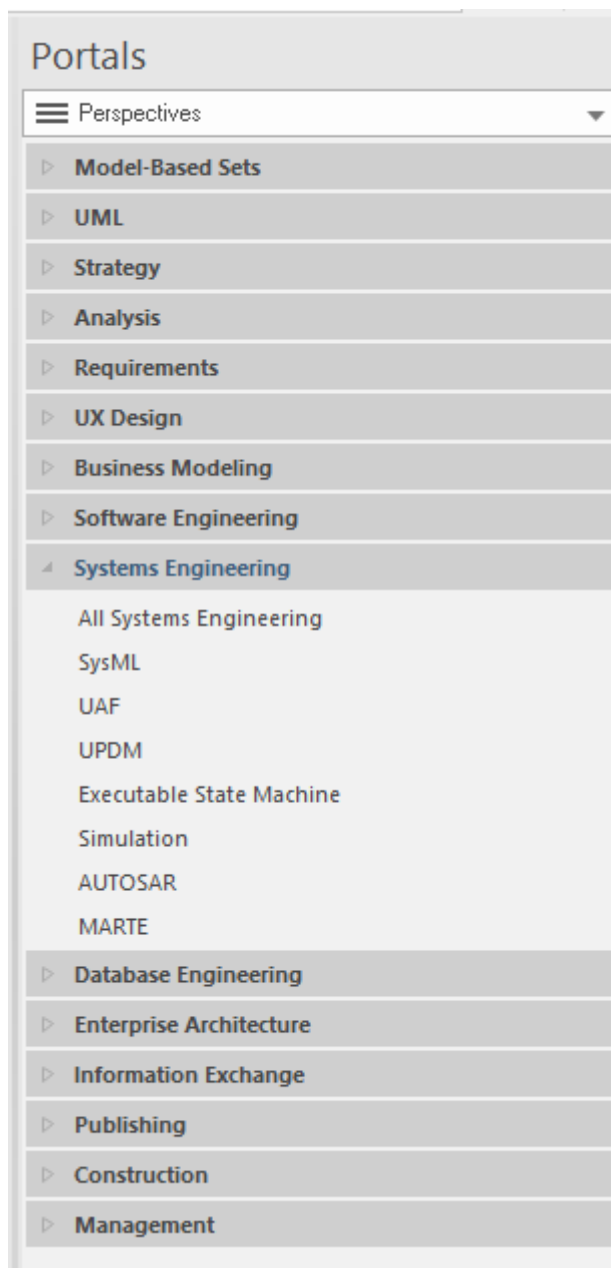
Enterprise Architect est avant tout un référentiel de modèles qui permet de gérer les modèles depuis leur création jusqu'à leur mise hors service. Le référentiel est stocké dans une base de données relationnelle qui peut être hébergée dans une configuration client-serveur ou dans le facilité d'un service Cloud , sur site ou hors site dans un environnement Cloud . Ainsi, même si les modélisateurs travaillent avec diagrammes et des éléments visuels, ces diagramme sont tous codifiés et stockés dans la base de données du référentiel. Le référentiel peut contenir n'importe quel nombre de modèles et peut être organisé pour être réutilisé et pour les modèles d'entreprise et de projet. Pour plus d'informations, consultez la rubrique [The Model Repository](#) .



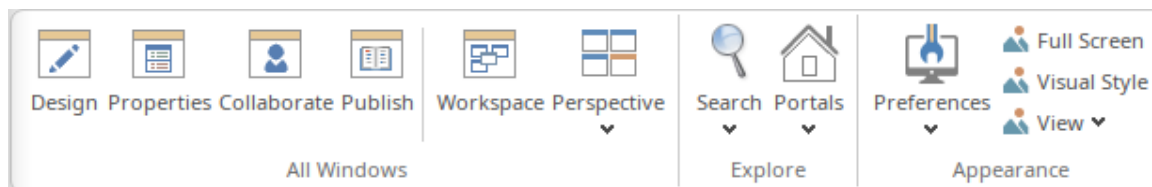
# Démarrage

Démarrer avec un nouvel outil est souvent l'un des défis les plus difficiles, mais Enterprise Architect facilite cette tâche en fournissant un certain nombre de facilités pour aider le nouveau venu à utiliser l'outil. Enterprise Architect est une application volumineuse et multi-fonctions et l'étendue de sa couverture peut submerger une personne novice dans le programme, mais heureusement, une solution à ce problème a été intégrée dans la conception.

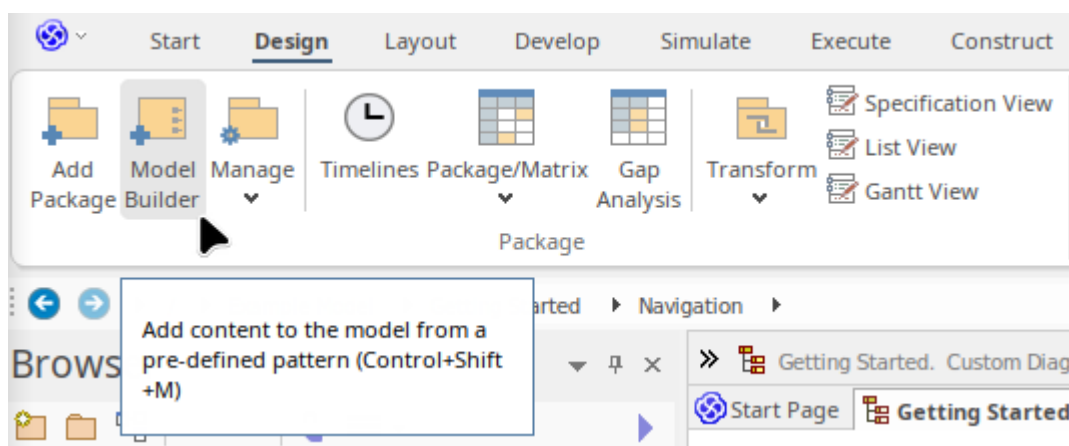
Perspectives peuvent être utilisées pour limiter la fonctionnalité à un domaine spécifique, comme l'ingénierie système, ce qui permet à un ingénieur système ou à un responsable de se lancer facilement. Un utilisateur a toujours la possibilité d'utiliser d'autres fonctionnalités qui pourraient être utiles, telles que Modélisation Stratégique, la cartographie mentale, l'ingénierie de code et bien plus encore, simplement en modifiant Perspectives, le tout sans avoir à ouvrir un autre outil. Il convient de noter que Perspectives existent pour un large éventail de disciplines modélisation supporte par Enterprise Architect. Pour plus d'informations, consultez la rubrique d'aide [Model Perspectives](#).



L'utilisateur dispose également d'une grande flexibilité pour personnaliser son propre environnement et l'interface utilisateur en définissant des préférences et en sélectionnant des espaces de travail et des styles visuels. Pour plus d'informations, consultez la rubrique d'aide [Advanced Customization](#).



La configuration d'un nouveau projet est simple grâce à l'utilisation du Constructeur de Modèle Motifs (avec la documentation qui l'accompagne) qui peut être utilisé pour créer automatiquement une structure de projet MBSE pour vous aider à démarrer. Constructeur de Modèle peut ensuite être utilisé pour créer n'importe quel nombre de diagrammes SysML au fur et à mesure que le modèle est développé et que les espaces de problèmes et de solutions sont étoffés.

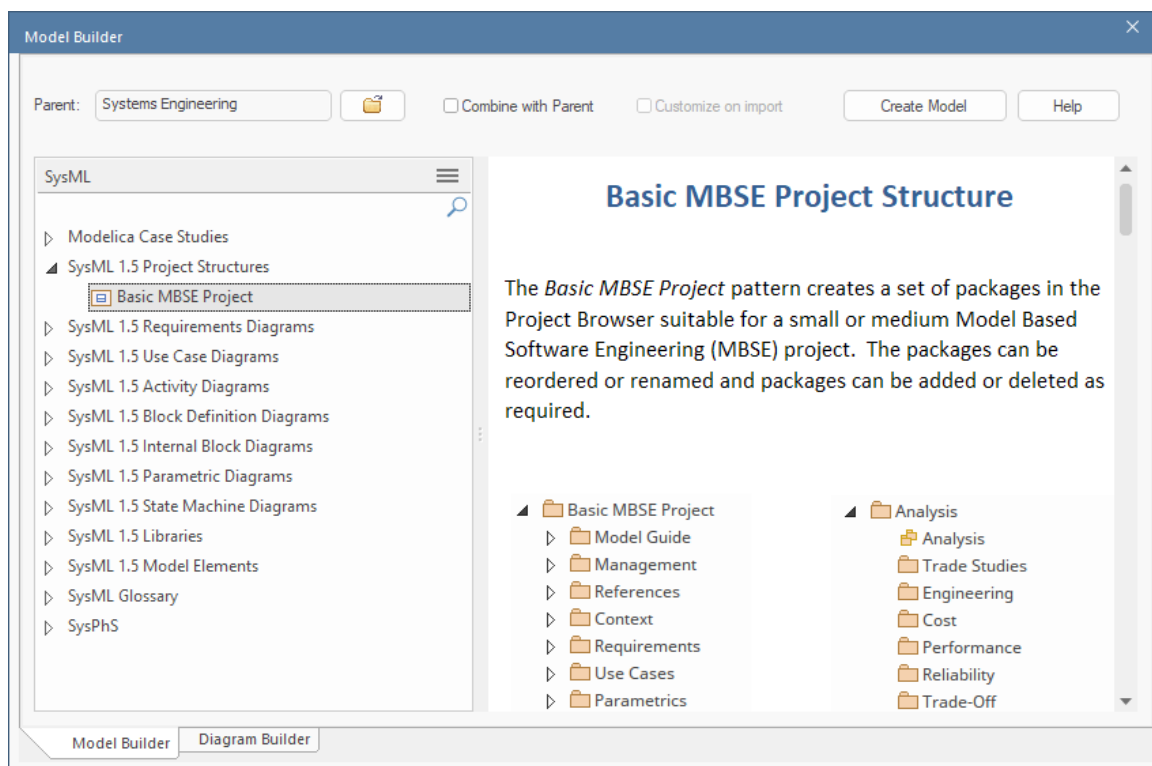


Toutes ces facilités facilitent la prise en main d'un nouveau venu, l'aidant à devenir un membre productif d'une équipe et à commencer à contribuer aux modèles rapidement et sans délai. Un ingénieur novice sera surpris de constater à quel point il peut être productif par rapport à un travail avec des outils de modélisation textuels ou d'autres outils plus rudimentaires. Vous rencontrerez des défis tout au long du parcours, car vous repousserez vos limites et celles de l'outil, mais un système d'aide approfondi, une grande communauté d'utilisateurs, des forums complets, un site communautaire et des services support de premier ordre rendront le parcours facile et instructif.

## Configuration d'une structure Modèle

Enterprise Architect a été conçu dès le départ comme un outil de productivité, et la mise en place d'une structure de modèle - parfois une tâche ardue pour le débutant et fastidieuse pour l'utilisateur expérimenté - est simplifiée dans Enterprise Architect par l'utilisation du Constructeur de Modèle .

La structure d'une nouvelle initiative (projet) peut être créée à l'aide du Constructeur de Modèle , qui créera une structure de projet entière qui pourra être personnalisée lors de l'importation, fournissant tous les Paquetages prêts à démarrer le projet.



La structure du référentiel est un sujet qui sera abordé dans une rubrique ultérieure, car elle est essentielle au succès d'une approche d'ingénierie basée sur les modèles pour Ingénierie des Systèmes . Nous apprendrons plus tard que Paquetages sont des unités importantes dans l'organisation et la maintenance d'un référentiel de modèles, et il existe une rubrique entière consacrée au sujet de l'utilisation Paquetages pour structurer le référentiel. Pour plus d'informations, consultez la rubrique d'aide [The Model Builder](#) .

# Adaptation de l'application pour MBSE

Enterprise Architect est un outil doté d'une multitude de fonctionnalités, ce qui explique en partie sa popularité en tant qu'outil de modélisation de systèmes de toutes sortes. Pour garantir que l'outil apporte le plus d'avantages à une organisation, une équipe, un projet ou un individu, une certaine personnalisation de l'interface en fonction de l'objectif de modélisation permettra à toutes les parties d'obtenir les meilleurs résultats. La plupart des paramètres peuvent être modifiés d'un simple clic sur un bouton, transformant l'outil pour l'adapter à l'objectif, qui - pour nous - consiste à collaborer sur les projets Ingénierie Systèmes Modèles Basée .

Nous allons examiner un certain nombre d'endroits où nous pouvons modifier l'application d'un outil de modélisation générique à un outil d'ingénierie système. Nous allons examiner ces sujets.

## Sélection d'une perspective

La sélection d'une perspective est similaire à l'application d'un filtre sur une lentille optique. Elle permet au modélisateur de voir uniquement les facilités d'application pertinentes pour cette perspective - dans notre cas, le Systems Modeling Language (SysML) .

## Sélection d'un espace de travail

La sélection d'un espace de travail est importante car elle permet à l'utilisateur de s'assurer que les fenêtres, rubans, barres d'outils et autres éléments visuels offrent un environnement de travail efficace et un accès facile aux facilités importantes dont il a besoin.

## Styles Visuels

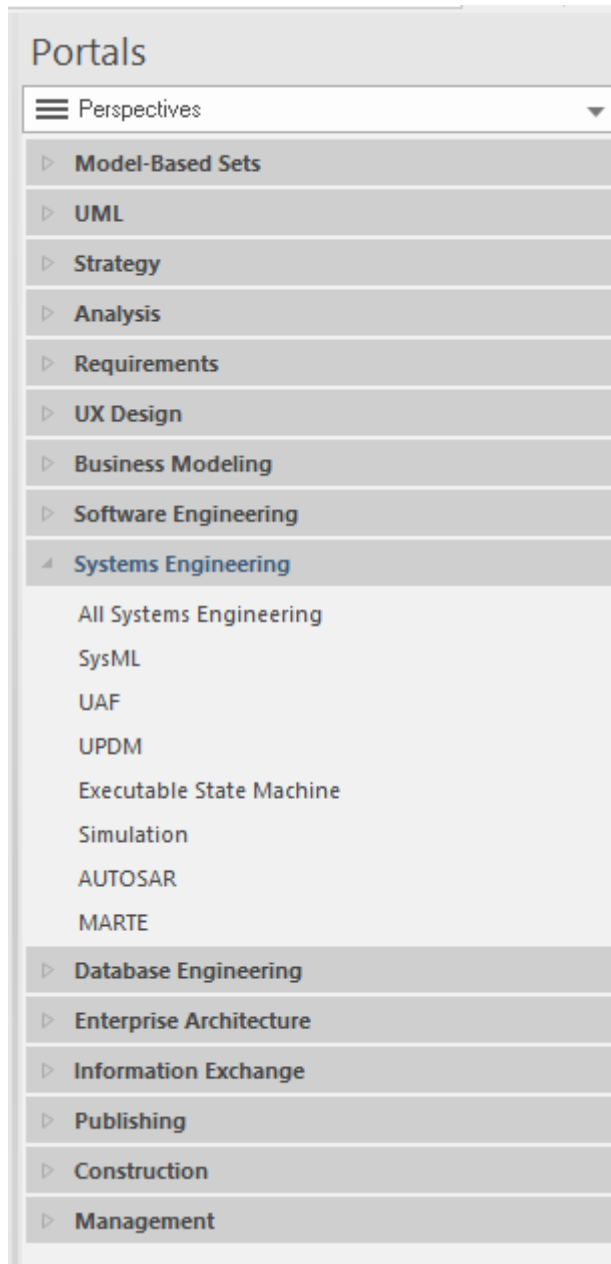
Les styles visuels fournissent une série d'options pour l'apparence de l'application, y compris des éléments tels que les couleurs et les positions des onglets.

## Définition des préférences

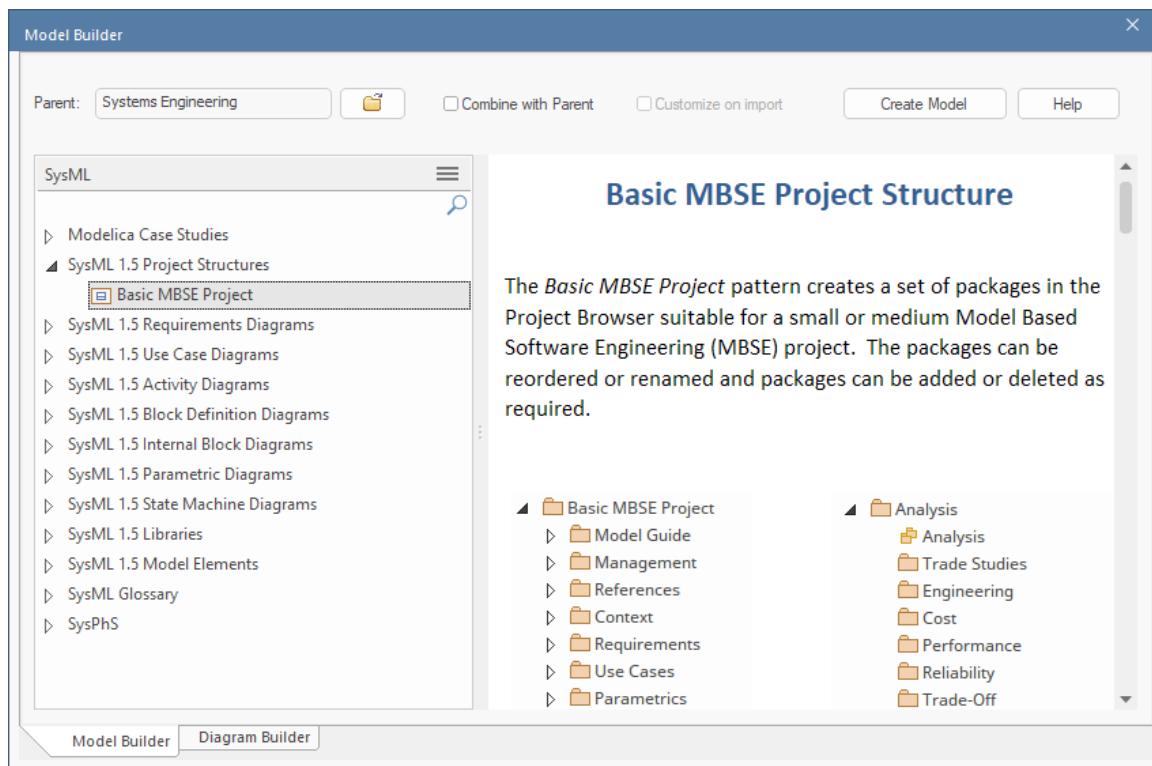
Les préférences offrent une large gamme d'options permettant de personnaliser l'apparence et le fonctionnement Enterprise Architect depuis les paramètres généraux tels que les options de la fenêtre Navigateur jusqu'aux options Diagramme , Objets et Ingénierie. De nombreuses options s'appliquent à un utilisateur individuel, tandis que d'autres concernent l'ensemble du référentiel.

## Définir une perspective

Enterprise Architect est un outil doté de fonctionnalités pour un large éventail de disciplines, méthodes, langages et cadres. Perspectives permettent à l'utilisateur de sélectionner une facette de l'outil qui lui permet de se concentrer sur un sous-ensemble particulier des fonctionnalités et facilités de l'outil. Le groupe de Perspectives Ingénierie des Systèmes constitue un point de départ naturel pour les ingénieurs systèmes, mais à tout moment, si vous décidez d'utiliser d'autres facilités dans l'outil, vous pouvez simplement changer Perspectives et l'outil changera pour se concentrer sur la zone sélectionnée.



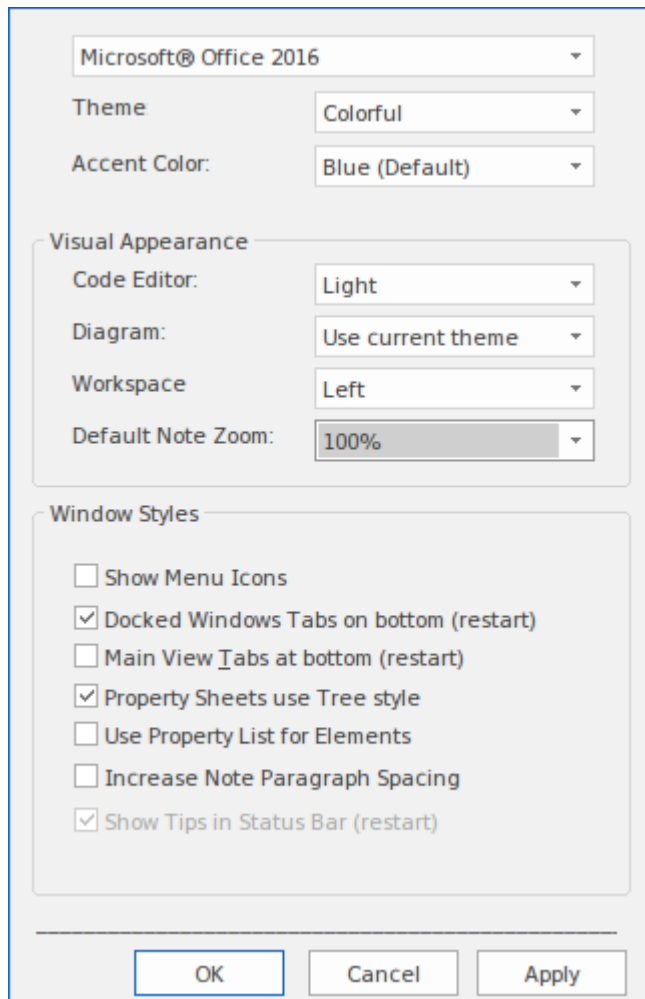
La sélection de l'une des Perspectives Ingénierie des Systèmes modifiera les outils pour se concentrer sur l'aspect sélectionné d' Ingénierie des Systèmes . Par exemple, le choix de la perspective SysML affichera une série de motifs de modèle donnant à l'utilisateur un coup de pouce en lui permettant de charger un motif pour un fragment de modèle standard ou diagramme . La dialogue « Nouveau Diagramme » affichera également uniquement les types diagramme SysML.



Il existe également une facilité pratique pour un utilisateur de créer un nombre illimité de ses propres Perspectives , en ajoutant des ensembles de technologies à chaque perspective. Cela permet à un modélisateur dont la préoccupation principale est diagrammes SysML d'ajouter d'autres facilités telles que des modèles stratégiques, diagrammes Kanban et des dizaines d'autres mécanismes utiles de création de diagrammes et modélisation . Pour plus d'informations, consultez la rubrique [Model Perspectives](#) .

## Sélection d'un style visuel

Chaque modélisateur a ses propres préférences concernant le jeu de couleurs et le style de l'interface utilisateur. Enterprise Architects permet de les définir et de les enregistrer pour chaque utilisateur, ce qui rend l'application plus attrayante. Par exemple, certains modélisateurs voudront un jeu de couleurs sombres et d'autres préféreront un jeu de couleurs claires ou colorées.

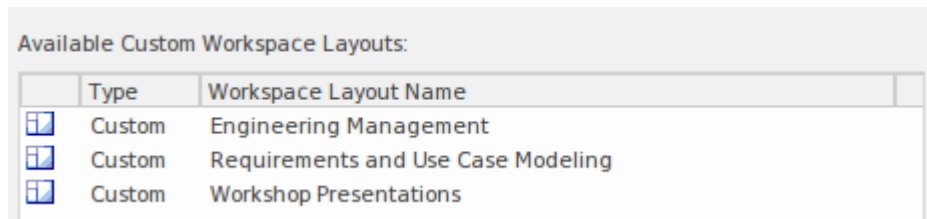





The screenshot shows a settings dialog box for 'Visual Appearance'. At the top, there is a dropdown menu for 'Microsoft® Office 2016'. Below it are three more dropdown menus: 'Theme' set to 'Colorful', 'Accent Color' set to 'Blue (Default)', and 'Visual Appearance' set to 'Light'. Under the 'Visual Appearance' section, there are four more dropdown menus: 'Code Editor' set to 'Light', 'Diagram' set to 'Use current theme', 'Workspace' set to 'Left', and 'Default Note Zoom' set to '100%'. Below these is a 'Window Styles' section with a list of checkboxes: 'Show Menu Icons' (unchecked), 'Docked Windows Tabs on bottom (restart)' (checked), 'Main View Tabs at bottom (restart)' (unchecked), 'Property Sheets use Tree style' (checked), 'Use Property List for Elements' (unchecked), 'Increase Note Paragraph Spacing' (unchecked), and 'Show Tips in Status Bar (restart)' (checked). At the bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Apply'.

Il existe ici une gamme d'options, notamment la définition de la position des onglets de la fenêtre principale, la taille du texte dans la fenêtre notes, et bien plus encore. La définition du style visuel permet de personnaliser l'environnement de modélisation et de mettre à l'aise les différents modélisateurs tout en conservant des modèles cohérents et rigoureux. Pour plus d'informations, consultez la rubrique d'aide [Visual Styles](#).

## Sélection d'un espace de travail

Enterprise Architect dispose d'un moyen pratique de modifier rapidement la disposition de l' Interface Utilisateur pour faciliter certaines tâches ou méthodes de travail. Pour cela, il suffit de sélectionner un espace de travail qui modifiera les fenêtres et les outils visibles, afin de fournir la manière la plus efficace de travailler en fonction de la tâche. Par exemple, un espace de travail est défini pour Ingénierie des Systèmes Simulations, un pour Use Case Modélisation et un autre pour Tester . Vous pouvez également définir autant de dispositions d'espace de travail que vous le souhaitez, en ouvrant des fenêtres et des outils et en les positionnant dans une disposition qui facilite le travail sur une tâche ou un ensemble de tâches particulier, puis en les enregistrant. Dans cet exemple, un modélisateur a défini trois dispositions d'espace de travail personnalisées. Pour plus d'informations, consultez la rubrique [Workspace Layouts](#) .



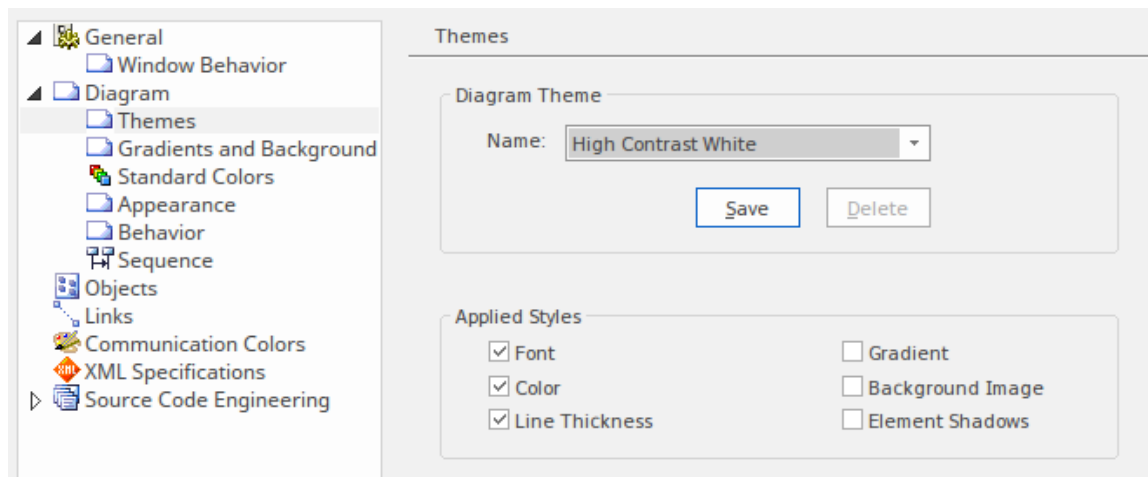
Available Custom Workspace Layouts:		
	Type	Workspace Layout Name
	Custom	Engineering Management
	Custom	Requirements and Use Case Modeling
	Custom	Workshop Presentations



## Définition des préférences

Enterprise Architect dispose d'un formidable ensemble de préférences, dont certaines peuvent être définies pour l'ensemble du référentiel et d'autres pour chaque utilisateur. Elles permettent d'adapter l'application à un ingénieur individuel ou à une équipe entière. Pour plus d'informations, consultez la rubrique [User Preferences](#).

Ce diagramme montre comment les thèmes diagramme peuvent être définis et les éléments du style peuvent être spécifiés, y compris les polices, les couleurs, l'épaisseur des lignes et les dégradés.



# Importation de matériel existant

Une fois Enterprise Architect configuré, il est probable que vous disposiez encore de certains artefacts de projet existants sous forme de diagrammes, de documents, de feuilles de calcul et d'éléments dans d'autres formats. La plupart d'entre eux peuvent être facilement importés dans Enterprise Architect ou référencés à partir de l'outil. L'outil offre également une connectivité côté serveur étendue à une large gamme d'autres outils, notamment des outils Gestion des Exigences tels que DOORS Next Generation, des outils de gestion de projet tels que Wrike et des outils d'implémentation de projet tels que Jira, via Pro Cloud Server (un composant côté serveur sous licence séparée).

## Importation de feuilles de calcul

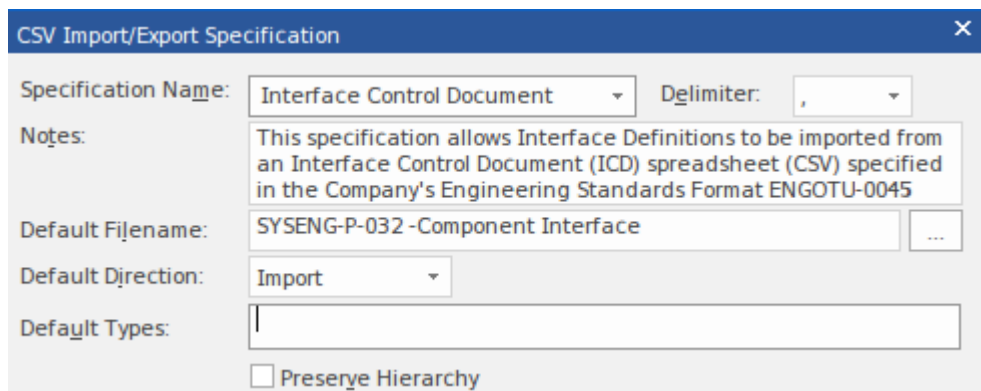
Les feuilles de calcul sont couramment utilisées comme conteneur à usage général pour une large gamme de données de projet numériques et textuelles. Les utilisations incluent :

- Exigences
- Analyse des parties prenantes
- Planification
- Feuilles de Route
- Sous-systèmes
- Composants
- Définitions d'interface
- Gestion des Tâches

Bien que le tableur soit un outil très familier, il manque de nombreuses fonctionnalités rigoureuses et utiles d'une plate-forme de gestion de l'information telle qu'Enterprise Enterprise Architect, notamment :

- Collaboration,
- Diagramme
- Traçabilité,
- Lignes de base,
- Visualisations,
- Simulations
- Versions et plus.

Enterprise Architect intègre un support pour toutes ces formes d'informations et bien d'autres qui sont généralement stockées dans des feuilles de calcul. L'outil est également doté de facilités pratiques pour importer et exporter les données de la feuille de calcul à l'aide du format de fichier CSV.



Le scénario le plus courant est que les informations des feuilles de calcul soient importées dans Enterprise Architect,

puis la feuille de calcul peut être mise hors service et les ressources d'information peuvent être gérées dans Enterprise Architect à partir de ce moment.

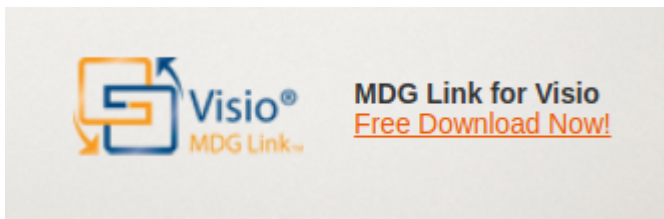
Il peut y avoir des situations dans lesquelles l'utilisation des aspects d'analyse numérique d'une feuille de calcul peut s'avérer utile. Enterprise Architect fournit un outil pratique pour exporter des informations au format de fichier CSV afin de les importer dans une feuille de calcul. Pour plus d'informations, consultez la rubrique [Import and Export Spreadsheets](#).

## Importation de Diagrammes Visio

Microsoft Visio est couramment utilisé par les équipes d'ingénierie, souvent parce qu'il n'existe pas d'autres outils plus sophistiqués et qu'il sert à créer des diagrammes à usage général. Il est courant qu'une équipe qui adopte Enterprise Architect comme plate-forme d'ingénierie de choix dispose d'une collection de diagrammes Visio préexistants. Tous ces diagrammes peuvent être importés dans Enterprise Architect mais les résultats sont plus efficaces lorsque ces diagrammes Visio ont été construits de manière cohérente ou à l'aide de palettes industrielles standard plutôt que de formes géométriques libres telles que des carrés et des cercles qui n'ont pas de signification commune. Une fois importés, les diagrammes peuvent être massés et mis à jour pour faire partie du référentiel, et les diagrammes Visio d'origine peuvent être mis hors service.

Enterprise Architect fournit un outil gratuit qui peut être utilisé pour se connecter à un moteur MS Visio et importer des diagrammes sélectionnés dans le référentiel.

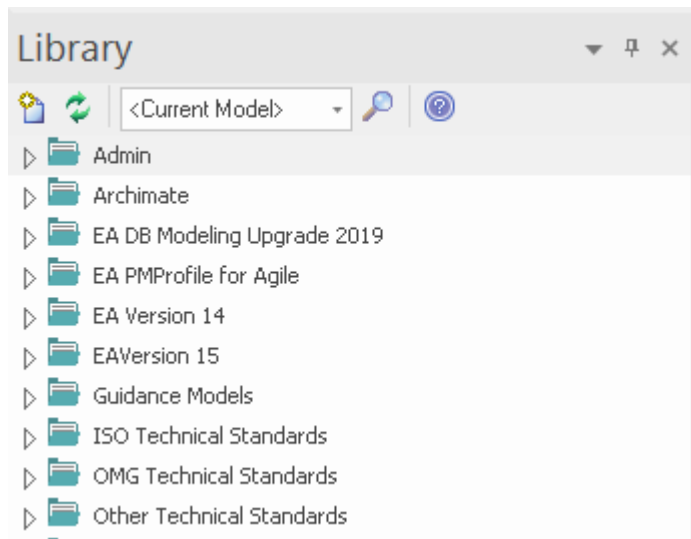
Il faut décider si les diagrammes dans Visio doivent être désactivés et si Enterprise Architect doit désormais gérer les diagrammes. Le fait d'avoir les diagrammes dans le référentiel offre une grande puissance, car les éléments des diagrammes peuvent être liés à d'autres éléments du référentiel. Pour plus d'informations, consultez la rubrique [Extensions - MDG Technologies](#).



## Utiliser la Bibliothèque d'Équipe

Même si une équipe d'ingénierie a adopté l'Ingénierie Systèmes Modèles Basée, il est probable qu'il existe encore une gamme de documents et de ressources Web essentiels à la gestion et au développement de solutions d'ingénierie. Enterprise Architect propose une approche pragmatique de ce besoin en incorporant une fonctionnalité Bibliothèque d'Équipe où les documents et les ressources Web (à la fois locaux et distants) peuvent être rassemblés en tant que références.

Toutes les références répertoriées dans la Bibliothèque d'Équipe peuvent être incluses dans un diagramme en tant qu'Artefact interne ou externe, mais il est plus pratique de les importer ou de les référencer. Pour plus d'informations, consultez la rubrique d'aide [The Model Library](#).



## Intégration Microsoft Office

Enterprise Architect a la capacité de s'intégrer à la suite d'applications Microsoft Office à l'aide de MDG Link pour Microsoft Office, ce qui facilite l'échange d'informations entre n'importe quel modèle Enterprise Architect et MS Powerpoint, MS Word et MS Excel . Il existe des options pour importer, exporter et synchroniser le contenu.

### Microsoft PowerPoint

L'intégration PowerPoint permet d'accéder facilement au référentiel de modèles d' Enterprise Architect dans les présentations PowerPoint. Vous pouvez insérer des références au modèle, utiliser des noms d'éléments de modèle hyperliés, insérer diagrammes sous forme d'images et tabuler le contenu Paquetage sur des diapositives.

### Microsoft Excel

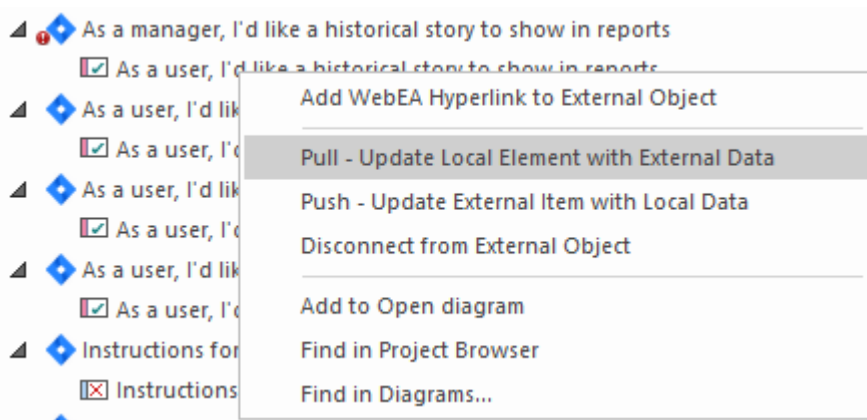
L'outil Microsoft Excel Importer vous permet d'importer le contenu des classeurs Microsoft Excel dans Enterprise Architect sous forme d'éléments de modèle. Cela inclut l'importation de données de feuille de calcul sous forme d'éléments UML , de connecteurs, d'attributs et d'opérations.

### Microsoft Word

L'outil Microsoft Word Importer intègre Exigences , les cas d'utilisation, Processus , les classes et d'autres données des documents Microsoft Word dans Enterprise Architect sous forme d'éléments de modèle. L'outil Microsoft Word Importer propose une approche étape par étape qui vous aide à mapper des éléments tels que des sections, tableaux et des paires nom-valeur délimitées aux éléments et propriétés Enterprise Architect , y compris la définition Valeur Étiquetés personnalisés.

## Intégration avec des outils externes

Enterprise Architect fournit une interface (dans le cadre de Pro Cloud Server ) pour connecter vos référentiels de modèles à des outils externes. Cela permet à Enterprise Architect de synchroniser les éléments des outils externes avec les vues des éléments dans Enterprise Architect , ce qui est particulièrement utile si Enterprise Architect et un autre outil partagent un intérêt pour des types d'informations particuliers. Un exemple est l'intégration avec le produit DOORS Next Generation (NG), où les exigences modélisées dans DOORS peuvent être visualisées dans Enterprise Architect , et des substituts locaux des éléments peuvent être placés sur diagrammes et liés à un certain nombre d'autres éléments modélisation , y compris des stratégies, des études commerciales, des cas d'utilisation et des composants. (Il existe également une facilité intégrée au modèle disponible pour se connecter à des versions antérieures de DOORS .)



Il existe une large gamme d'intégrations disponibles et les équipes peuvent créer leurs propres intégrations à l' facilité du service Open Services for Lifecycle Collaboration (OSLC), disponible dans le cadre de Pro Cloud Server . Pour plus d'informations, consultez la rubrique d'aide [Integrate Data from External Providers](#) .

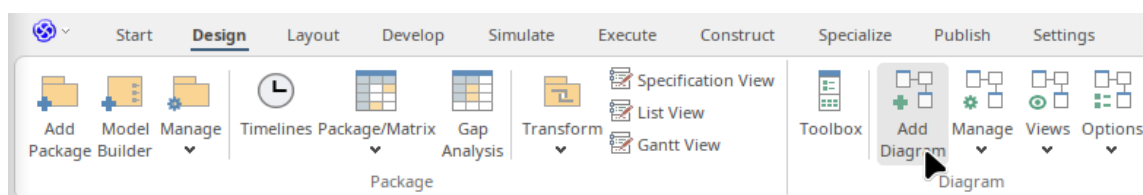
# Création Diagrammes d'éléments et Relations

Une fois la structure du modèle configurée et l'application adaptée à vos besoins, y compris la sélection de la perspective SysML et d'un espace de travail approprié, vous êtes prêt à commencer à créer des éléments et diagrammes . Bien qu'il soit possible de créer un élément sans créer au préalable un diagramme il est courant de créer d'abord un diagramme comme canevas pour la visualisation des éléments. La première chose à faire est de choisir un emplacement pour le diagramme dans la fenêtre Navigateur . Par exemple, vous pouvez définir l' architecture fondamentale de votre système et avoir défini un Paquetage appelé « Sous-systèmes ». En sélectionnant ce Paquetage vous indiquez Enterprise Architect qu'il s'agit de l'emplacement où vous souhaitez insérer un nouveau diagramme de sous-systèmes.

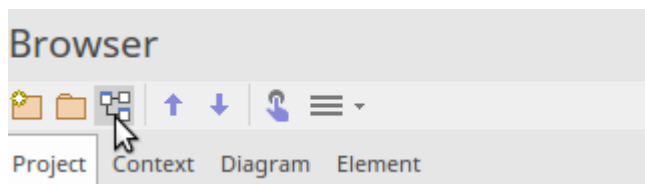
## Créer un Nouveau Diagramme

Enterprise Architect est un outil flexible et offre plusieurs façons d'insérer un nouveau diagramme , notamment :

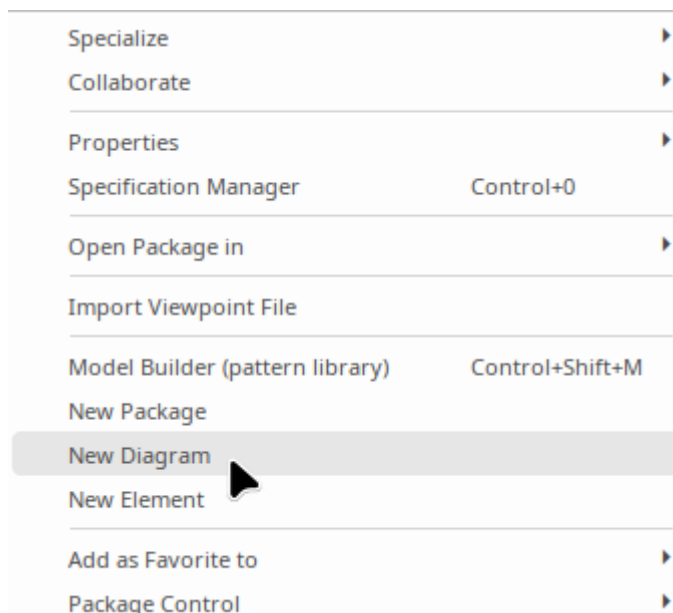
Sélection des rubans :



Sélection depuis la barre d'en-tête de la fenêtre Navigateur :

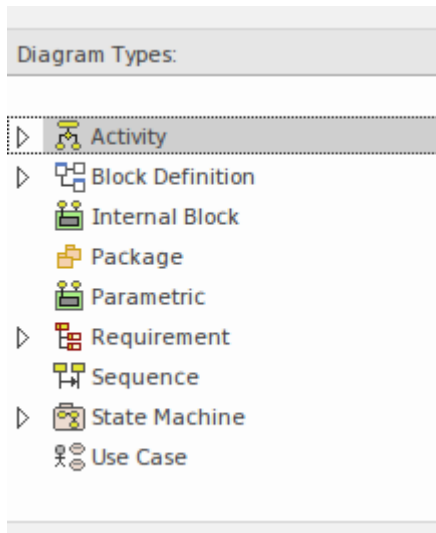


Sélection depuis le menu contextuel (clic droit) :



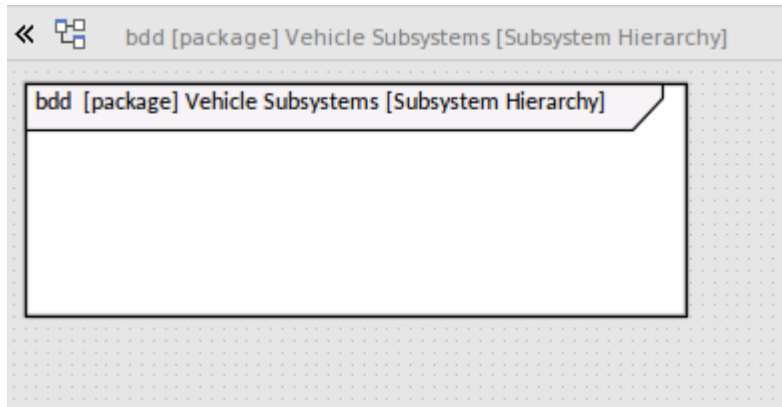
Quelle que soit la méthode choisie, vous pourrez sélectionner le type diagramme SysML à partir du panneau « Types

Diagramme » de la dialogue « Nouveau Diagramme ».



Continuons à créer un diagramme de définition Bloc pour représenter le sous-système. Sélectionnez le diagramme de définition Bloc comme type diagramme et entrez un nom approprié. Une fois que vous avez cliqué sur le bouton OK, un nouveau diagramme BDD (vierge) sera créé et la boîte à outils Interrompre lorsqu'une Variable Change de Valeur s'affichera, prête à vous permettre, ou à un membre de votre équipe, de créer des éléments et des relations.

Enterprise Architect crée un canevas diagramme avec un cadre visible qui représente la bordure du diagramme. Le cadre diagramme est inclus car certains utilisateurs préfèrent le voir, mais il peut être masqué sans perte de sens ou de conformité ; une fois masqué, le canevas devient alors le cadre et les informations d'en-tête sont contenues en haut du canevas. Le cadre peut être inclus dans diagrammes enregistrés ou publiés en choisissant cette option dans la dialogue « Préférences ».



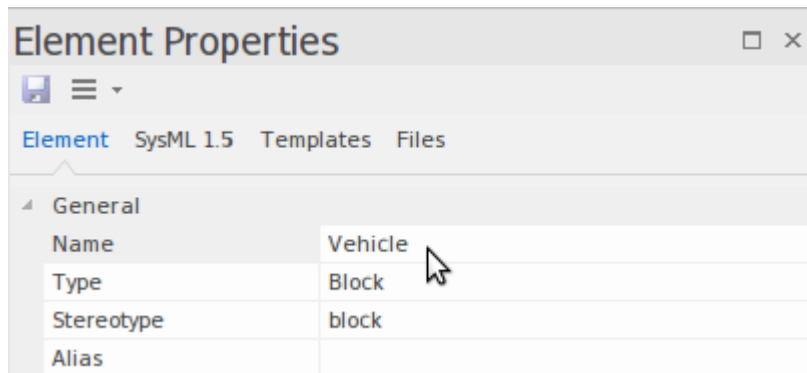
## Ajout d'éléments à un Diagramme

Une fois le nouveau diagramme ouvert, vous êtes prêt à commencer à créer des éléments et des relations pour décrire les sous-systèmes. Il existe essentiellement deux types d' Object qui peuvent être ajoutés à un diagramme :

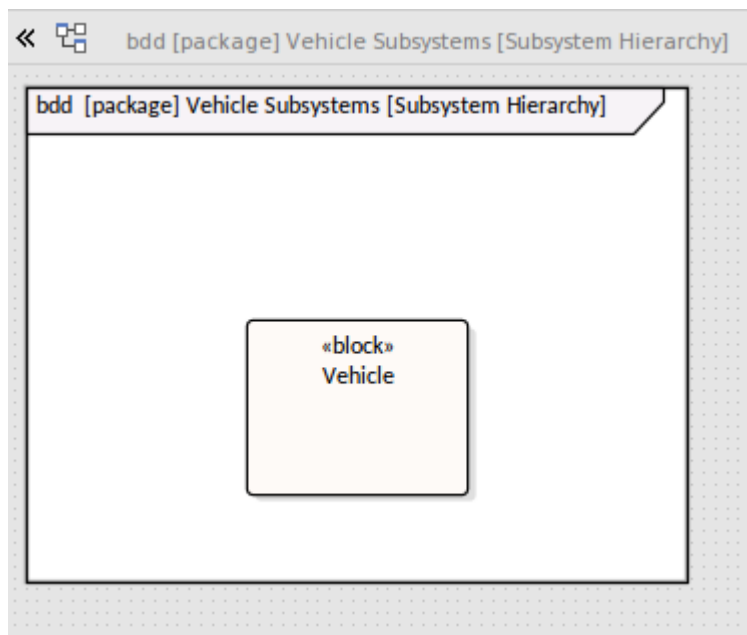
- Nouveaux éléments - *Créés en faisant glisser un élément de la boîte à outils et en le déposant sur le canevas diagramme*
- Éléments existants - *Placés sur le diagramme par glisser-déposer d'un élément depuis la fenêtre Navigateur*

Si vous démarrez un nouveau projet et venez de configurer votre référentiel, vous n'aurez généralement pas d'éléments dans la fenêtre Navigateur. Vous utiliserez donc davantage la première option et créez des éléments à partir de la boîte à outils. Au fur et à mesure de l'avancement de votre projet, il deviendra plus courant d'utiliser la deuxième option et de faire glisser les éléments existants depuis la fenêtre du Navigateur.

Nous allons créer un certain nombre de blocs. Tout d'abord, nous avons besoin d'un Bloc pour représenter l'ensemble du véhicule. Nous allons donc faire glisser et déposer un élément Bloc de la boîte à outils sur le canevas diagramme . L'outil redimensionnera le cadre pour inclure le Bloc quel que soit l'endroit où vous l'avez placé sur le canevas. L'élément recevra un nom par défaut de « Block1 ». Maintenant, en utilisant la fenêtre Propriétés , généralement ancrée sur le côté du diagramme , changez le nom de l'élément en « Véhicule » en tapant sur le nom par défaut « Block1 ».



Cela modifiera le nom de l'élément dans la fenêtre Navigateur et dans le diagramme . En revenant au diagramme vous verrez le Bloc nouvellement ajouté avec le nom « Véhicule » inclus dans le cadre diagramme .



Nous pourrions maintenant utiliser la même méthode pour ajouter une série de blocs pour représenter chacun des sous-systèmes.

## Ajout Relations à un Diagramme

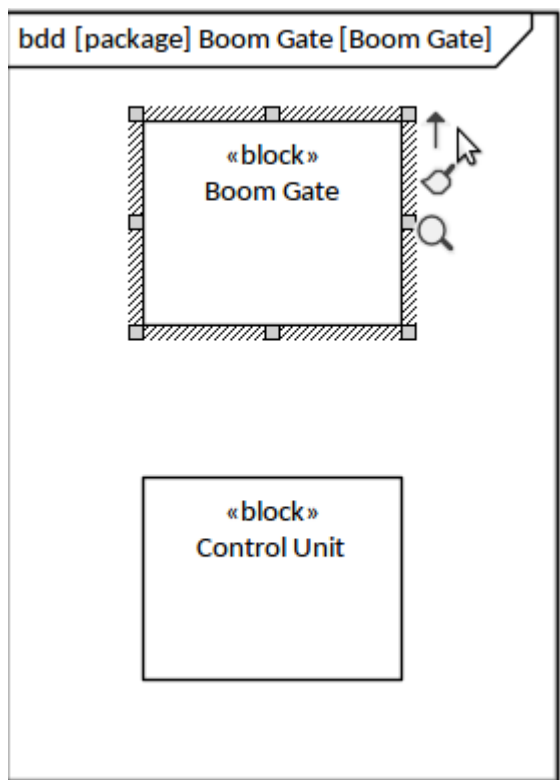
Une fois que vous avez ajouté deux éléments ou plus, vous pouvez les connecter à l'aide de relations, qui fournissent le lien sémantique entre les différents éléments du modèle. Par exemple, un élément Bloc peut être connecté à un autre élément Bloc à l'aide d'une relation *Part Association* . Il existe deux manières principales d'ajouter des connecteurs à un diagramme :

1. Quick Linker - *un dispositif diagramme intuitif lancé en faisant glisser un lien entre la flèche Quick Linker (en haut à droite de l'élément) et un autre objet diagramme*
2. Items de la boîte à outils : *les connecteurs peuvent être sélectionnés dans la boîte à outils, puis glissés entre deux*

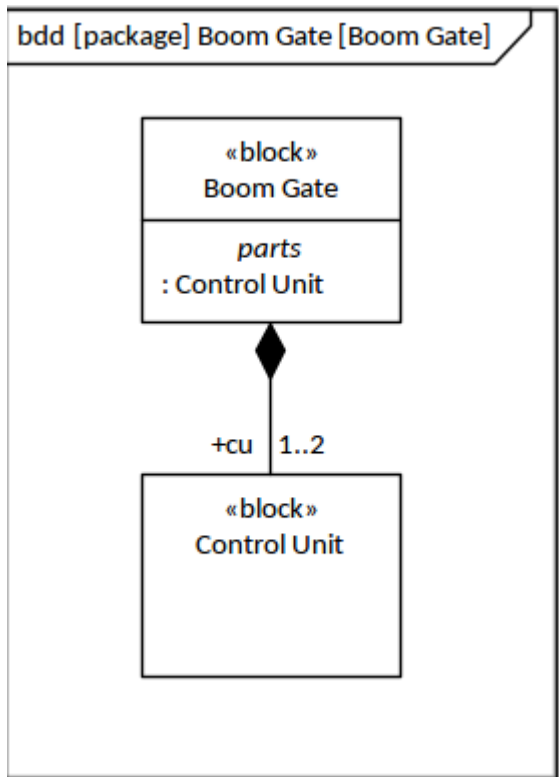


objets diagramme .

Quelle que soit la méthode utilisée, le connecteur spécifié sera dessiné entre les deux éléments. Il faut veiller à faire glisser le connecteur dans la bonne direction. Par exemple, la relation Part Association doit être déplacée du Bloc situé à l'extrémité de la pièce vers le Bloc situé à l'extrémité entière . Cela permettra de s'assurer que le petit marqueur en forme de losange à l'extrémité de la relation est positionné à l'extrémité correcte, indiquant la relation ensemble-partie.



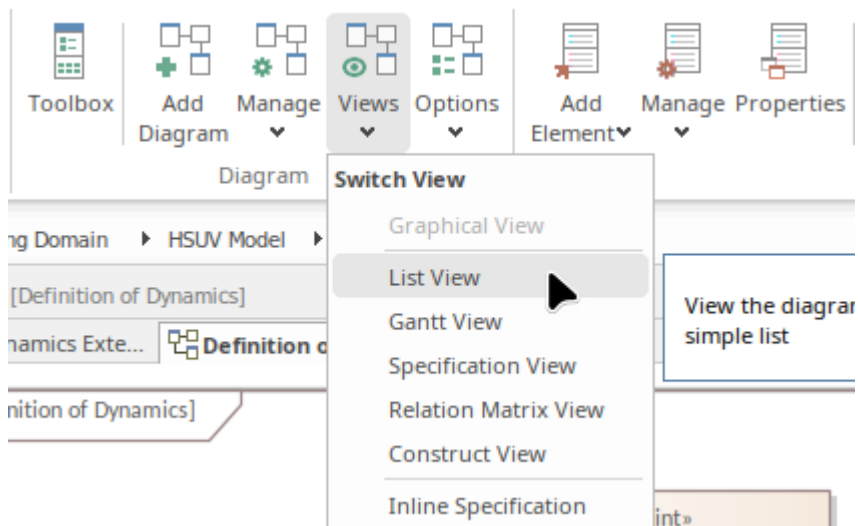
Quelle que soit la méthode utilisée, le résultat sera une relation d'exigence d'information reliant les deux blocs. La direction et le style du connecteur peuvent être modifiés et un nombre quelconque de points de cheminement peut être ajouté pour l'acheminer différemment au fur et à mesure du développement du modèle. Ce diagramme montre la relation ajoutée où le modélisateur a également ajouté un nom de rôle (+cu) et une multiplicité (1..2), indiquant qu'une barrière de flèche doit avoir au moins une unité de contrôle mais peut en avoir jusqu'à deux. Si un modélisateur devait ajouter par inadvertance le connecteur dans la mauvaise direction, il peut facilement l'inverser en accédant aux options du sous-menu *Avancé* du menu contextuel du connecteur.



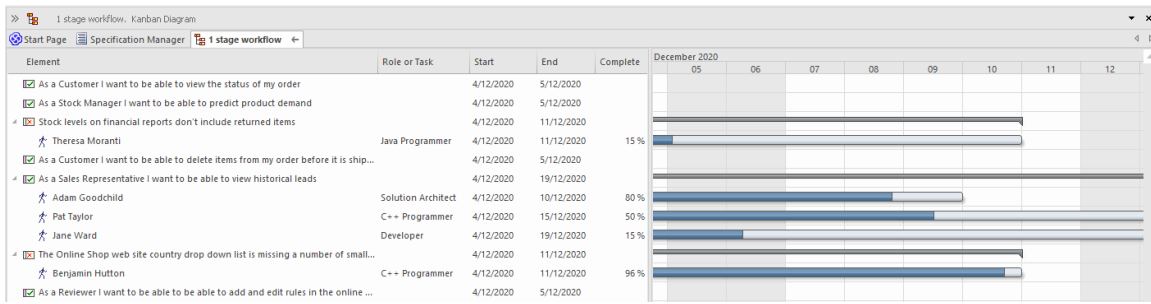
## Visualisation des modèles

Vous avez peut-être découvert SysML en lisant des documents d'introduction dans des manuels ou dans notes de cours dans le cadre d'une qualification tertiaire ou dans des pages en ligne. Une grande partie de ce contenu présente SysML comme un langage de diagrammes , mais Enterprise Architect élargit les horizons du langage - et de modélisation en général - en fournissant une variété de façons de visualiser les informations dans le référentiel. Les diagrammes jouent toujours un rôle important, mais dans de nombreuses circonstances, il existe des moyens plus convaincants de visualiser ou de travailler avec le contenu du référentiel, notamment :

- Spécification Vue - une vue de feuille de calcul ou de document pour ceux qui sont plus familiers avec ce mode de travail
- Liste Vue - une vue basée sur un tableau similaire à une feuille de calcul où les propriétés de plusieurs éléments peuvent être visualisées
- Vue de traçabilité - une vue hiérarchique et graphique où les relations entre les éléments peuvent être visualisées à n'importe quel niveau d'imbrication
- Matrice d'Analyse des Écarts - montre les éléments sources et cibles et les lacunes dans les modèles architecturaux
- Matrice de relations - une vue basée sur une grille avec deux axes contenant des ensembles d'éléments
- Tableau State Vue - une vue disponible pour diagrammes Statemachine pour afficher les transitions d'état dans un tableau
- Diagramme de Gantt Vue - affichez une chronologie lorsque des ressources ont été affectées
- Recherches - afficher les listes d'éléments conformes à un ensemble de critères intégrés ou définis par l'utilisateur
- Publications - afficher les éléments d'une publication à l'aide gabarits intégrés ou définis par l'utilisateur
- Vue graphique - visualiser les éléments sous forme de diagramme (il s'agit de la vue par défaut)



Ce diagramme montre un diagramme de Gantt qui a été créé automatiquement à partir des informations sur les ressources saisies par rapport aux éléments visualisés sur un diagramme Kanban. Pour plus d'informations, consultez la rubrique d'aide [Gantt View](#) .



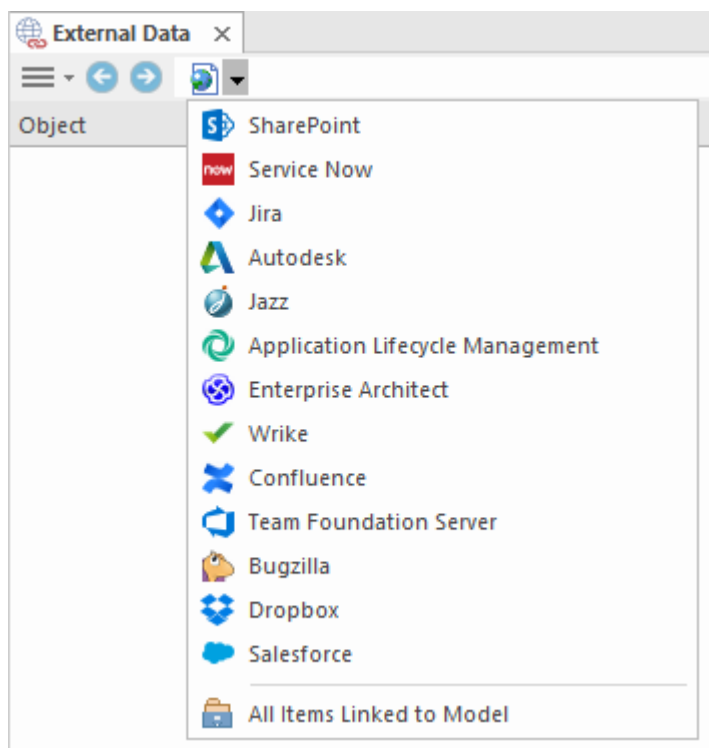
## Synchronisation avec des données externes

Bien qu'Enterprise Architect soit un outil central pour la gestion des projets Ingénierie Systèmes Modèles Basée , il existe probablement un certain nombre d'autres outils qu'un bureau d'ingénierie aura mis en place ou acquerra pour garantir qu'un projet atteigne ses objectifs. Il peut s'agir d'outils Gestion de Projet , d'outils de visualisation, de catalogues Gestion des Exigences , de systèmes de gestion de configuration et de logiciels de suivi des problèmes.

Enterprise Architect , via Pro Cloud Server (un composant côté serveur sous licence séparée), fournit une intégration bidirectionnelle avec un grand nombre d'outils, notamment :

- Portes Next Generation - utilisées pour Gestion des Exigences
- Wrike - utilisé à des fins générales Gestion de Projet
- Jira - utilisé pour le suivi des problèmes
- ServiceNow - utilisé pour la gestion de la configuration
- SharePoint - utilisé pour la gestion des documents
- Team Foundation Server - utilisé pour Contrôle de Version dans la discipline logicielle

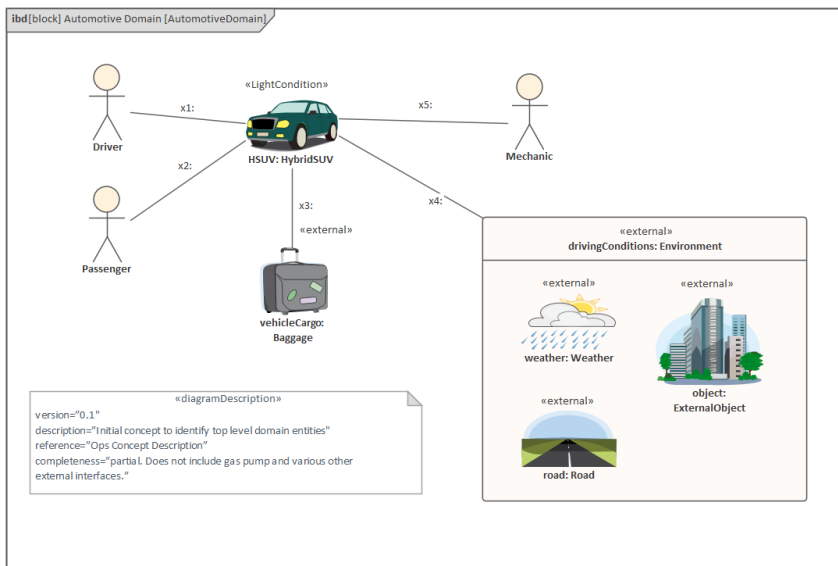
Les représentations des éléments provenant de ces sources de données externes peuvent être incluses dans diagrammes et reliées à d'autres éléments du référentiel. Cela permet à Enterprise Architect de devenir un centre d'informations techniques, en se connectant et en s'intégrant à une large gamme d'outils pour fournir une vue unique d'un projet. Cette image montre une liste actuelle de fournisseurs.



## Où nous allons

La discipline de Ingénierie des Systèmes remonte au début des années 1900 ; le terme aurait été inventé par les laboratoires Bell au début des années 1940. Une soixantaine d'années se sont écoulées avant que le besoin d'un langage dédié à modélisation des systèmes ne soit perçu en 2001, mais ce n'est qu'en 2006 que le Systems Modeling Language (SysML) a été adopté par l' Object Management Group et est devenu le langage de choix pour décrire les systèmes.

Dans le domaine de Ingénierie des Systèmes , Enterprise Architect est devenu l'outil de choix pour de nombreux leaders de l'industrie, en raison de son approche flexible, extensible et pragmatique de modélisation de systèmes complexes et de sa forte conformité avec la spécification Langage Modélisation des Systèmes .



En tant que plate-forme, Enterprise Architect offre une capacité unique de prise en charge de l'intégration de modèles stratégiques, commerciaux, d'ingénierie et technologiques, depuis les modèles de motivation jusqu'à la mise en œuvre des systèmes et support . L'outil aide l'ingénieur système à créer des modèles stratégiques - y compris diagrammes tels que le Tableau de Bord Équilibré - des modèles de capacités, des modèles tactiques - tels que l'analyse des écarts et Feuilles de Route - et des modèles opérationnels, attrayants pour les parties prenantes, notamment les cadres supérieurs, les responsables techniques, les équipes de solutions et de mise en œuvre et les ingénieurs.

## Comment cela vous aidera

Les lecteurs abordent généralement le sujet de Ingénierie Systèmes Modèles Basée avec une certaine connaissance ou expérience, même s'il s'agit de quelque chose qui a été appris lors de cours magistraux ou par une formation sur le terrain, ou peut-être en utilisant un autre outil. Les lecteurs bénéficieront de la compréhension fonctionnalités d' Enterprise Architect et des outils disponibles pour développer et gérer Ingénierie Systèmes Modèles Basée dans Enterprise Architect . Ces connaissances leur permettront d'être plus productifs en tant qu'individu et également en tant que membre d'une équipe. Le lecteur découvrira également la syntaxe et la sémantique du Langage Modélisation des Systèmes connu sous le nom de SysML, et comment il peut être utilisé pour modéliser et même simuler des systèmes d'ingénierie complexes.

## Qui en bénéficiera

Toute personne impliquée dans le développement, la gestion ou support des initiatives Ingénierie Systèmes Modèles Basée que ce soit à un :

- Niveau stratégique,
- Niveau de valeur Métier ,
- Niveau de gestion ou
- Niveau ingénieur

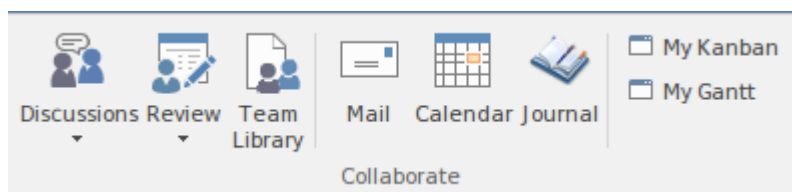
bénéficieront de la lecture de ces informations. Celles-ci couvrent un large éventail de rôles, notamment :

- Clients,
- Penseurs stratégiques,
- Haute direction,
- Gestion de l'ingénierie,
- Concepteurs et architectes de systèmes,
- Concepteurs et architectes de logiciels,
- Ingénieurs Systèmes,
- Ingénieurs en logiciel,
- Équipes de fabrication,
- Équipes de mise en œuvre,
- Personnel Support .

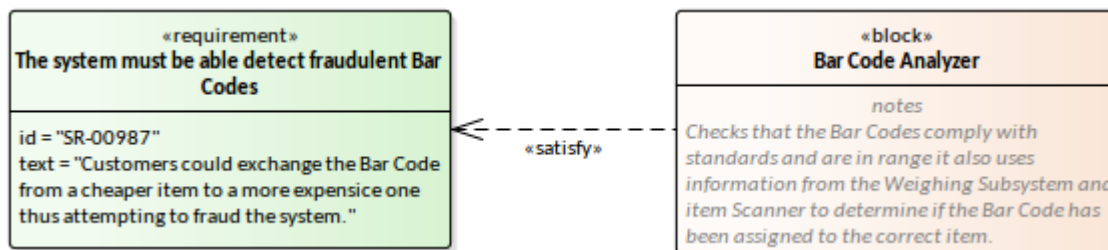
La véritable puissance de cette façon de travailler et d'utiliser Enterprise Architect comme outil réside dans l'unification des équipes interdisciplinaires. Cela permet ainsi à des personnes issues d'un large éventail de disciplines, potentiellement dispersées géographiquement et travaillant à des niveaux très différents, de la stratégie au support , partager une vision commune du produit ou du système en cours de développement.

## Ce que vous apprendrez

Ce guide vous apprendra à utiliser les riches fonctionnalités d' Enterprise Architect pour développer et gérer des initiatives Ingénierie Systèmes Modèles Basée , à écrire et à lire le Systems Modeling Language (SysML) , à créer de la documentation et à travailler en collaboration en tant que membre d'une équipe en utilisant un processus d'ingénierie formel ou informel.



Vous apprendrez quels facilités sont disponibles et quels outils doivent être utilisés pour réaliser une technique particulière et, à l'aide de liens vers des rubriques d'aide, comment les utiliser. Par exemple, quel que soit le processus ou le cadre adopté, à un moment donné, *une répartition entre les éléments structurels et Comportementale* devra être effectuée ; cette rubrique décrira la technique et la meilleure façon de l'atteindre à l'aide des outils et facilités dont vous disposez en adoptant Enterprise Architect .



## Aperçu de la documentation

Ce tableau fournit une liste et une description des sujets inclus dans cet ouvrage, donnant un aperçu du matériel.

### Une équation à quatre variables

Ce sujet décrit Ingénierie Systèmes Modèles Basée d'un point de vue mathématique comme une équation à quatre variables, à savoir : *Modélisation* , *Enterprise Architect l'outil de choix* , *le langage modélisation des systèmes* et *un processus d'ingénierie* . Les mathématiques élémentaires suggèrent que nous avons besoin de quatre équations simultanées pour résoudre un tel problème. Ce sujet présentera ces quatre variables et suggérera des moyens pour que vous, en tant qu'ingénieur ou gestionnaire, deveniez compétent dans cette discipline sans avoir besoin de résoudre les équations simultanément.

Nous verrons comment Enterprise Architect peut jouer un rôle important dans les quatre parties de l'équation et a été conçu pour vous aider à réussir vos initiatives d'ingénierie.

### Démarrage

Cette rubrique présente aux nouveaux utilisateurs de l'outil les aspects les plus importants de l'application, notamment la configuration d'une structure de modèle, la personnalisation de l'application, l'utilisation de diagrammes , d'autres moyens de visualiser le contenu et les intégrations avec d'autres outils. Elle comprend également des discussions sur l'interface utilisateur, notamment les rubans et Perspectives qui sont fondamentaux pour travailler avec l'outil.

L'outil dispose d'un système d'aide complet qui a été développé sur plus de vingt ans et qui peut fournir des réponses à presque toutes les questions qu'un ingénieur ou un autre intervenant pourrait avoir lorsqu'il travaille avec l'outil.

### Où nous allons

Cette rubrique décrit la liste des rubriques du Guide, donnant un aperçu de chaque rubrique.

### Découvrir les Diagrammes SysML

Cette rubrique présente le diagramme comme la méthode préférée et la plus couramment utilisée pour visualiser les modèles et le contenu des référentiels. Elle donne une brève introduction à chaque diagramme et permet au lecteur de construire une carte mentale de ce que sont les diagrammes et de la manière dont ils peuvent être utilisés à la fois pour exprimer ses idées et pour interpréter celles des autres.

### Présentation Langage Modélisation des Systèmes

Cette rubrique donne un aperçu de l'anatomie et de la physiologie du Langage Modélisation des Systèmes . Elle fournit un résumé utile de la notation et décrit l'architecture du langage. Elle introduit également les concepts fondamentaux, notamment : les éléments, Diagrammes , les modèles et Vues . Bien que ces informations puissent être glanées dans la Spécification elle-même, elles sont résumées dans cette rubrique de manière suffisamment détaillée pour éviter au nouveau venu de devoir parcourir la spécification.

Enterprise Architect fournit des exemples de modèles et un système d'aide complet qui serviront de ressources utiles pour aider les nouveaux arrivants et les praticiens



expérimentés à acquérir de l'expérience avec le langage.

### **Collaborer en tant qu'équipe d'ingénierie**

Cette rubrique présente les formidables outils de collaboration grâce auxquels les ingénieurs système, les gestionnaires, les clients, les ingénieurs-conseils, les régulateurs et les organismes de normalisation peuvent tous contribuer aux modèles. Cela comprend les discussions, les chats, le courrier électronique des modèles, une bibliothèque d'équipe, révisions et bien plus encore.

### **Utiliser Paquetages pour Structurer le Référentiel**

Cette rubrique couvre l'aspect fondamental de la structure du référentiel et la manière dont elle est utilisée comme principe d'organisation pour garantir que le référentiel est adapté à son objectif. Elle couvre également la manière dont le Paquetage peut être utilisé comme un conteneur qui permet à la fois d'ajouter du contenu mais fournit également un mécanisme de gestion des éléments, propriétés, diagrammes et vues ajoutés à chaque Paquetage .

### **Définition et gestion Exigences**

Cette rubrique présente aux ingénieurs et aux autres parties prenantes cette discipline essentielle et centrale qui constitue la base de tous les autres aspects d'un projet modélisation . Enterprise Architect dispose d'un ensemble d'outils formidables et inégalés pour développer, gérer, visualiser et documenter les exigences. Ceux-ci sont présentés et des exemples pratiques sont donnés sur la façon dont l'outil peut être utilisé.

### **Décrire les objectifs de l'utilisateur avec des cas d'utilisation**

Cette rubrique décrit une méthode d'articulation des exigences centrée sur l'utilisateur, dans laquelle l'objectif que l'utilisateur tente d'atteindre constitue la base sur laquelle une exigence est rédigée. Les objectifs des utilisateurs sont codifiés dans des cas d'utilisation, qui sont représentés simplement sur un diagramme mais dont les détails sont développés dans Enterprise Architect à l'aide du générateur de scénarios. Cette facilité permet d'écrire la description, les contraintes telles que les conditions préalables et postérieures, et les étapes des scénarios dans une interface utilisateur convaincante et productive. Elle décrit également comment diagrammes comportementaux peuvent être générés automatiquement à partir de l'outil, et les éléments tels que les activités peuvent être liés à des éléments modélisation en amont tels que Exigences et à des éléments en aval tels que les composants, créant ainsi une traçabilité efficace et utile.

### **Utilisation de blocs pour Modèle la structure et les contraintes**

Le Bloc , qui est introduit dans cette rubrique, est l'unité fondamentale de structure dans le langage et peut également être utilisé pour modéliser des contraintes. C'est l'atome du langage SysML et peut avoir Fonctionnalités , Propriétés et des Points d'interaction qui décrivent en détail l'anatomie. Cette rubrique décrit également les relations que les Blocs entretiennent avec d'autres éléments du modèle, y compris la relation *d'Allocation* très importante qui relie les Blocs aux Activités. Les Allocations relient les deux piliers importants du SysML : *la Structure* et *le Behavior* . Elle introduit également des dispositifs de modélisation des Types *de Quantité* et *de Valeur* qui peuvent être utilisés pour modéliser les dimensions dans le monde physique.

### **Utilisation Propriétés et des pièces pour l'utilisation Modèle Bloc**

Cette rubrique fait suite à la rubrique précédente et présente le diagramme Bloc internes, qui permet de visualiser la manière dont les blocs sont utilisés dans un contexte donné. Ces diagrammes montrent comment les propriétés de pièces d'un Bloc peuvent être connectées entre elles. Le Bloc propriétaire est représenté sous la forme d'un cadre diagramme et les pièces qui apparaissent à l'extrémité de la pièce de l'association de pièces sur un diagramme de définition Bloc apparaissent sur le diagramme Bloc internes sous la forme d'un élément de pièce

### **Coordonner le comportement avec les activités**

Cette rubrique présente les activités et l'unité plus atomique, l' Action , qui sont toutes deux utilisées pour décrire les aspects comportementaux d'un système à différents niveaux. Ces éléments essentiels sont équivalents aux verbes de nos langues naturelles et, comme les verbes, ont une fonction d'organisation dans le

modèle. Enterprise Architect dispose d'un certain nombre d'outils utiles tels que les simulations qui peuvent donner vie à ces modèles et permettre de simplifier et de visualiser des problèmes complexes du monde réel.

### **Visualisation avec des paramètres et des simulations**

Cette rubrique explore l'utilisation des diagrammes Paramétriques en relation avec diagrammes de définition Bloc , qui définissent des blocs de contraintes modélisant des équations mathématiques et les paramètres qu'ils utilisent. La rubrique décrit comment ces contraintes et paramètres peuvent être représentés sur le diagramme Paramétriques , qui est un cousin du diagramme Bloc internes. Simulation des diagrammes Paramétriques est également présentée et vous apprendrez à installer et à utiliser l'interface OpenModelica. Vous apprendrez comment visualiser des tracés d'équations avancés sans avoir à quitter l'environnement Enterprise Architect .

### **Modélisation du changement avec Statemachines**

Cette rubrique présente la Statemachine comme une méthode de description des conditions discrètes ( States ) qu'une entité telle qu'un Bloc peut présenter. Ce dispositif comportemental peut apporter une grande clarté à un modèle et résoudre des problèmes autrement insolubles. Enterprise Architect dispose d'un certain nombre de fonctionnalités d'outils qui peuvent donner vie à ces modèles, à savoir le Tableau State et Statemachines Exécutables qui permettent aux States et aux Transitions d'être actifs dans des visualisations convaincantes.

### **Interactions comme Séquence de messages**

Dans cette rubrique, le diagramme Séquence est présenté comme un moyen de modélisation les comportements basés sur les messages. Vous apprendrez à utiliser ce diagramme pour modéliser une gamme de concepts d'ingénierie. Les diagrammes modélisent l'interaction entre les blocs internes au système, ou entre le système lui-même et son environnement, et ils peuvent être utilisés pour modéliser la communication qui se produit avec les étapes d'un cas d'utilisation. Les lignes de vie et les activations sont étudiées et l'orchestration des messages est introduite avec l'utilisation de fragments.

### **Un premier exemple Modèle SysML**

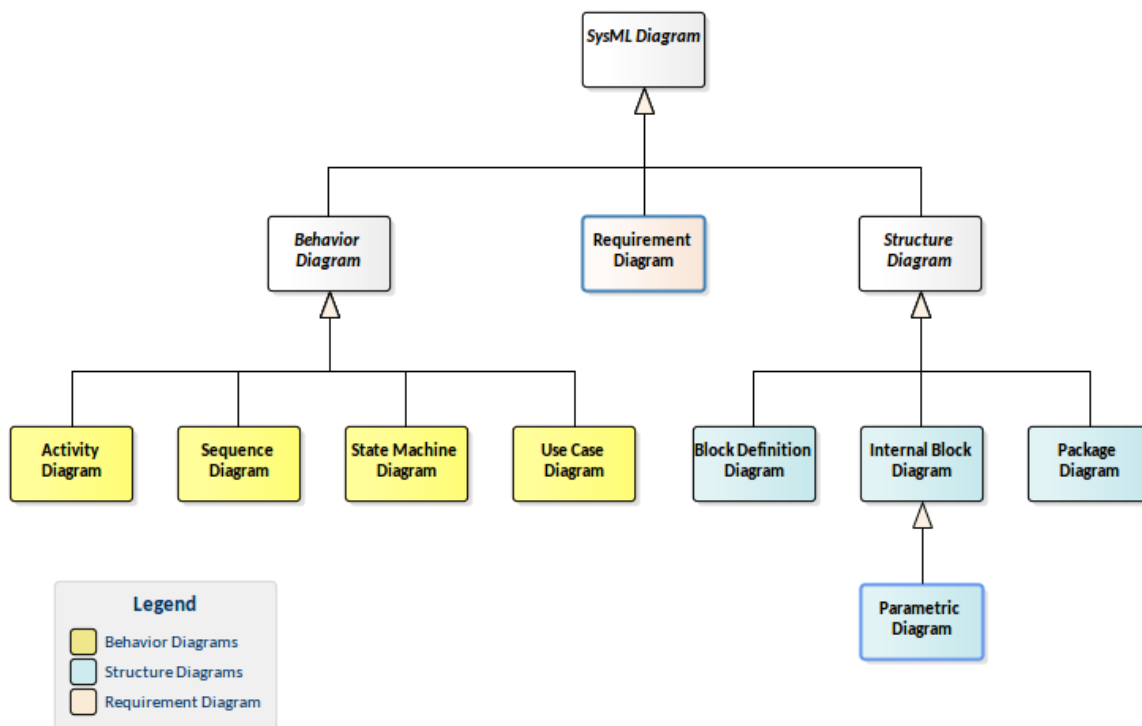
Dans cette rubrique, nous explorons les diagrammes qui spécifient, conçoivent et testent un véhicule SUV hybride qui utilise le carburant et l'électricité comme sources d'énergie, avec des mécanismes tels que le freinage régénératif pour répondre à ses exigences d'efficacité énergétique. Le modèle d'exemple est publié dans une annexe de la spécification SysML ; dans cette rubrique, nous explorons les diagrammes qui ont été créés dans Enterprise Architect , expliquant fonctionnalités importantes du langage et des outils.

### **Rencontrez les outils d'Ingénierie des Systèmes**

Cette rubrique présente certains des outils les plus importants pour travailler avec les modèles Ingénierie des Systèmes , en décrivant ce qu'ils sont, où ils peuvent être trouvés et comment ils peuvent être utilisés, y compris les options et où trouver plus d'aide. Il existe de nombreux autres outils que les modélisateurs peuvent trouver utiles ; ceux-ci sont présentés aux points pertinents du document.

## Découvrir les Diagrammes SysML

Les diagrammes SysML peuvent être considérés comme des types de canevas, où un ingénieur créera des représentations visuelles des concepts d'ingénierie qui font partie du modèle. Il existe neuf types de diagramme SysML, chacun axé sur un aspect particulier du problème ou de la solution. Bien que les types de diagramme contiennent généralement différents types d'éléments, ils sont tous conformes à une représentation standard composée de : un cadre qui contient, un en-tête et une zone de contenu.



Cette section décrit certains des outils et fonctionnalités les plus utiles qui peuvent être utilisés lors du travail avec des diagrammes, mais il en existe de nombreux autres qui peuvent être utiles. Pour plus d'informations, consultez la [Model Diagrams](#) Rubrique d'aide.

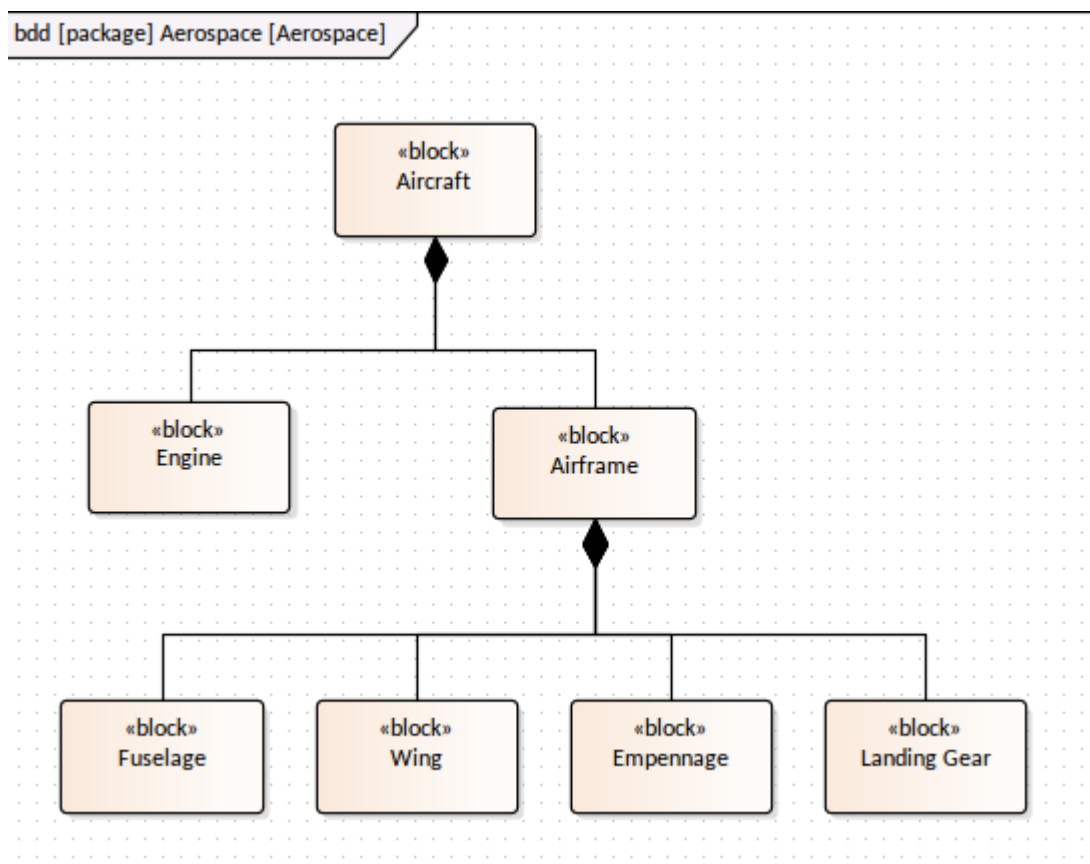
### Cadres de Diagramme

Un cadre Diagramme est un dispositif visuel qui enferme les éléments et les relations sur un diagramme. Le cadre est composé de deux parties :

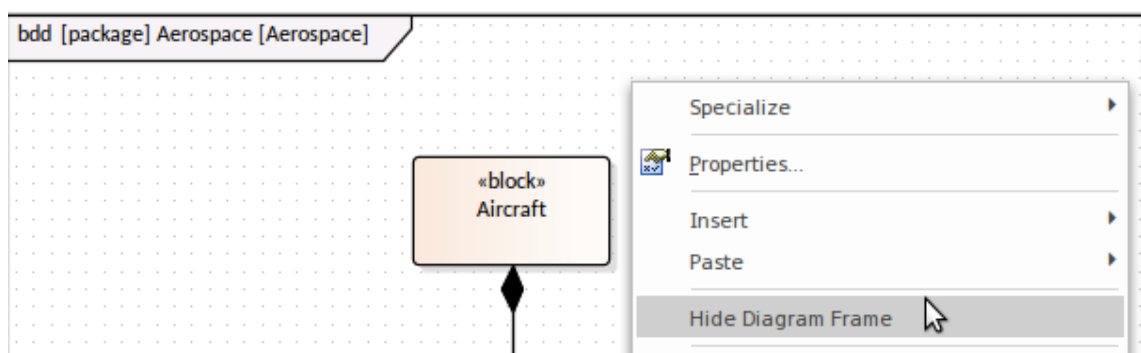
- Une *zone d'en-tête* qui contient un nom qualifié pour l'élément de modèle dans le cadre, qui est fourni s'il n'est pas contenu dans l'espace de noms par défaut associé au cadre ; il a la forme :

diagrammeKind [modelElementType] modelElementName [diagramName]

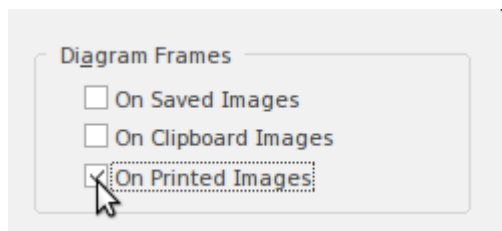
- Une *zone de contenu* qui contient les éléments visuels qui composent le diagramme



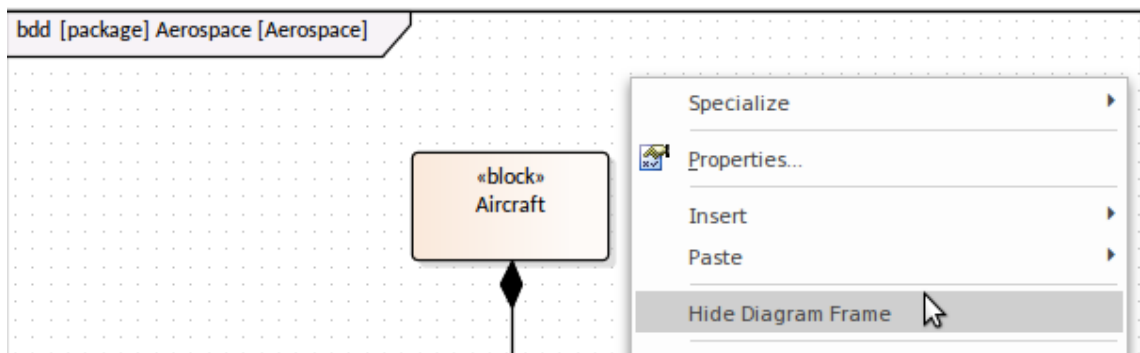
Par défaut, Enterprise Architect affiche les cadres dans une fenêtre diagramme, mais conformément à la spécification SysML, le cadre peut être supprimé pour créer une interface de création de diagrammes moins encombrée. (En gardant à l'esprit qu'Enterprise Architect affiche de manière pratique les informations d'en-tête de cadre dans l'en-tête diagramme).



Un cadre peut être réactivé à tout moment et diagrammes envoyés vers le presse-papiers ou l'imprimante peuvent être configurés pour afficher les cadres, qu'ils aient été masqués ou non dans l'interface utilisateur.



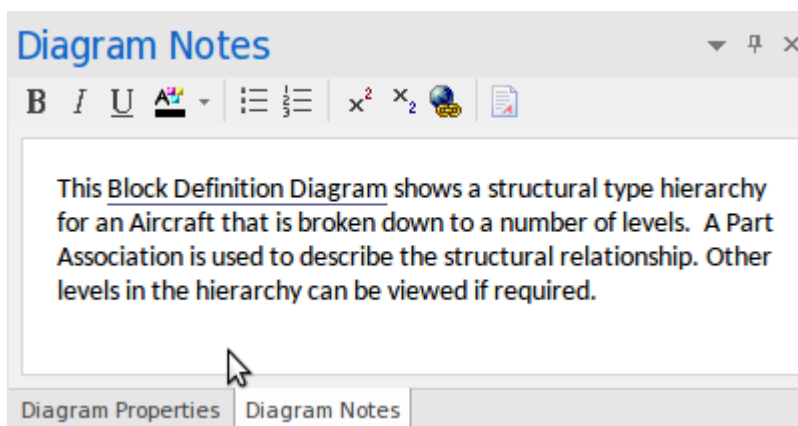
Si cette option est définie sur non sélectionnable, le cadre se redimensionnera automatiquement pour s'adapter aux limites du diagramme, en s'élargissant par rapport à sa taille par défaut mais sans rétrécir. Ils peuvent cependant être rendus sélectionnables et ajustés pour s'adapter aux préférences du modélisateur.



Note que diagrammes affichant des cadres Diagramme appliqués à l'aide de la version 14.0 ou ultérieure d' Enterprise Architect dessineront l' object parent sur le diagramme lorsqu'ils seront ouverts à l'aide d'une version d' Enterprise Architect antérieure à la version 14.0.

## Descriptions Diagramme ( Notes )

En plus des méta-informations contenues dans l'en-tête, un diagramme peut avoir une description utile aux nouveaux venus pour comprendre le but et l'intention du diagramme . La description peut être ajoutée, visualisée et conservée dans la fenêtre notes du diagramme .



Un diagramme est souvent créé pour décrire les aspects d'un modèle ou d'un système. Bien que le diagramme lui-même et les éléments et connecteurs qu'il contient racontent une histoire, il est souvent nécessaire d'annoter le diagramme avec des informations supplémentaires sous forme de texte descriptif. Ce texte peut par exemple :

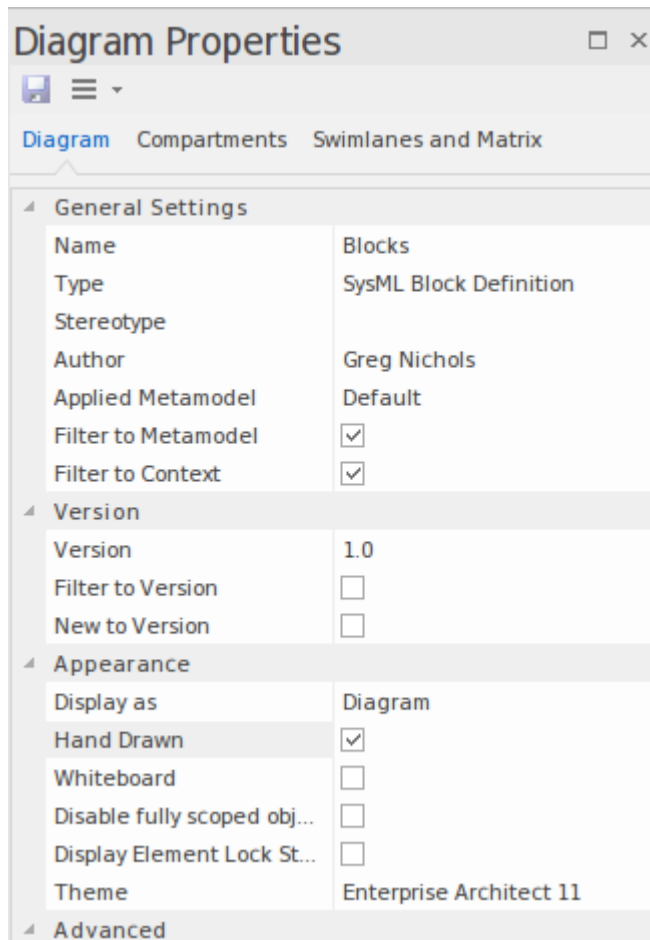
- Décrivez le but du diagramme ,
- Mettre en évidence comment interpréter le diagramme ,
- Contient un lien vers d'autres informations dans le modèle,
- Donnez une explication des symboles utilisés.

Les notes seront générées dans la documentation et sont visibles via l'interface WebEA .

## Propriétés du Diagramme

Chaque diagramme possède une série de propriétés qui décrivent le diagramme à un niveau méta, y compris des éléments tels que :

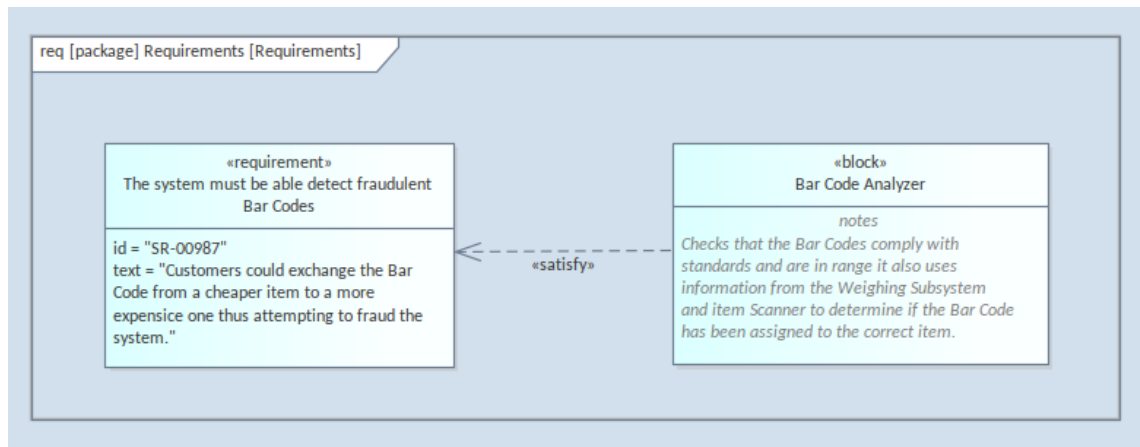
- Date de création du diagramme ,
- Date à laquelle le diagramme a été modifié,
- Auteur Modèle qui a créé le diagramme ,
- Mode dessin à la main et tableau blanc
- Thème visuel et bien d'autres.



De plus, il existe un onglet dédié à la spécification des compartiments visibles dans le diagramme et un autre onglet utilisé pour spécifier les matrices et les couloirs.

## Modification des thèmes et de l'apparence

Enterprise Architect propose une facilité qui vous permet d'appliquer un thème sélectionné à tous diagrammes présentés sur votre appareil. Vous pouvez l'utiliser pour créer un style particulier du diagramme et il peut affecter la couleur, la police, le dégradé, l'épaisseur de la ligne et l'image d'arrière-plan (tuile). C'est un mécanisme utile pour donner plus d'attrait à votre diagramme ou pour unifier son apparence.

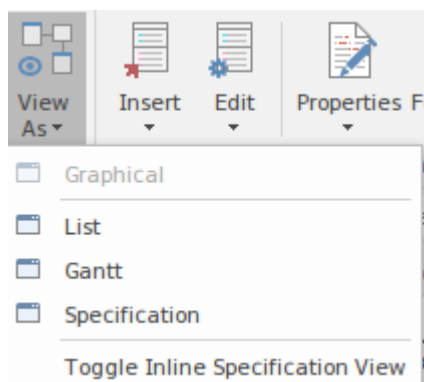


Les couleurs et les styles peuvent également être définis pour chaque élément diagramme individuellement, soit par défaut (chaque diagramme dans lequel l'élément apparaît), soit uniquement pour l'élément du diagramme actuel.

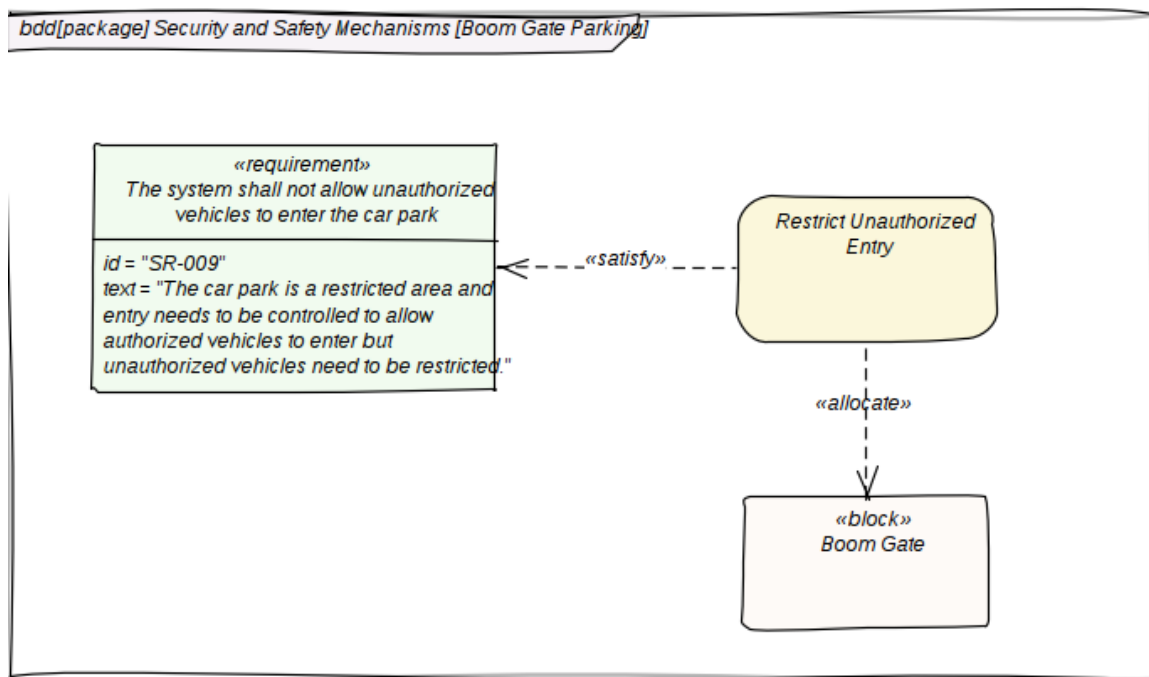
## Visualisations alternatives des Diagrammes

Tout en étant conforme à la spécification SysML et en fournissant toutes les fonctionnalités diagramme attendues dans un outil modélisation, Enterprise Architect offre aux modélisateurs un certain nombre de façons de visualiser le diagramme différemment, en apportant des visualisations enrichies du diagramme et de ses éléments. Il s'agit notamment de :

- *Spécification Vue* - présente les éléments dans un format familier de traitement de texte ou de feuille de calcul, permettant de mettre à jour les éléments et le texte
- *Liste Vue* - présente les éléments d'une liste qui peuvent être triés et regroupés, et les champs mis à jour
- *Inline Spécification Vue* - présente le diagramme avec une vue narrative similaire à Spécification Vue
- *Gantt Vue* - présente les éléments d'une vue Gantt montrant l'allocation des ressources et d'autres informations temporelles



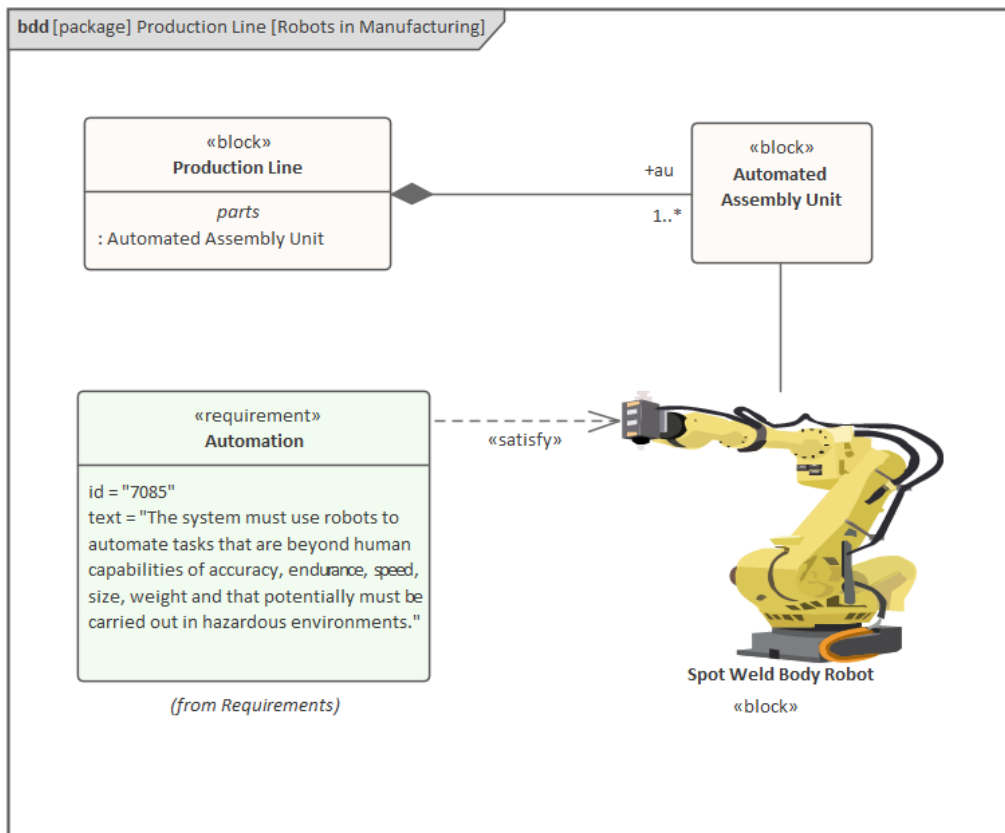
De plus, le diagramme peut être visualisé selon deux modes qui modifient l'apparence des diagrammes adaptés aux présentations, ajoutant un grand attrait et atténuant l'accent mis sur la rigueur du langage modélisation sous-jacent.



## Créer de l'attrait avec des images alternatives

Les diagrammes créés dans le cadre d'un modèle sont destinés à communiquer des idées à un public et l'intention du diagramme est souvent mieux transmise à un public non technique ou commercial par l'utilisation d'une image. Enterprise Architect fournit un mécanisme permettant de remplacer la notation graphique basique (et parfois) peu attrayante du Langage Modélisation des Systèmes par une image dans divers formats, y compris des images vectorielles.





L'image peut être appliquée à chaque instance de l'image dans diagramme ou uniquement à un diagramme particulier. Un ensemble d'images par défaut peut être importé dans le gestionnaire d'images ou un utilisateur ou une équipe est libre de créer ses propres images spécifiques à un domaine ou à un secteur particulier.

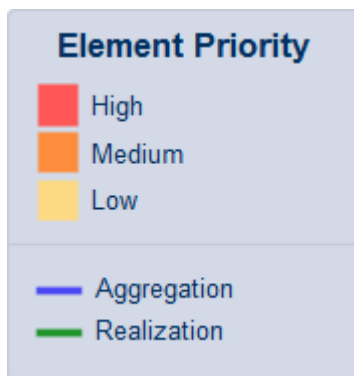
### Filtres de Diagramme

Filtres de Diagramme fournissent un mécanisme permettant de filtrer les parties d'un diagramme ou d'une liste d'éléments qui ne présentent pas d'intérêt, en ne laissant apparaître que les éléments et les connecteurs pertinents pour la vue. Les filtres peuvent être définis pour des éléments ou des connecteurs et il existe une large gamme de critères qui peuvent être définis, comme le filtrage de tous les éléments qui n'ont pas le statut « Validé » et qui ont été créés depuis une date d'étape. Plus simplement, un type d'élément ou un stéréotype peut être filtré. Les éléments qui ne répondent pas aux critères peuvent être masqués, mis en niveaux de gris ou simplement atténués (estompés) afin qu'ils soient visibles mais pas proéminents.



## Légendes Diagramme

Les légendes Diagramme permettent de décrire les éléments et les connecteurs utilisés dans le diagramme . Les légendes peuvent modifier de manière dynamique l'aspect visuel des éléments et des connecteurs du diagramme , par exemple en modifiant la couleur de remplissage, la couleur de ligne et la largeur de ligne en fonction des propriétés des éléments ou Valeur Étiquetés .



N'importe quel nombre de légendes peut être créé et elles peuvent être appliquées à un ou plusieurs diagrammes .

## Aspects communs des Diagrammes

Diagrammes sont l'un des moyens les plus importants pour visualiser le contenu d'un modèle et représentent l'expression de l'auteur d'un diagramme de ce qu'il considère comme important. Son attrait visuel convaincant et sa capacité à agir comme un récit racontant une histoire sur un aspect du système modélisé font du diagramme l'une des vues les plus importantes. Chaque diagramme dans le SysML a des aspects communs (ou fonctionnalités ), notamment :

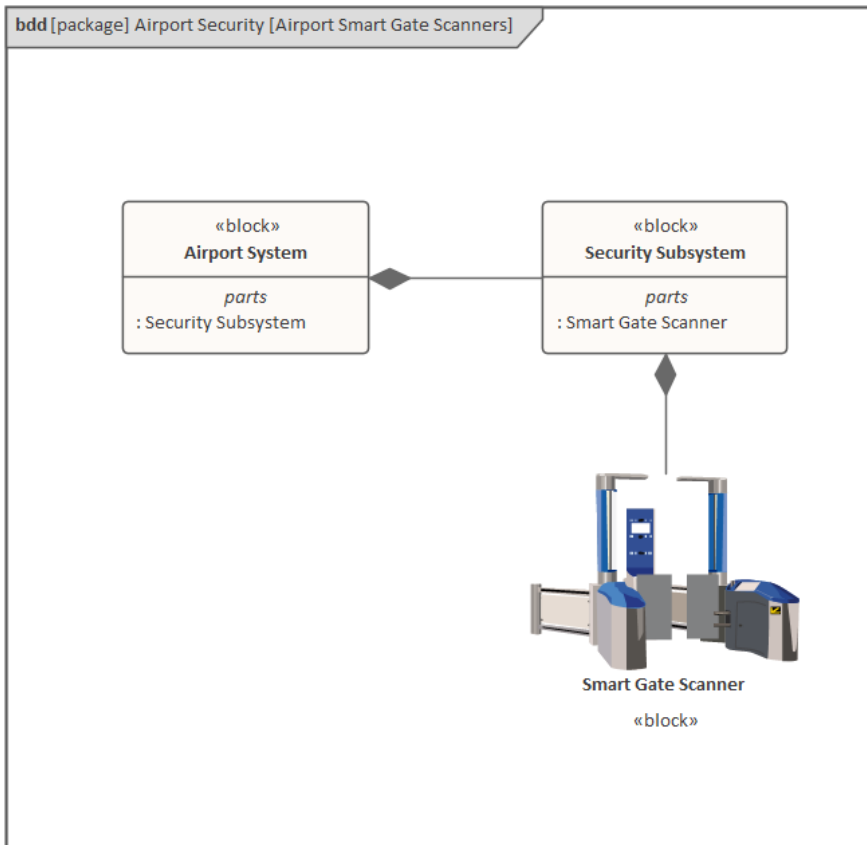
- un diagramme Cadre
- un diagramme En-tête
- un diagramme Zone de contenu (Toile)
- un diagramme Description ( Notes )
- une fiche Propriétés d'un diagramme
- Légende d'un diagramme
- Et plus

Il existe également une large gamme d'autres facilités qui aideront l'ingénieur lorsqu'il travaille avec diagrammes , notamment :

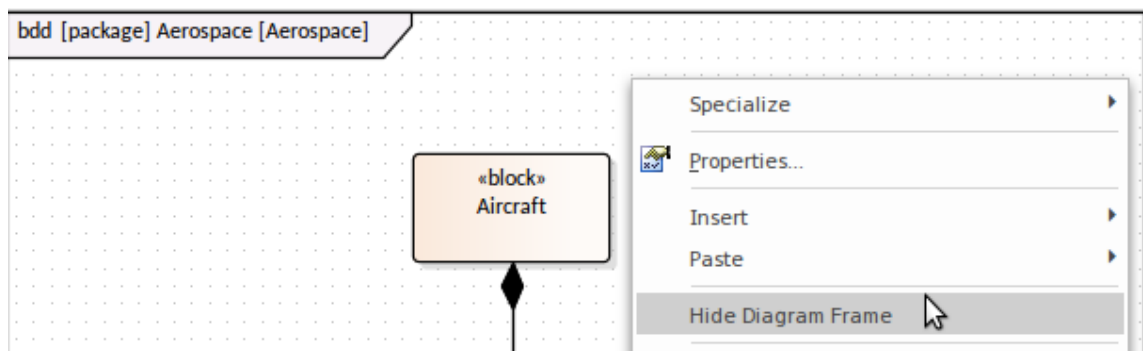
- Fenêtre de traçabilité
- Fenêtre Relations
- Pan et zoom
- Disposition Diagramme
- Filtres de Diagramme
- Feuilles de Route
- Kanban
- Options de zoom
- Outils d'apparence, d'alignement et de style

### Cadre Diagramme

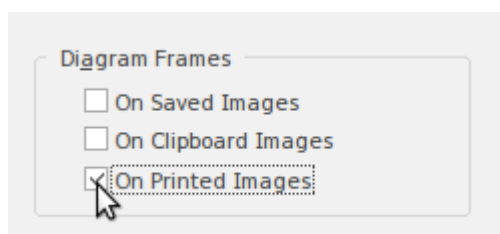
Le cadre Diagramme ressemble à un rideau entourant les éléments qui font partie du diagramme . Bien qu'il soit plus important lors de la visualisation diagrammes en ligne avec un texte écrit, certains modélisateurs préfèrent avoir le diagramme visible lors de modélisation .



Le cadre peut être facilement affiché ou masqué pour chaque diagramme , et lorsqu'il est masqué, les informations diagramme - telles que le type, le parent et le nom du diagramme - sont toujours visibles dans l'en-tête diagramme .



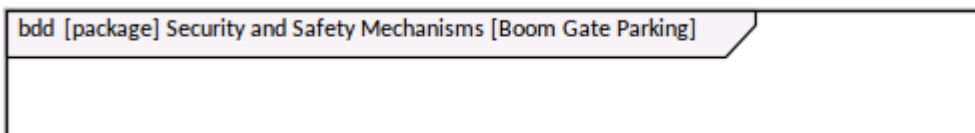
Les préférences peuvent également être définies pour afficher le cadre Diagramme lorsque diagrammes sont exportés dans le cadre de la documentation, dans le presse-papiers ou dans des images enregistrées. Ces options se trouvent sur la page « Diagramme » de la dialogue « Préférences ».



Le cadre contient un en-tête dans le coin supérieur gauche, qui contient des informations utiles sur le diagramme . Cette syntaxe décrit le contenu de l'en-tête.

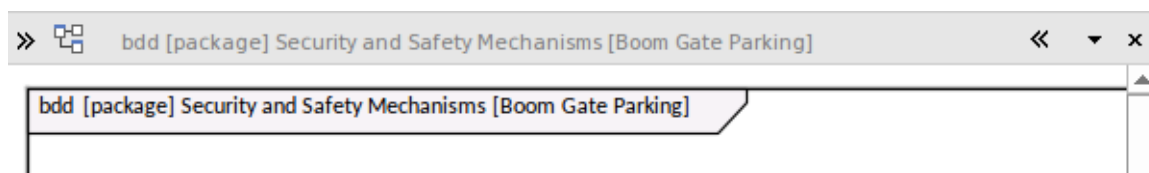
diagrammeKind [modelElementType] modelElementName [diagramName]

1. *diagramKind* - est un code pour le type de diagramme , tel que bdd ( Bloc Definition diagramme ),
2. *modelElementType* - est le type d'élément qui agit comme l'espace de noms du diagramme ,
3. *modelElementName* - est le nom de l'élément de l'espace de noms,
4. *diagramName* - est le nom du diagramme , fourni par l'utilisateur.



## Barre d'en-tête du Diagramme

La barre d'en-tête diagramme fournit des informations utiles sur le diagramme et des outils pour travailler avec tous diagrammes ouverts. Même lorsque le cadre n'est pas visible, l'en-tête affichera les détails diagrammes .

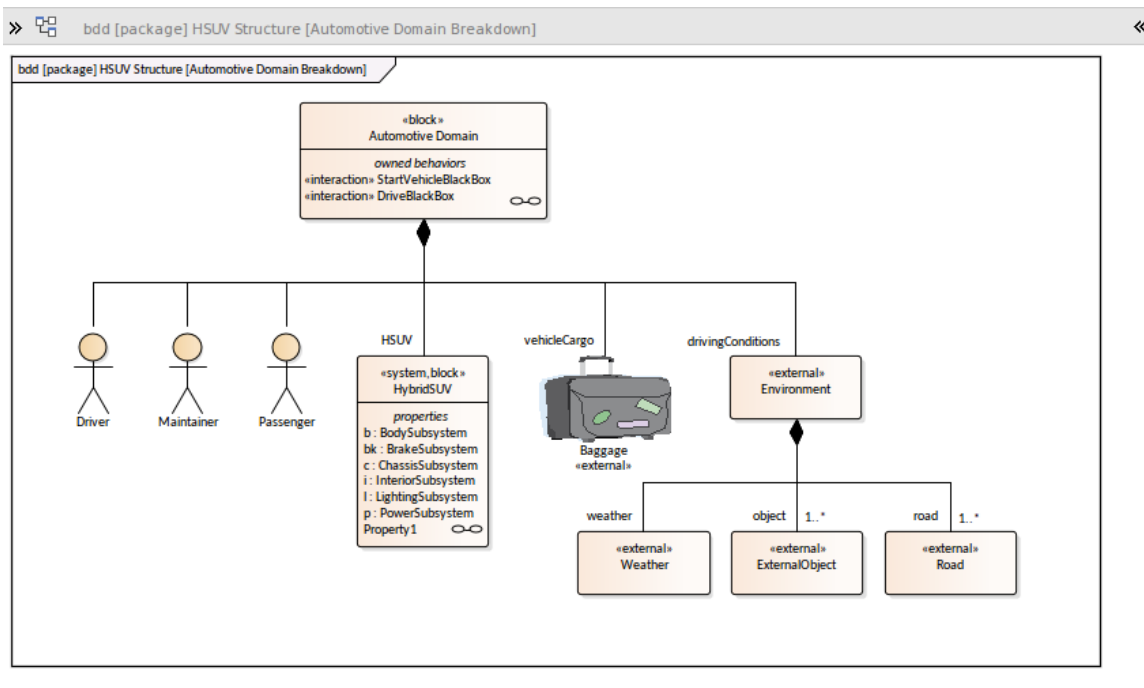


Il dispose également d'un certain nombre d'autres icônes utiles qui sont utilisées pour contrôler certains aspects de l'affichage, notamment :

- *Boîte à outils Chevron* - qui masque ou affiche la boîte à outils pour tous diagrammes affichés,
- *Document Chevron* - qui affiche ou masque la vue Spécification en ligne du diagramme ,
- *Flèche Diagrammes ouverts* - qui affiche une liste de diagrammes ouverts, indiquant celui avec des modifications non enregistrées,
- *Fermer Diagramme Icône* - permet de fermer le diagramme .

## Contenu Diagramme

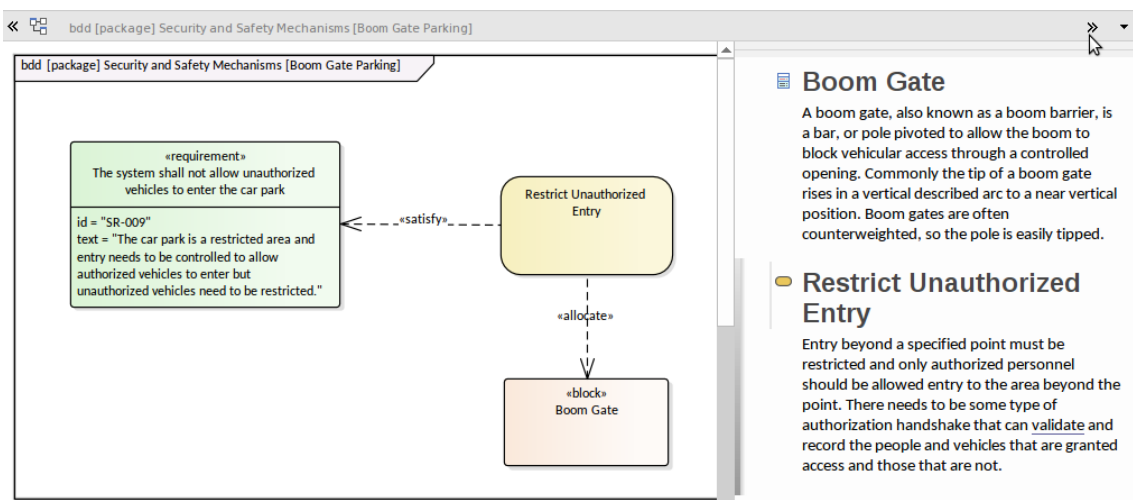
Le contenu diagramme est la toile sur laquelle vous visualisez et travaillez sur votre diagramme . Il s'agit d'une zone de travail extrêmement flexible avec de nombreux outils utiles pour créer et visualiser diagrammes existants. Des images peuvent être ajoutées et mélangées aux éléments géométriques standard SysML, ce qui permet de créer diagrammes expressifs et convaincants qui aident à transmettre des concepts d'ingénierie importants à des publics techniques et non techniques.



Enterprise Architect étend la manière d'afficher un diagramme à l'aide de plusieurs techniques de visualisation. Celles-ci vous fourniront des moyens alternatifs de travailler avec le contenu diagramme et sont bien accueillies par les nouveaux venus qui sont peut-être plus familiarisés avec le travail avec des éléments dans des feuilles de calcul, des listes et des documents. Notez également dans ce diagramme que les images peuvent être utilisées comme alternative aux formes SysML classiques.

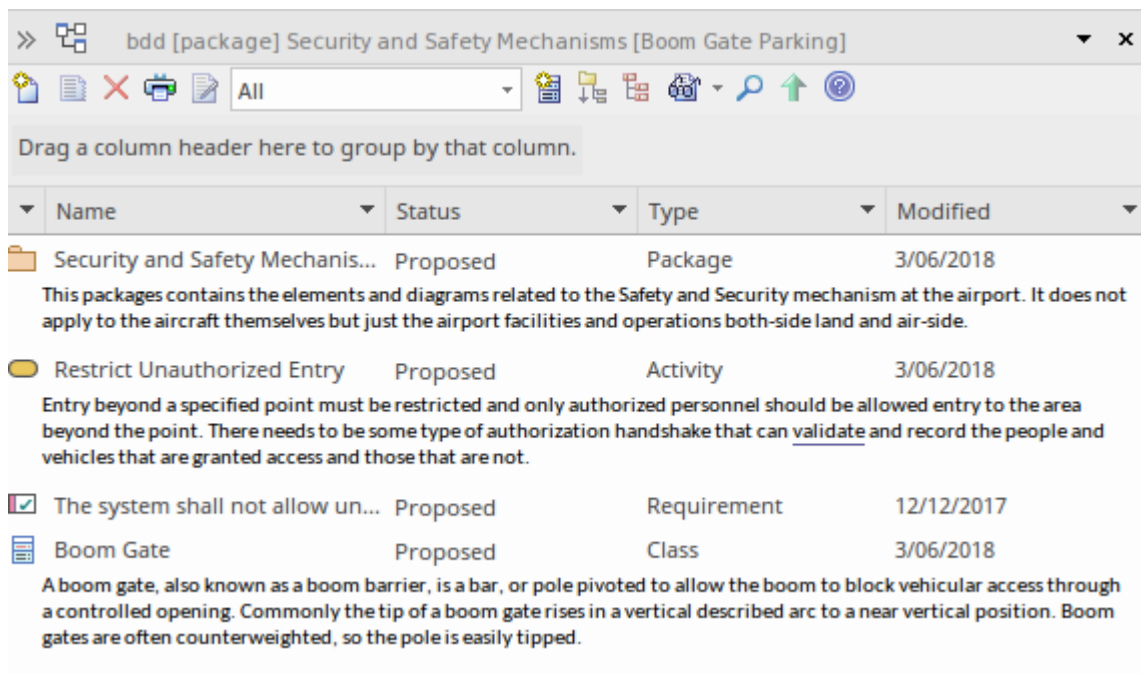
### Vue du document

Il s'agit d'une vue pratique qui affiche les éléments du diagramme sous forme narrative dans un document. Elle est également connue sous le nom de panneau « Spécification en ligne » et n'est pas une vue séparée mais un panneau lié au diagramme sélectionné. Le document est ouvert de manière pratique à droite du diagramme afin que diagramme et le document puissent être visualisés simultanément. Chaque élément présent dans le diagramme aura un titre dans le document avec les notes de l'élément (description) affichées en position sous le titre. Les éléments et leurs notes présentés dans cette vue peuvent être facilement modifiés, tout en visualisant à la fois la vue du document et le diagramme, et les deux restent synchronisés. Les éléments sont par défaut classés par ordre alphabétique, mais cet ordre de tri peut être modifié à partir du menu contextuel pour suivre de haut en bas ou de gauche à droite.



## Liste Vue

Il s'agit d'une manière pratique de visualiser les éléments d'un diagramme tout en leur permettant d'être visualisés, mis à jour et créés dans une vue familière de type feuille de calcul. Les propriétés des éléments, y compris les propriétés standard, Valeur Étiquetés et notes peuvent être gérées, et des listes déroulantes pour les propriétés avec une liste discrète de valeurs sont disponibles, offrant un moyen apprécié de visualiser ces propriétés sur plusieurs éléments.



Name	Status	Type	Modified
Security and Safety Mechanis...	Proposed	Package	3/06/2018
This packages contains the elements and diagrams related to the Safety and Security mechanism at the airport. It does not apply to the aircraft themselves but just the airport facilities and operations both-side land and air-side.			
Restrict Unauthorized Entry	Proposed	Activity	3/06/2018
Entry beyond a specified point must be restricted and only authorized personnel should be allowed entry to the area beyond the point. There needs to be some type of authorization handshake that can <u>validate</u> and record the people and vehicles that are granted access and those that are not.			
<input checked="" type="checkbox"/> The system shall not allow un...	Proposed	Requirement	12/12/2017
Boom Gate	Proposed	Class	3/06/2018
A boom gate, also known as a boom barrier, is a bar, or pole pivoted to allow the boom to block vehicular access through a controlled opening. Commonly the tip of a boom gate rises in a vertical described arc to a near vertical position. Boom gates are often counterweighted, so the pole is easily tipped.			

## Spécification Vue

Le Gestionnaire de Spécification ressemble au visualiseur de documents en ligne, mais offre plus de puissance et s'ouvre dans une fenêtre ancrable distincte. C'est l'outil parfait, conçu pour les ingénieurs et autres intervenants qui sont plus habitués à travailler avec des feuilles de calcul ou des documents. Il permet essentiellement à un modélisateur ou à un visualiseur de visualiser le contenu d'un diagramme (ou Paquetage) sous forme de document ou de feuille de calcul. La vue du document ressemble au document de traitement de texte familier, qui peut être modifié en ligne pour créer de nouveaux éléments et leurs descriptions. La visualisation peut être modifiée pour ressembler à une feuille de calcul où les propriétés sont affichées en colonnes.

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.

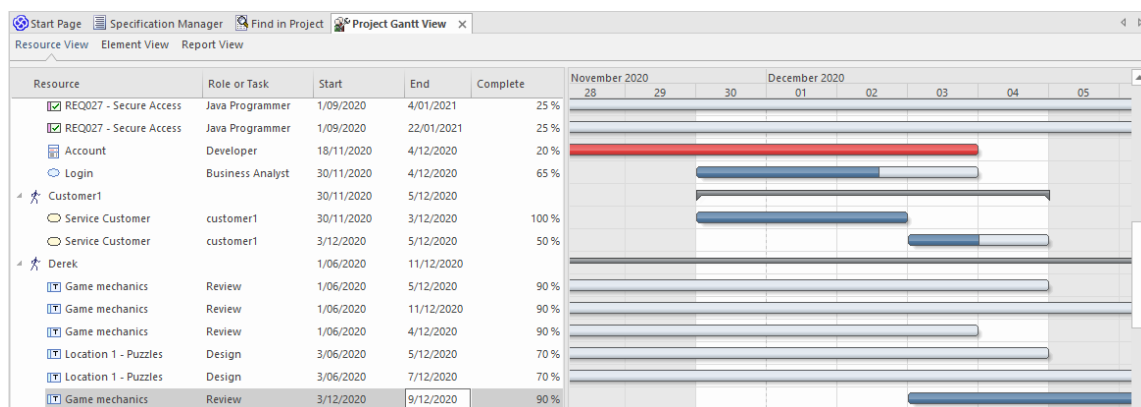
### 1.2.2 REQ021 - List Stock Levels

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

Il s'agit d'une vue très appréciée par ceux qui passent des processus d'ingénierie des systèmes basés sur des documents, et d'un favori pour les ingénieurs qui saisissent des informations basées sur des tests telles qu'un ensemble d'exigences, d'activités, de composants et d'autres éléments.

## Diagramme de Gantt

Cela fournit une vue de diagramme de Gantt typique des éléments du diagramme où l'allocation des ressources a été appliquée aux éléments. Cela peut être visualisé pour donner à un ingénieur ou à un chef de projet une vue sur la progression du modèle.





## Outils de création de diagrammes

Il existe également une large gamme d'autres outils qui peuvent être utilisés pour afficher les informations dans le référentiel et qui aideront l'ingénieur lorsqu'il travaille avec des éléments dans diagrammes , y compris, mais sans s'y limiter :

### Fenêtre de traçabilité

Permet de visualiser la manière dont les éléments du modèle sont connectés dans un graphique d'éléments et leurs relations. Pour plus d'informations, consultez la rubrique [The Traceability Window](#) .

### Fenêtre Relations

Permet d'afficher la relation entre un élément sélectionné et d'autres éléments du modèle. Ces relations ne sont pas visibles mais peuvent être facilement visualisées dans cette fenêtre séparée. Pour plus d'informations, consultez la rubrique [The Relationships Window](#) .

### Pan et zoom

Permet de se déplacer dans un grand diagramme à l'aide d'un petit rectangle qui représente la fenêtre d'affichage et de zoomer à l'aide d'un curseur. Pour plus d'informations, consultez la rubrique [Pan and Zoom](#) .

### Disposition Diagramme

Permet de créer des présentations attrayantes d'un diagramme , en utilisant motifs disposition visuelle sélectionnés tels que des digrammes et des ressorts. Pour plus d'informations, consultez la rubrique [Layout Diagrams](#) .

### Filtres de Diagramme

Permet de filtrer les éléments visibles dans un diagramme soit en les masquant, soit en les remplaçant par une échelle de gris, soit en les atténuant. Vous pouvez également inverser le comportement et sélectionner l'élément à inclure. Pour plus d'informations, consultez la rubrique [Diagram Filters](#) .

### Feuilles de Route

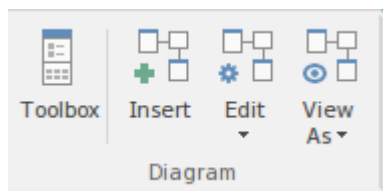
Permet de créer des représentations temporelles d'éléments, avec une échelle de temps appliquée qui peut être configurée en fonction du projet ou du modélisateur pour afficher une feuille de route à des fins stratégiques et de développement. Pour plus d'informations, consultez la rubrique [Roadmap Diagram](#) .

### Kanban

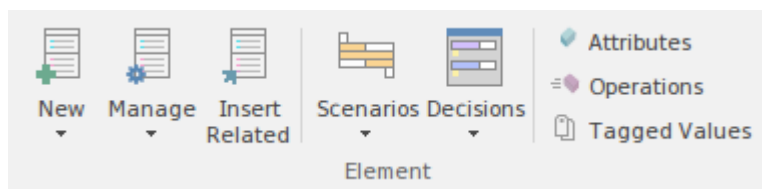
Utilisé pour gérer les éléments dans un diagramme Kanban traditionnel, où les éléments sont déplacés entre des colonnes qui représentent leur ordre dans un processus par étapes ; les ressources travaillant sur les éléments peuvent être visualisées, offrant ainsi un moyen utile à une équipe de gérer son modèle ou le développement de son produit. Pour plus d'informations, consultez la rubrique [Kanban Boards](#) .

## Diagramme Rubans et Menus

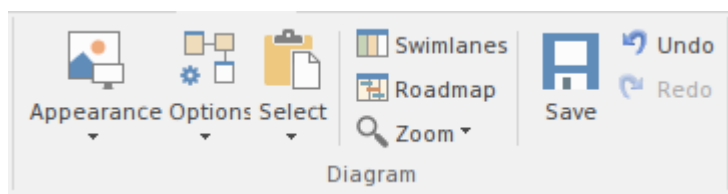
Il existe un certain nombre de rubans utiles pour travailler avec diagrammes . Le point de départ, comme décrit précédemment, est le panneau « Diagramme » du ruban Conception, qui vous permet d'insérer de nouveaux diagrammes et de modifier et de changer l'affichage des diagrammes existants.



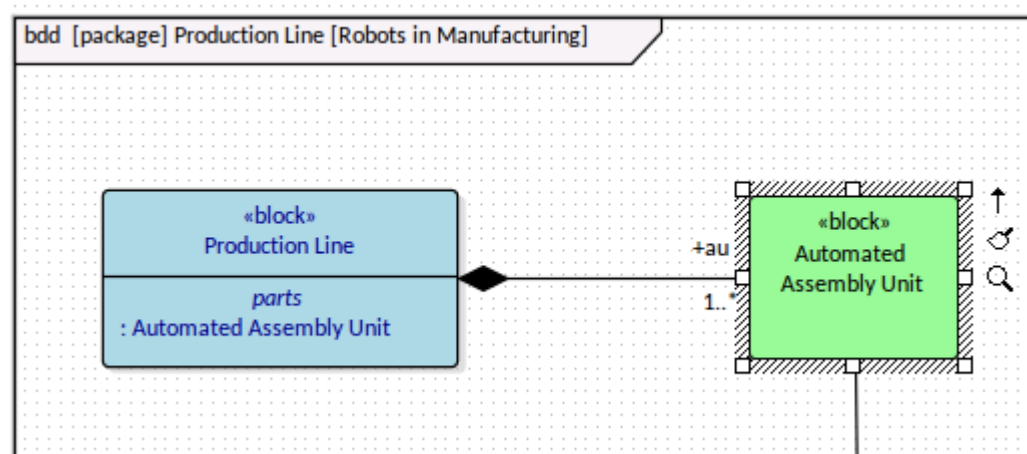
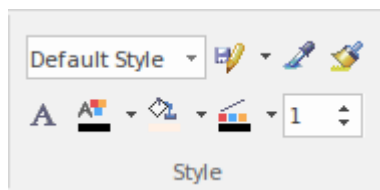
Le panneau 'Elément' du ruban Conception sera également utile lorsque vous travaillerez avec des éléments du diagramme (ou sélectionnés dans la fenêtre du Navigateur ) vous permettant d'insérer de nouveaux éléments, de modifier les éléments existants et de gérer les propriétés, fonctionnalités et responsabilités des éléments. Pour plus d'informations, consultez la rubrique d'aide [Design Ribbon](#) .



Un autre ruban important est le ruban Disposition , qui contient un certain nombre de panneaux qui seront utiles pour travailler avec diagrammes . Cela inclut le panneau « Diagramme », qui contient des options pour définir des thèmes et changer le mode diagramme , par exemple en Dessiné à la main. Pour plus d'informations, consultez la rubrique [Layout Ribbon](#) .



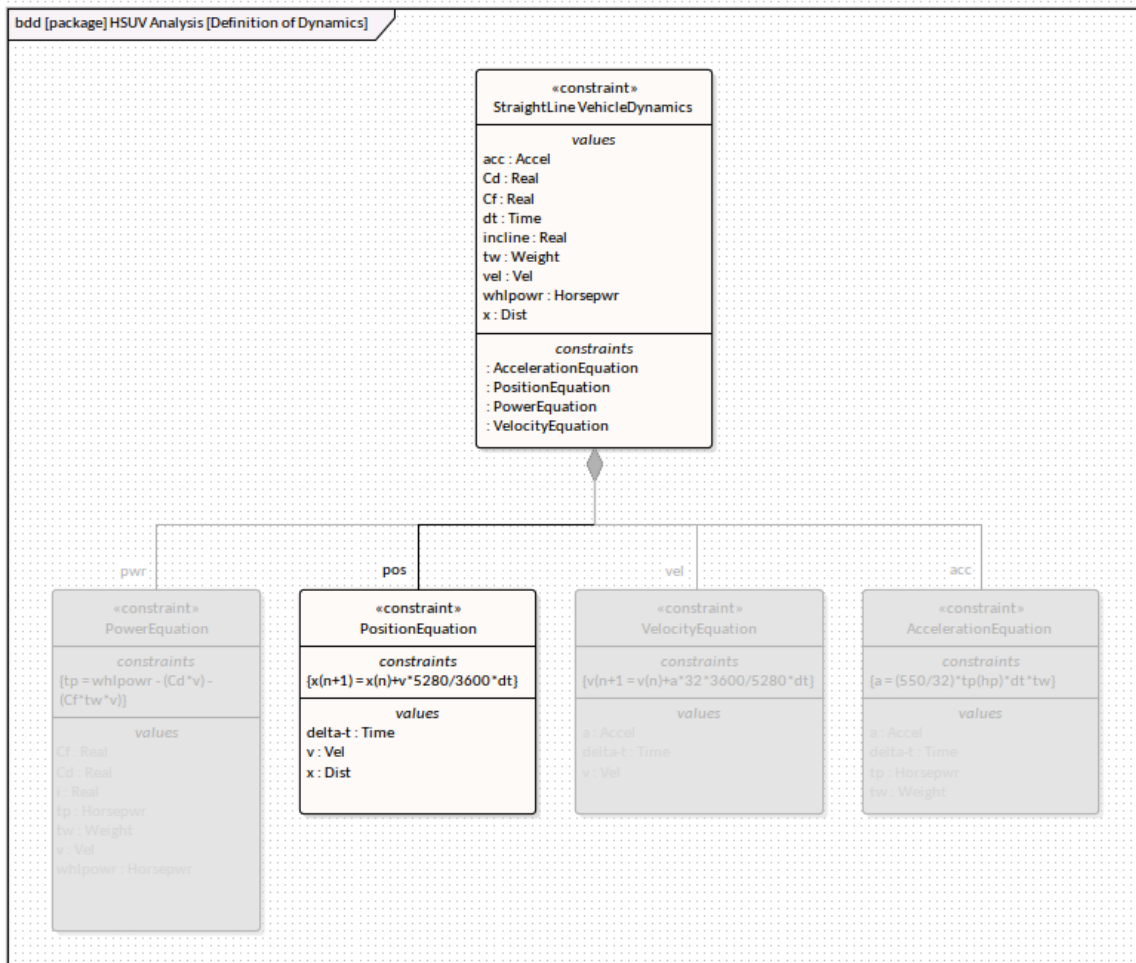
Le panneau « Style » permet de définir les propriétés de style visuel des éléments diagramme soit en tant que groupe, soit pour des éléments individuels.



Le panneau « Alignements » propose un ensemble complet d'outils permettant d'aligner les éléments d'un diagramme . Ceux-ci sont très utiles et offrent une gamme d'options d'alignement qui ne sont généralement pas disponibles dans la plupart Paquetages de dessin, permettant aux éléments, par exemple, d'être espacés uniformément horizontalement ou verticalement.

Le panneau « Outils » fournit une série d'outils pour travailler avec diagrammes , permettant de filtrer le contenu des

diagrammes (par exemple, permettre à un modélisateur d'afficher uniquement les éléments de contrainte avec un statut spécifié, qui ont été créés après une date spécifiée) ainsi que les options Pan , Zoom et Disposition . Le panneau Filtre fournit un moyen rapide et ad hoc de filtrer le contenu du diagramme sans avoir à créer un filtre diagramme .



Une série d'aides sont également disponibles pour travailler avec diagrammes et leurs éléments, contrôlant des éléments tels que le maintien horizontal et vertical pour restreindre le mouvement diagonal des éléments.

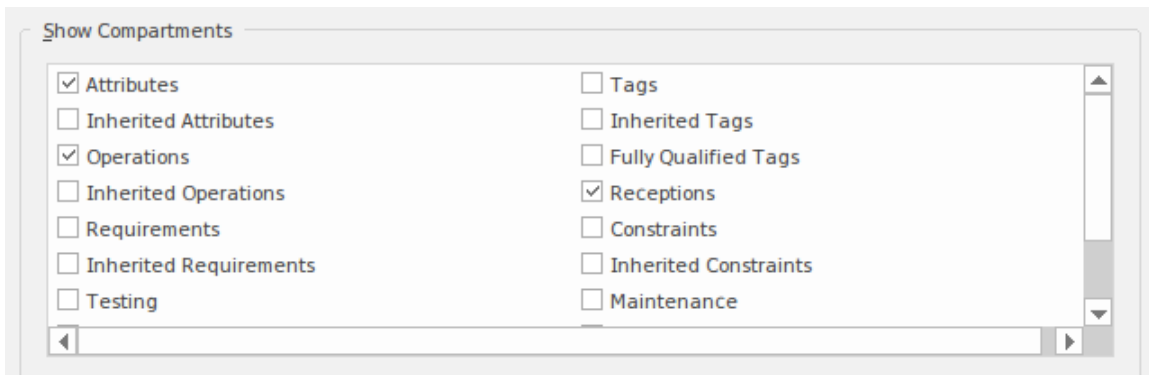
## Propriétés du Diagramme

Un diagramme dans Enterprise Architect possède un riche ensemble de propriétés, certaines étant descriptives - comme le nom - et d'autres étant prescriptives, spécifiant comment le diagramme doit être affiché et quels éléments, compartiments et autres fonctionnalités doivent être visibles, y compris les styles de ligne.

La plupart de ces propriétés peuvent être définies à la fois au niveau du diagramme et au niveau de l'élément ou du connecteur, ce qui permet d'afficher différentes parties du diagramme différemment des autres.

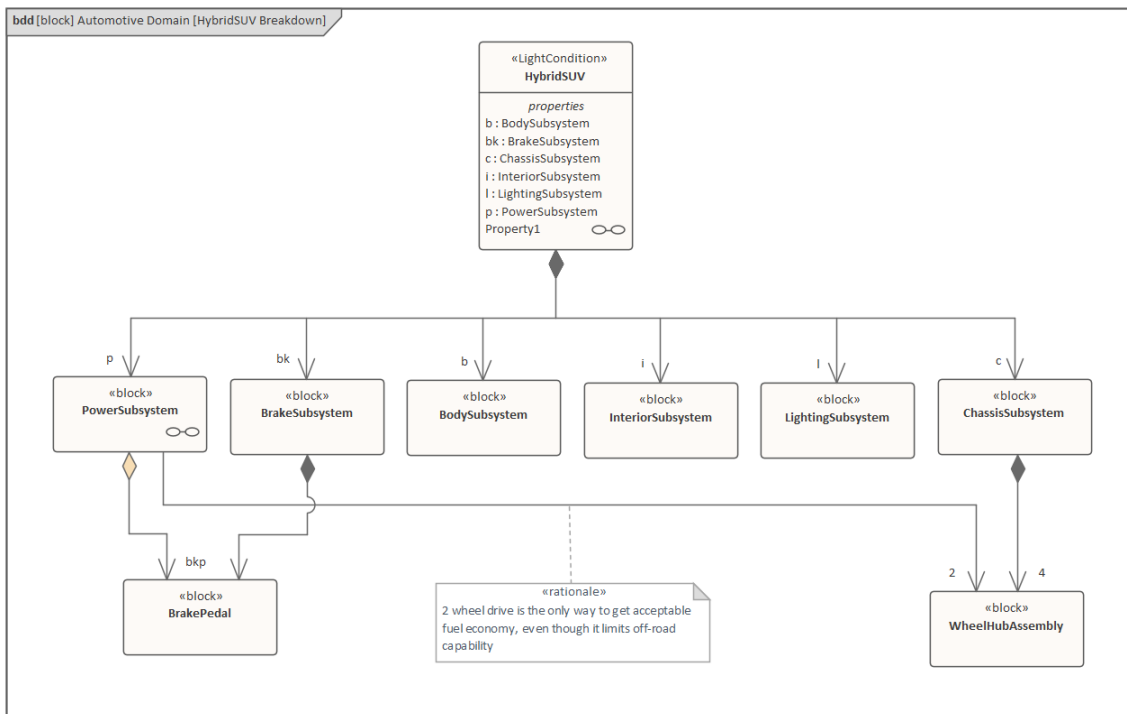
La possibilité de définir la visibilité des compartiments d'éléments est particulièrement utile pour diagrammes d'ingénierie, car SysML fournit un ensemble riche de compartiments pour une large gamme d'éléments. Ces compartiments, s'ils ne sont pas gérés, peuvent encombrer un diagramme et réduire la capacité du lecteur à comprendre la signification du diagramme .

Les compartiments, comme pour d'autres propriétés, peuvent être définis au niveau diagramme ou au niveau de l'élément, ce qui permet au modélisateur de sélectionner les compartiments à afficher pour des éléments spécifiques.



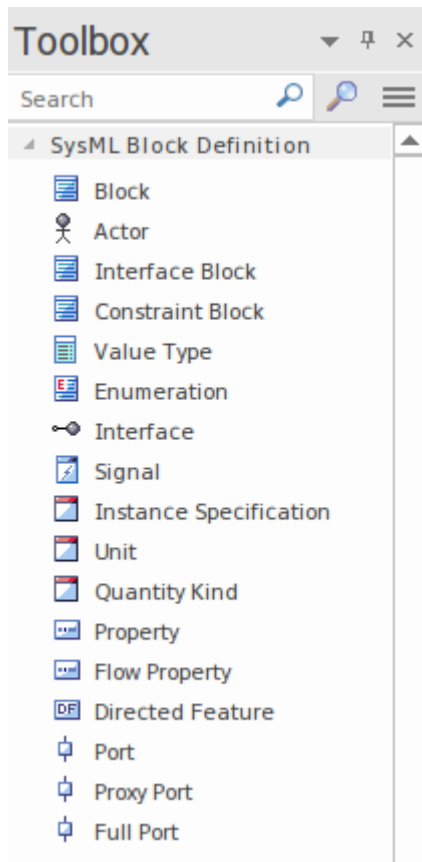
# Interrompre lorsqu'une Variable Change de Valeur

Le diagramme de définition Bloc est le plus utilisé des diagrammes SysML ; il est utilisé pour modéliser les blocs, leurs relations avec d'autres éléments (y compris d'autres blocs) et leurs fonctionnalités sous la forme de Propriétés , d'Opérations et de Réceptions. Les blocs sont des unités modulaires de description de système et fournissent un moyen de modélisation les systèmes sous forme de graphe ou d'arbre d'unités modulaires. D'autres éléments, tels que les ConstraintBlocks et Propriétés , peuvent également apparaître sur le diagramme et aider à décrire le système modélisé. Pour plus d'informations, consultez la rubrique d'aide [Block Definition Diagrams \(BDDs\)](#) .

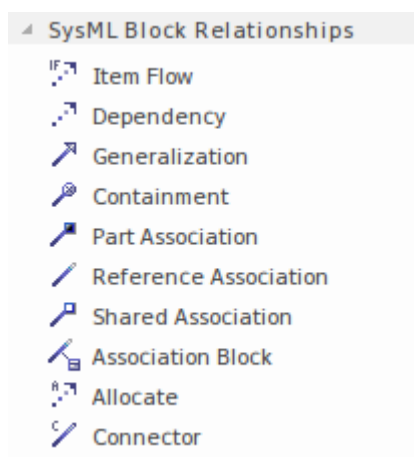


## Éléments

Les principaux éléments qui peuvent apparaître dans diagrammes Bloc Definition sont fournis par la boîte à outils Diagramme pour ce type diagramme :



Les principaux connecteurs qui peuvent apparaître dans diagrammes de définition Bloc sont les suivants :



## Outils

Une variété d'outils peuvent être utilisés avec modélisation structurelle et diagrammes de définition Bloc , notamment :

- Filtres de Diagramme - qui permettent à un utilisateur de filtrer les éléments du diagramme pour obtenir une mise au point plus spécifique
- Pan et zoom - qui permet à un modélisateur ou à un spectateur de se déplacer facilement dans de grands diagrammes
- Importation et exportation de feuilles de calcul (CSV) - qui permet d'importer ou d'exporter le contenu des feuilles de calcul à partir du modèle

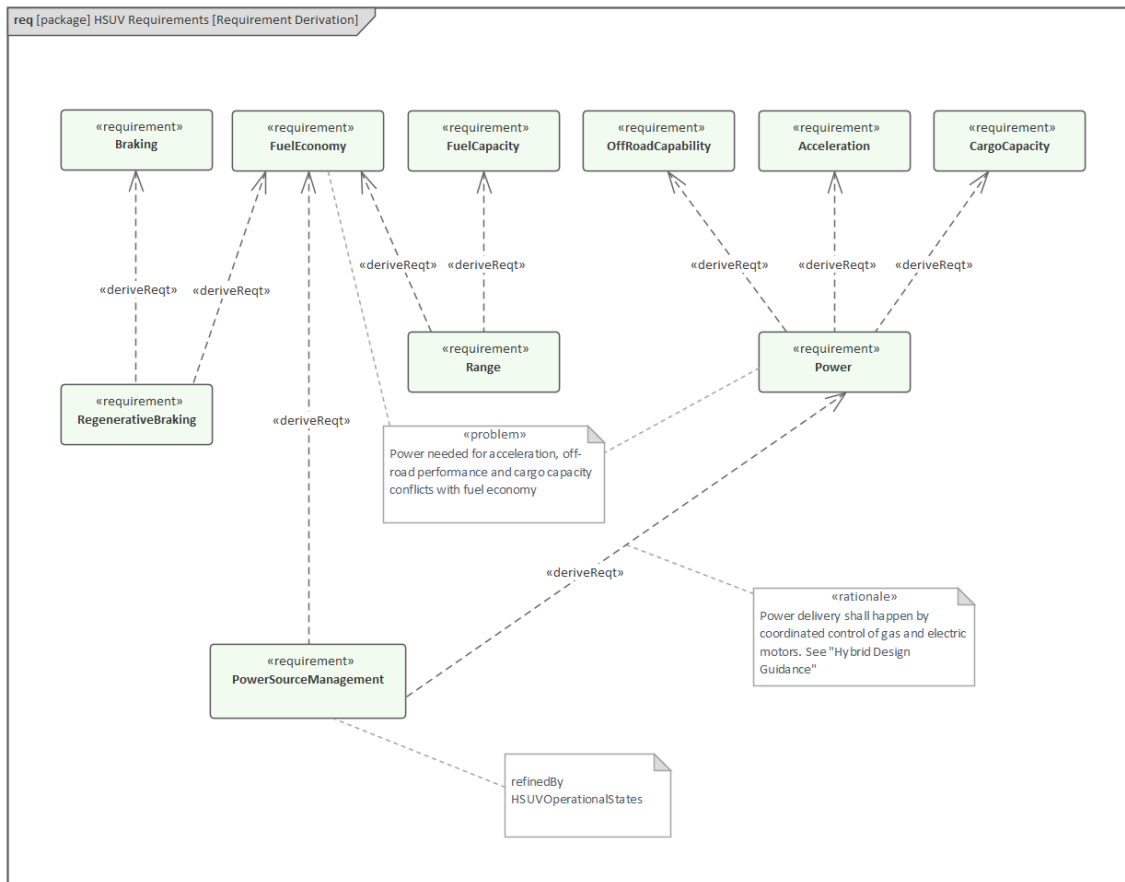
- Documentation - qui permet de générer une documentation formelle ou informelle à partir du modèle dans une variété de formats
- Traçabilité - qui fournit une vue hiérarchique des relations d'un élément avec d'autres éléments du modèle
- Fenêtre de responsabilité - qui fournit une vue composite des responsabilités importantes d'un élément, y compris les contraintes, Exigences et les scénarios
- Matrice de relations - qui permet de visualiser les connexions entre les blocs (ou d'autres éléments) et d'autres éléments tels que Exigences et les cas d'utilisation dans une matrice

## Usage

L'Interrompre lorsqu'une Variable Change de Valeur est un diagramme à usage général qui peut être utilisé pour décrire les aspects structurels d'un système.

## Diagramme des exigences

Le diagramme Exigences permet de créer et d'afficher Exigences et leurs relations avec d'autres éléments, y compris d'autres Exigences . Exigences peuvent être spécifiées à n'importe quel niveau, des exigences stratégiques de l'entreprise ou de l'entreprise aux exigences des parties prenantes, en passant par l'ingénierie de bas niveau et même les exigences logicielles et de transition. Pour plus d'informations, consultez la rubrique d'aide [SysML Requirements Modeling](#) .



Les éléments contenus dans ce diagramme peuvent être visualisés de différentes manières, notamment :

- *Spécification Vue* - permet d'afficher les éléments et leurs notes au format traitement de texte ou tableur dans une fenêtre ancrable séparée
- *Spécification en ligne Vue* - permet de visualiser côte à côte le diagramme et une liste de ses éléments sous une forme narrative
- *Liste Vue* - permet de visualiser les éléments diagramme dans une liste qui peut être triée et les éléments regroupés par propriétés
- *Gantt Vue* - permet de représenter les éléments diagramme sur un diagramme de Gantt montrant comment les ressources sont utilisées au fil du temps

## Éléments

Les principaux éléments qui peuvent apparaître dans diagrammes Exigences sont :

- Exigence
- Cas Test

Les principaux connecteurs qui peuvent apparaître dans diagrammes Exigences sont :

- Endiguement



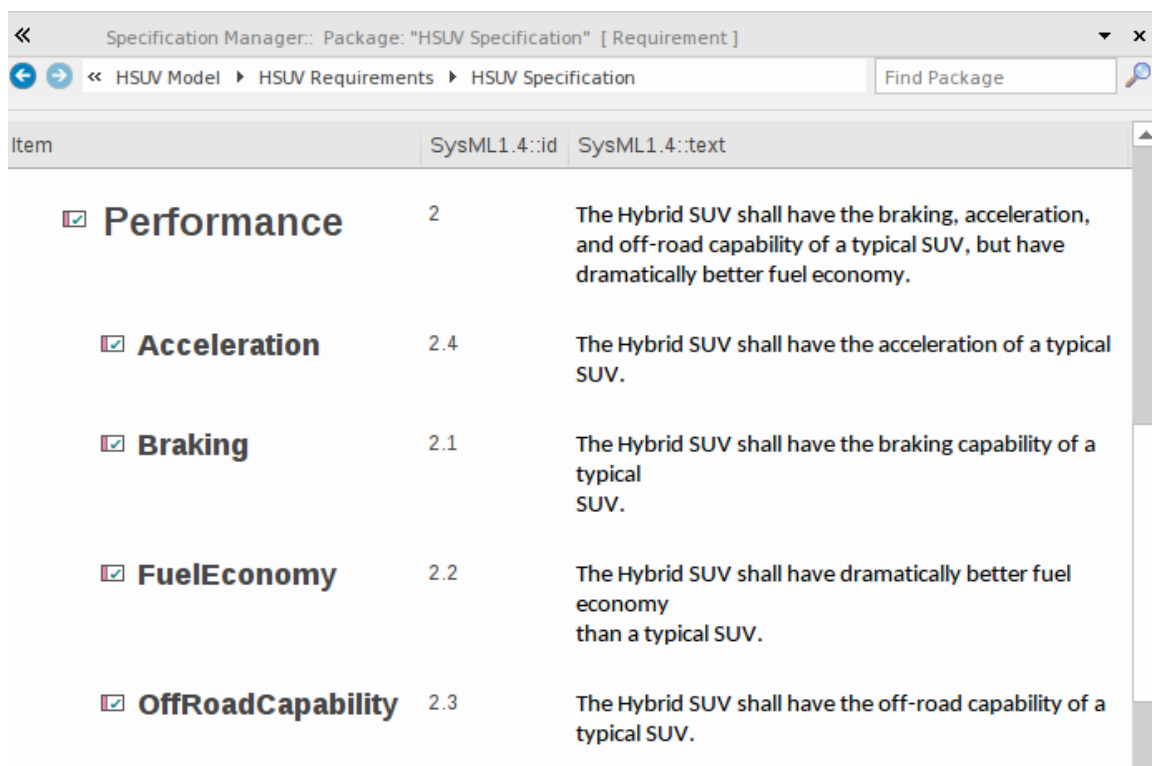
- Tracer
- Copie
- Dériver
- Vérifier
- Affiner
- Satisfaire

## Outils

Une variété d'outils peuvent être utilisés avec modélisation des exigences, notamment :

- Gestionnaire de Spécification - qui permet à un utilisateur de travailler avec des exigences dans un format de type tableur ou traitement de texte
- Importation et exportation de feuilles de calcul (CSV) - qui permet d'importer ou d'exporter le contenu des feuilles de calcul à partir du modèle
- Documentation - qui permet de générer une documentation formelle ou informelle à partir du modèle dans une variété de formats
- Traçabilité - qui fournit une vue hiérarchique des relations d'un élément avec d'autres éléments du modèle
- Fenêtre de responsabilité - qui fournit une vue composite des responsabilités importantes d'un élément, y compris les contraintes, Exigences et les scénarios
- Matrice de relations - qui permet de visualiser dans une matrice les connexions entre les exigences (ou d'autres éléments) et d'autres éléments tels que les besoins des parties prenantes
- Cartographie mentale - qui permet d'enregistrer le déroulement d'une réunion, enregistrant ainsi - par exemple - les besoins des parties prenantes

L'outil le plus utilisé de ces outils de gestion des exigences est probablement le Gestionnaire de Spécification, qui offre un moyen pratique et familier de travailler avec des spécifications textuelles telles que des exigences ou des contraintes. Le Gestionnaire de Spécification peut être utilisé pour afficher une liste d'éléments contenus dans un Paquetage ou un diagramme.



The screenshot shows a software window titled "Specification Manager: Package: 'HSUV Specification' [ Requirement ]". The breadcrumb navigation shows the path: "HSUV Model > HSUV Requirements > HSUV Specification". There is a search bar labeled "Find Package". Below the navigation is a table with three columns: "Item", "SysML1.4::id", and "SysML1.4::text". The table lists five requirements, each with a checked checkbox in the "Item" column.

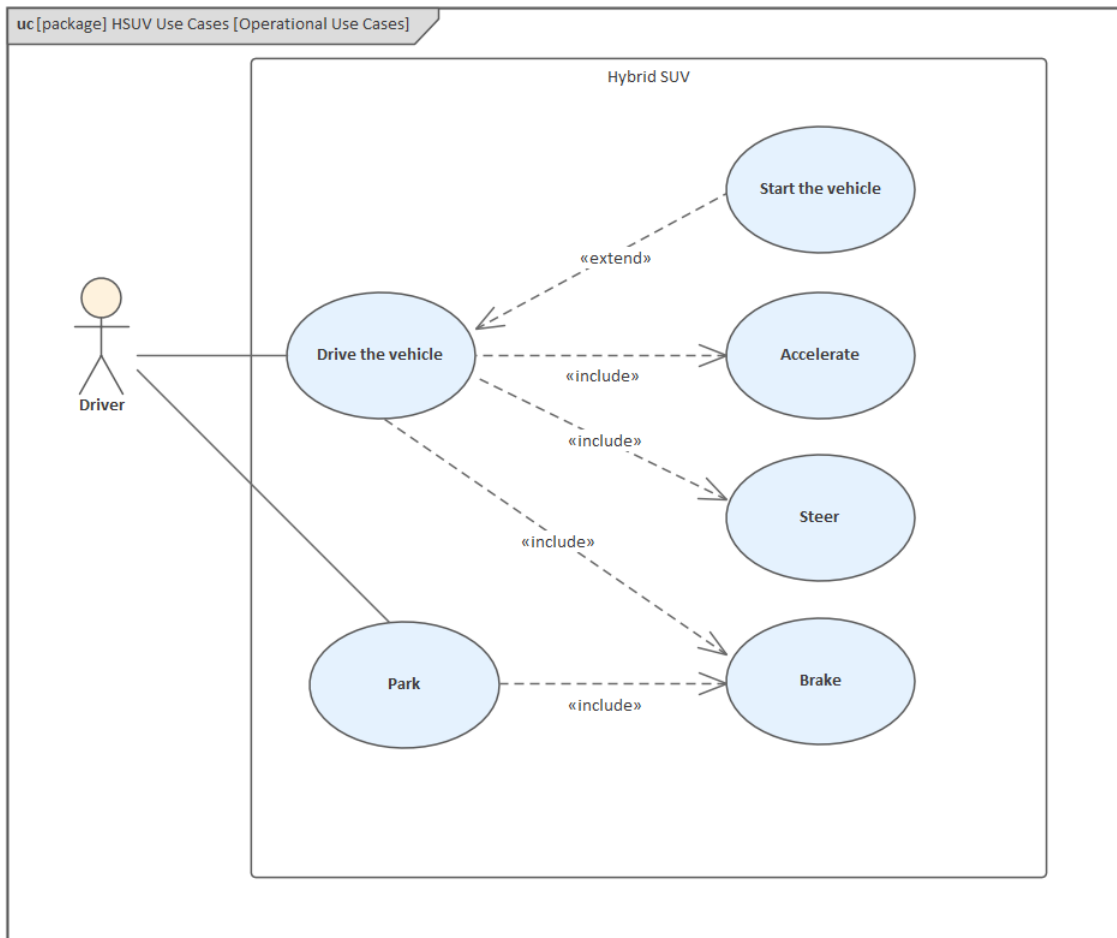
Item	SysML1.4::id	SysML1.4::text
<input checked="" type="checkbox"/> <b>Performance</b>	2	The Hybrid SUV shall have the braking, acceleration, and off-road capability of a typical SUV, but have dramatically better fuel economy.
<input checked="" type="checkbox"/> <b>Acceleration</b>	2.4	The Hybrid SUV shall have the acceleration of a typical SUV.
<input checked="" type="checkbox"/> <b>Braking</b>	2.1	The Hybrid SUV shall have the braking capability of a typical SUV.
<input checked="" type="checkbox"/> <b>FuelEconomy</b>	2.2	The Hybrid SUV shall have dramatically better fuel economy than a typical SUV.
<input checked="" type="checkbox"/> <b>OffRoadCapability</b>	2.3	The Hybrid SUV shall have the off-road capability of a typical SUV.

## Usage

Le diagramme Exigences peut être utilisé pour afficher une hiérarchie d'exigences en utilisant la relation de confinement permettant à un observateur de voir comment les relations structurelles des exigences. Il est cependant plus convaincant lorsque Exigences sont visualisées dans un diagramme avec d'autres éléments utilisant d'autres relations, y compris d'autres exigences. Un exemple de cela est la relation entre Exigences et les cas Test ou Exigences et les composants d'une solution.

## Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation est utilisé pour définir et visualiser les cas d'utilisation et les acteurs qui tirent valeur du système. Le diagramme de cas d'utilisation décrit la relation entre les acteurs et les cas d'utilisation, en enfermant le cas d'utilisation dans une Bordure qui définit la frontière du système ; les acteurs, par définition, se trouvent à l'extérieur de la Bordure . Bien que le diagramme de cas d'utilisation puisse paraître simpliste, il s'agit d'un outil de communication utile qui décrit la valeur ou les objectifs que les rôles externes obtiennent en interagissant avec le système. Chaque cas d'utilisation peut être détaillé avec des descriptions, des contraintes et un certain nombre de scénarios contenant des ensembles d'étapes exécutées en alternance par l'acteur et le système pour atteindre l'objectif souhaité.



### Éléments

Les principaux éléments qui peuvent apparaître dans diagrammes de cas d'utilisation sont :

- Bordure
- Acteur
- Cas d'utilisation
- Scénario

Les principaux connecteurs qui peuvent apparaître dans diagrammes de cas d'utilisation sont :

- Chemin de Communication
- Généralisation

- Comprend
- S'étend

## Outils

Outre le diagramme de cas d'utilisation lui-même, il existe une variété d'outils permettant de travailler avec les cas d'utilisation. Il s'agit notamment de :

- *Fenêtre de scénario* - qui permet de détailler les descriptions, les contraintes et l'étape de chaque scénario
- *Générateur de documentation* - qui permet de créer des documents d'entreprise, des rapports ou des documents ad hoc dans une variété de formats, notamment docx, pdf et rtf
- *Traçabilité* - qui fournit une vue hiérarchique des relations d'un élément avec d'autres éléments du modèle
- *Fenêtre de responsabilité* - qui fournit une vue composite des responsabilités importantes d'un élément, y compris les contraintes, Exigences et les scénarios
- *Matrice de relations* - qui permet de visualiser dans une matrice les connexions entre les exigences (ou d'autres éléments) et d'autres éléments tels que les besoins des parties prenantes

L'outil principal utilisé pour travailler avec les cas d'utilisation est la fenêtre Scénario, qui est une facilité complète et spécialement conçue pour travailler avec les cas d'utilisation et les scénarios. L'outil peut être utilisé pour définir les détails d'un cas d'utilisation et ses scénarios et contraintes, ce qui fournit une alternative productive à l'approche traditionnelle basée sur des documents texte pour définir les cas d'utilisation. Cela garantit que le diagramme de cas d'utilisation et les détails textuels des cas d'utilisation et de ses scénarios et contraintes sont tous contenus dans le même modèle et peuvent être tracés.

Si les cas d'utilisation sont requis dans un format de document pour des raisons contractuelles ou de processus, un Rapport de cas d'utilisation peut être généré automatiquement à partir des modèles à l'aide du moteur de documentation intégré.

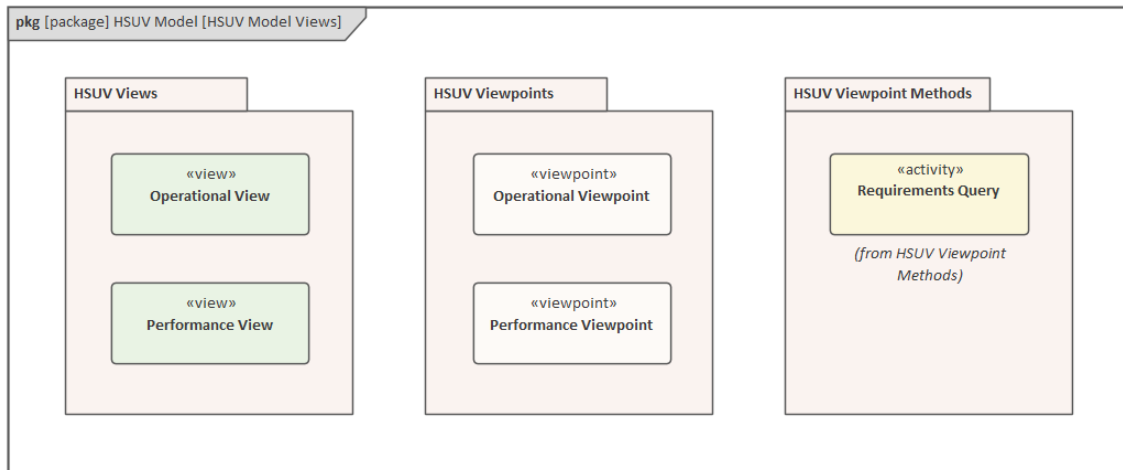
[Scenario Builder](#)

## Usage

Le diagramme de cas d'utilisation peut être utilisé pour définir les détails d'un cas d'utilisation, ses scénarios et ses contraintes. Il s'agit d'une alternative intéressante à l'approche traditionnelle basée sur des documents texte, couramment utilisée pour définir les cas d'utilisation. Cela garantit que le diagramme de cas d'utilisation et les détails textuels des cas d'utilisation, de leurs scénarios et de leurs contraintes sont tous contenus dans le même modèle et peuvent être tracés. Si les cas d'utilisation doivent être présentés dans un format de document pour des raisons contractuelles ou de processus, un Rapport de cas d'utilisation peut être généré automatiquement à partir des modèles à l'aide du moteur de documentation intégré.

# Diagramme Paquetage

Le diagramme Paquetage SysML est utilisé pour définir ou visualiser les Paquetages qui fournissent l'organisation fondamentale du référentiel. Ceux-ci peuvent inclure des espaces de noms et leurs sous-Packages et d'autres groupes d'éléments moins formellement définis. Les Paquetages qui apparaissent dans diagrammes peuvent également être visualisés dans la fenêtre Navigateur et leur hiérarchie peut être parcourue en développant et en réduisant l'arborescence.



L'élément principal représenté dans le diagramme Paquetage est le Paquetage lui-même, avec les éléments qu'il contient. Il existe un certain nombre de relations importantes entre Paquetages, notamment les Dépendances qui montrent qu'un Paquetage dépend d'un ou de plusieurs autres Paquetages. Paquetages peuvent être organisés en plusieurs types de hiérarchie différents.

## Éléments

Les principaux éléments qui peuvent apparaître dans diagrammes Paquetage sont :

- Modèle
- Bibliothèque d'Équipe
- Paquetage
- Vue
- Point Vue
- Partie prenante

Les principaux connecteurs qui peuvent apparaître dans diagrammes Paquetage sont :

- Se conformer
- Dépendance
- Importer
- Endiguement
- Réalisation
- Affiner
- Exposer

## Outils

Une variété d'outils peuvent être utilisés avec modélisation structurelle et diagrammes de définition Bloc , notamment :

- Documentation - qui permet de générer une documentation formelle ou informelle à partir du modèle dans une variété de formats
- Traçabilité - qui fournit une vue hiérarchique des relations d'un élément avec d'autres éléments du modèle
- Fenêtre de responsabilité - qui fournit une vue composite des responsabilités importantes d'un élément, y compris les contraintes, Exigences et les scénarios
- Matrice de relations - qui permet de visualiser les connexions entre Paquetages (ou d'autres éléments) et d'autres éléments tels que Exigences et les Cas d'Utilisation dans une matrice

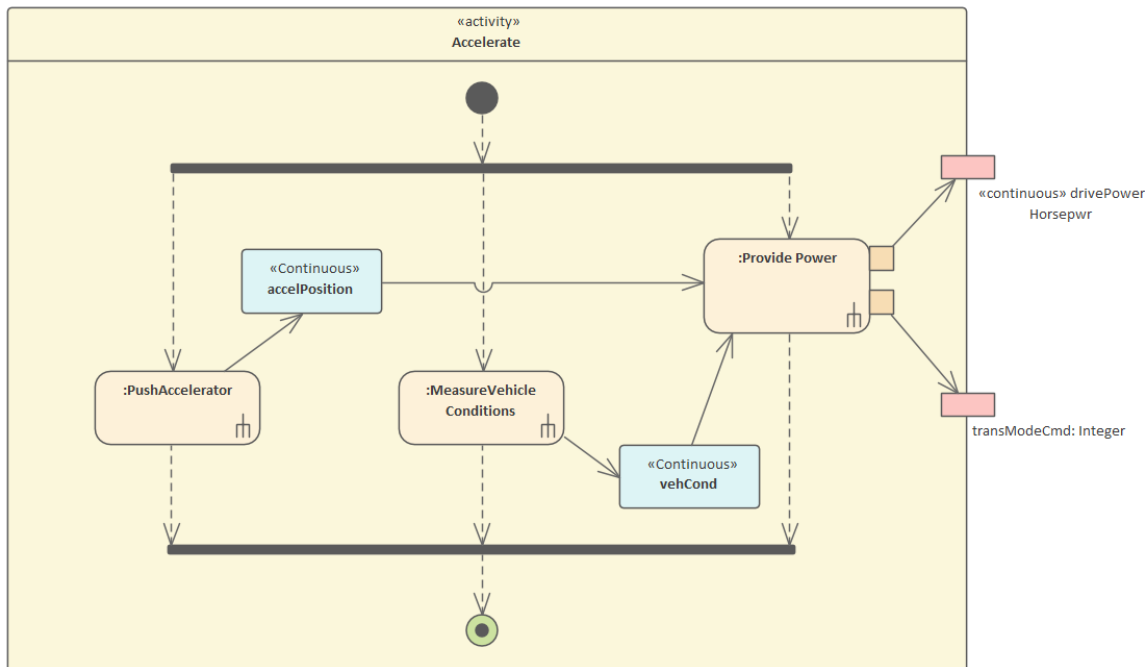
## Usage

Le diagramme Paquetage peut être utilisé pour décrire la relation entre Paquetages et les éléments qu'ils contiennent. Bien que les informations structurelles soient visibles dans la fenêtre Navigateur , il existe toute une série de relations qui peuvent exister entre Paquetages eux-mêmes et entre Paquetages et les éléments, qui ne peuvent pas être visualisées dans la fenêtre Navigateur . diagrammes Paquetage peuvent également être inclus dans la documentation et peuvent aider à orienter un public en lui donnant un aperçu d'une section de l' architecture ou de la conception d'une manière similaire à la présentation d'un tableau de contenu dans une publication.

## Diagrammes d'activité

Le diagramme d'activité est le diagramme de comportement le plus important et peut être utilisé pour modéliser un comportement basé sur un flux (discret ou continu) où les entrées sont converties en sorties en parcourant une séquence d'actions qui effectuent un travail sur les éléments. Ils sont analogues au diagramme de flux commun mais ont une sémantique plus sophistiquée et permettent également de relier les activités et les actions à des éléments tels que les blocs, Exigences et les cas d'utilisation.

Les actions qui apparaissent sur les diagrammes d'activité peuvent contenir pins d'entrée ou de sortie qui représentent les points d'interaction où les entrées sont alimentées dans une action et les sorties sont émises.



### Usage

Le diagramme d'activité peut être utilisé pour modéliser le comportement basé sur le flux et est similaire aux diagrammes de flux fonctionnels ou organigrammes largement disponibles qui ont été largement utilisés avant la conception de la spécification SysML. Ils sont généralement utilisés pour montrer la manière dont les parties du système se comportent, y compris l'entrée et la sortie des éléments et des signaux.

### Éléments

Les principaux éléments qui peuvent apparaître dans diagrammes d'activité sont :

- Activité
- Action (diverses formes)
- Action Pin
- Partition
- Nœud Object

- Nœud tampon central
- Magasin de données
- Décision
- Fusionner
- Initial
- Final

Les principaux connecteurs qui peuvent apparaître dans diagrammes d'activité sont :

- Contrôle du flux
- Flux Object
- Interrompre le flux
- Dépendance

## Outils

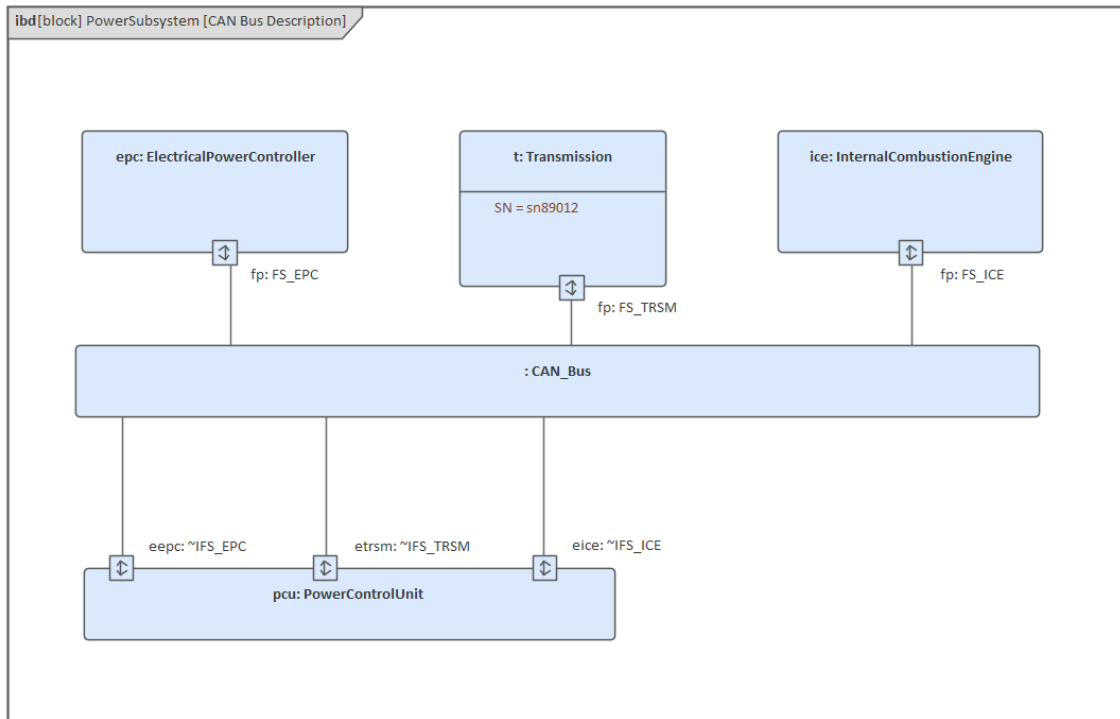
Une variété d'outils peuvent être utilisés avec modélisation comportementale et diagrammes d'activité, notamment :

- Filtres de Diagramme - qui permettent à un utilisateur de filtrer les éléments du diagramme pour obtenir une mise au point plus spécifique
- Pan et zoom - qui permet à un modélisateur ou à un spectateur de se déplacer facilement dans de grands diagrammes
- Importation et exportation de feuilles de calcul (CSV) - qui permet d'importer ou d'exporter le contenu des feuilles de calcul à partir du modèle
- Documentation - qui permet de générer une documentation formelle ou informelle à partir du modèle dans une variété de formats
- Traçabilité - qui fournit une vue hiérarchique des relations d'un élément avec d'autres éléments du modèle
- Fenêtre de responsabilité - qui fournit une vue composite des responsabilités importantes d'un élément, y compris les contraintes, Exigences et les scénarios
- Matrice de relations - qui permet de visualiser les connexions entre les activités (ou d'autres éléments) et d'autres éléments tels que les blocs, Exigences et les cas d'utilisation dans une matrice



## Diagramme Interne de Bloc

Le diagramme Bloc interne permet de visualiser la composition d'un bloc à l'aide des propriétés des parties connectées entre elles à l'aide de ports et de connecteurs. Le diagramme est utile pour montrer la composition d'un Bloc (représenté par le cadre diagramme ) et le flux des entrées et des sorties entre les différentes parties qui composent le bloc. Si nécessaire, le sens du flux peut être indiqué sur les connecteurs.



## Éléments

Les principaux éléments qui peuvent apparaître dans diagrammes de définition Bloc sont :

- Propriété
- Propriété du connecteur
- Propriété distribuée
- Propriété de flux
- Référence liée
- Multiplicité du chemin final
- Signal
- Port

Les principaux connecteurs qui peuvent apparaître dans diagrammes de définition Bloc sont :

- Flux Item
- Connecteur
- Connecteur de liaison
- Dépendance

## Outils

Une variété d'outils peuvent être utilisés avec modélisation structurelle et diagrammes Bloc internes, notamment :

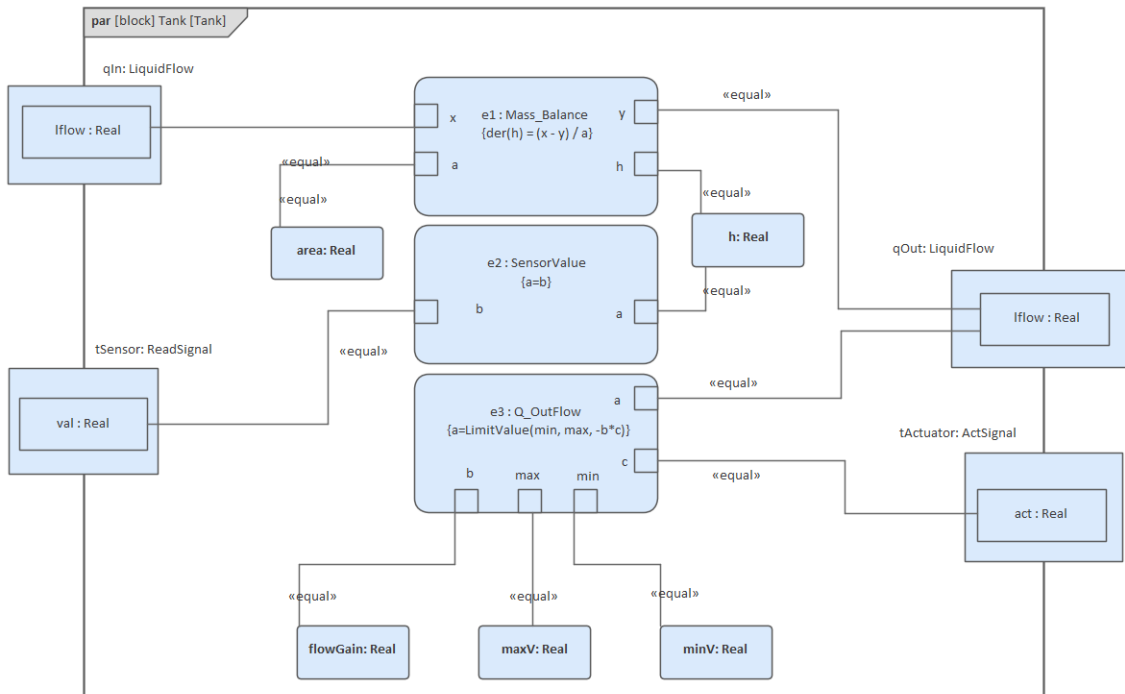
- Filtres de Diagramme - qui permettent à un utilisateur de filtrer les éléments du diagramme pour obtenir une mise au point plus spécifique
- Pan et zoom - qui permet à un modélisateur ou à un spectateur de se déplacer facilement dans de grands diagrammes
- Importation et exportation de feuilles de calcul (CSV) - qui permet d'importer ou d'exporter le contenu des feuilles de calcul à partir du modèle
- Documentation - qui permet de générer une documentation formelle ou informelle à partir du modèle dans une variété de formats
- Traçabilité - qui fournit une vue hiérarchique des relations d'un élément avec d'autres éléments du modèle
- Fenêtre de responsabilité - qui fournit une vue composite des responsabilités importantes d'un élément, y compris les contraintes, Exigences et les scénarios
- Matrice de relations - qui permet de visualiser les connexions entre les blocs (ou d'autres éléments) et d'autres éléments tels que Exigences et les cas d'utilisation dans une matrice

## Usage

Le diagramme Bloc interne est utilisé pour modéliser la structure interne d'un bloc, y compris ses parties et la relation entre ces parties.

## Diagramme Paramétriques

Le Diagramme Paramétriques SysML est un type de Diagramme Interne de Bloc (avec certaines restrictions) qui est utilisé pour modéliser une équation avec des paramètres. Il s'agit d'un outil important qui peut être utilisé pour décrire les équations et leurs paramètres et qui est important lors de la réalisation d'une analyse de compromis et de l'évaluation des alternatives de conception, car ils peuvent être combinés en systèmes d'équations et liés aux mesures d'efficacité MOE.



diagrammes Paramétriques décrivent l'utilisation de blocs de contraintes et fournissent un mécanisme permettant d'intégrer l'analyse technique telle que les performances et la fiabilité et d'autres facteurs d'intérêt avec d'autres modèles et diagrammes SysML.

diagrammes Paramétriques définissent la manière dont les blocs de contraintes sont utilisés pour contraindre les propriétés d'un autre bloc. L'utilisation d'une contrainte est censée lier les paramètres de la contrainte (par exemple  $F=m*a$ ), tels que  $F$ ,  $m$  et  $a$ , à des propriétés spécifiques d'un bloc, telles qu'une masse et une accélération, qui fournissent des valeurs pour les paramètres.

### Éléments

Les principaux éléments pouvant apparaître dans diagrammes Paramétriques sont :

- Propriété de contrainte
- Propriété
- Fonction objective
- Mesure de l'efficacité

Les principaux connecteurs pouvant apparaître dans diagrammes Paramétriques sont :

- Connecteur
- Connecteur de liaison
- Flux Item
- Dépendance

## Outils

Une variété d'outils peuvent être utilisés avec modélisation structurelle et diagrammes Bloc internes, notamment :

- Intégration Modelica - qui fournit un mécanisme de simulation,
- Filtres de Diagramme - qui permettent à un utilisateur de filtrer les éléments du diagramme pour obtenir une mise au point plus spécifique,
- Pan et zoom - qui permet à un modélisateur ou à un spectateur de se déplacer facilement dans de grands diagrammes ,
- Importation et exportation de feuilles de calcul (CSV) - qui permet d'importer ou d'exporter le contenu des feuilles de calcul à partir du modèle,
- Documentation - qui permet de générer une documentation formelle ou informelle à partir du modèle dans une variété de formats,
- Traçabilité - qui fournit une vue hiérarchique des relations d'un élément avec d'autres éléments du modèle,

## Usage

Le diagramme Paramétriques peut être utilisé pour montrer comment les propriétés physiques d'un système sont contraintes en spécifiant un réseau de contraintes qui représentent des expressions mathématiques telles que  $\{F=m*a\}$  et  $\{a=dv/dt\}$ .

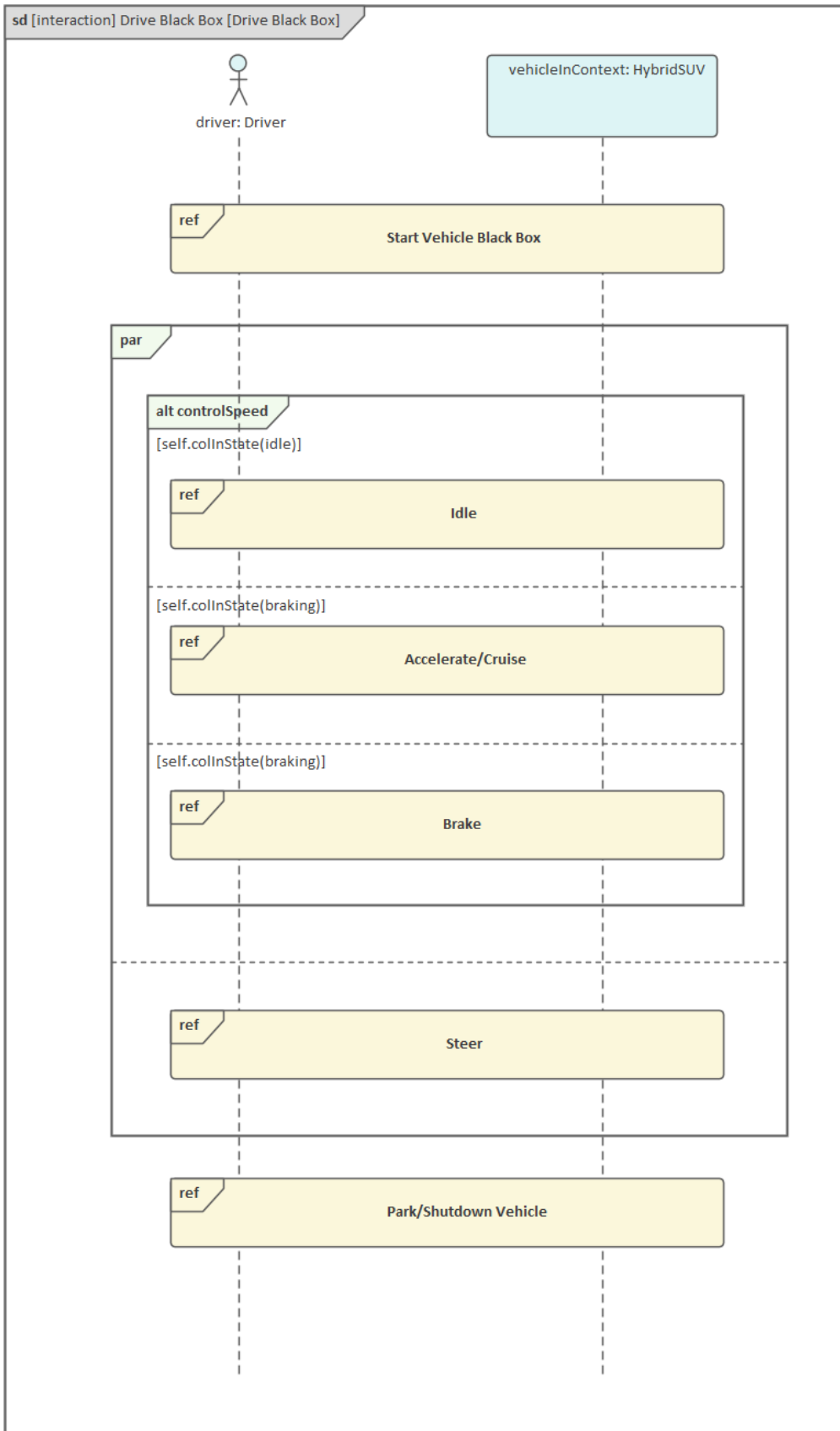
Ils peuvent également être utilisés pour l'analyse des compromis, où un Bloc de contraintes peut définir une fonction objective utilisée pour faire une comparaison entre des solutions alternatives.

Les paramètres de performance critiques et leurs relations avec d'autres paramètres peuvent être modélisés, puis suivis tout au long du cycle de vie du système.

## Diagramme de Séquence

Un diagramme Séquence est un type de diagramme d'interaction qui montre l'interaction ordonnée dans le temps entre les objets. Le diagramme comporte deux axes : l'axe vertical représente le temps et l'axe horizontal représente les objets qui participent à l'interaction, généralement ordonnés de manière à mieux mettre en évidence l'interaction. Ces diagrammes trouvent leur origine dans la modélisation des interactions logicielles, mais ils peuvent être utilisés avec l'ingénierie des systèmes pour prescrire la manière dont les éléments (tels que les blocs) doivent interagir, ou pour décrire la manière dont ils interagissent dans la pratique.

Ce diagramme Séquence montre les interactions et la séquence des flux de messages entre un conducteur et un véhicule. Le diagramme exprime les interactions nécessaires pour le cas d'utilisation « Conduire le véhicule ». L'interaction appartient au Bloc « AutomotiveDomain ».



## Éléments

Les principaux éléments pouvant apparaître dans diagrammes Paramétriques sont :

- Séquence
- Fragment
- Point final
- Porte Diagramme
- State / Suite

Les principaux connecteurs pouvant apparaître dans diagrammes Paramétriques sont :

- Message
- Message personnel
- Récursivité
- Dépendance

## Outils

Une variété d'outils peuvent être utilisés avec modélisation comportementale et diagrammes d'activité, notamment :

- Filtres de Diagramme - qui permettent à un utilisateur de filtrer les éléments du diagramme pour obtenir une mise au point plus spécifique,
- Pan et zoom - qui permet à un modélisateur ou à un spectateur de se déplacer facilement dans de grands diagrammes
- Importation et exportation de feuilles de calcul (CSV) - qui permet d'importer ou d'exporter le contenu des feuilles de calcul à partir du modèle,
- Documentation - qui permet de générer une documentation formelle ou informelle à partir du modèle dans une variété de formats,
- Traçabilité - qui fournit une vue hiérarchique des relations d'un élément avec d'autres éléments du modèle,
- Fenêtre de responsabilité - qui fournit une vue composite des responsabilités importantes d'un élément, y compris les contraintes, Exigences et les scénarios

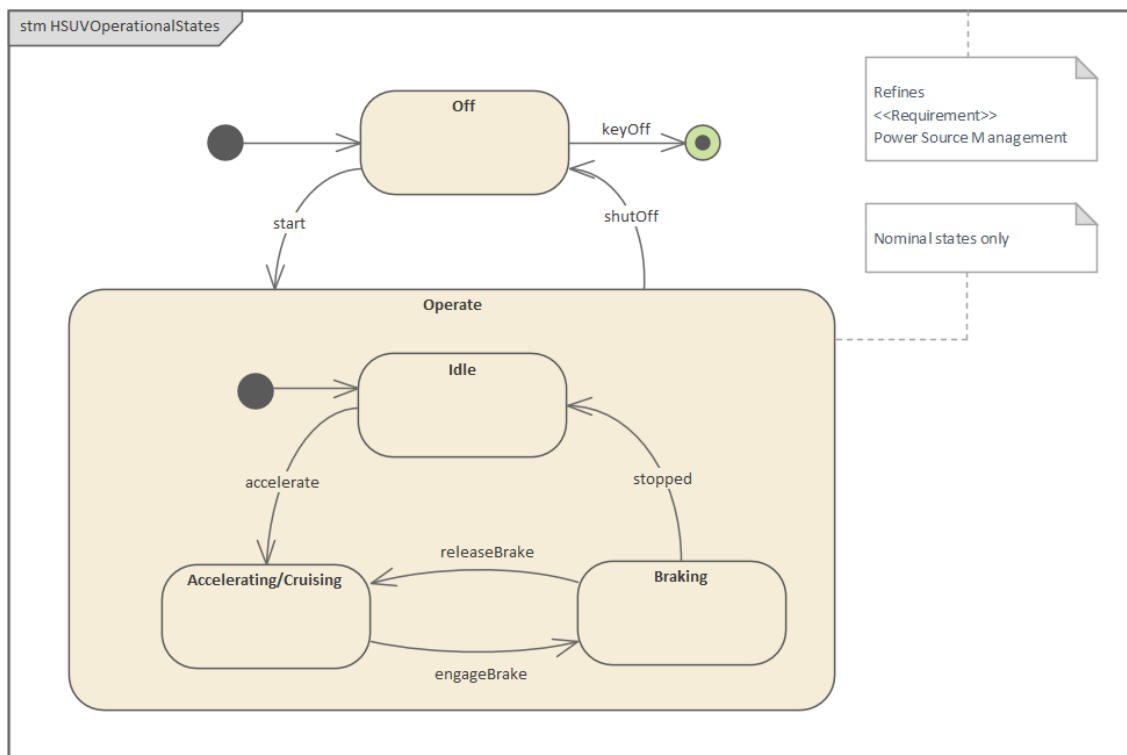
Matrice de relations - qui permet de visualiser les connexions entre les activités (ou d'autres éléments) et d'autres éléments tels que les blocs, Exigences et les cas d'utilisation dans une matrice.

## Usage

Le fait même que nous modélisons un système implique qu'il présente un degré de complexité qui ne peut être géré sans l'utilisation d'outils. Les diagrammes Séquence peuvent être utilisés pour représenter des scénarios système montrant comment les parties d'un système interagissent ensemble pour atteindre un résultat spécifique. On dit que les messages sont échangés entre des lignes de vie représentant la durée de vie de l'objet, les messages représentent des opérations ou des signaux envoyés et reçus par les objets.

## Diagramme Statemachine

Un diagramme Statemachine est un moyen efficace de présenter des informations sur la durée de vie d'un élément du système tel qu'un Bloc . Il peut être utilisé pour décrire les conditions importantes ( States ) qui se produisent au cours de la durée de vie ou des cycles d'une entité. En général, seules les entités qui ont des étapes importantes dans leur durée de vie sont modélisées avec diagrammes Statemachine . On dit que l'entité passe d'un State à un autre comme spécifié par le Statemachine . Déclencheurs et Événements peuvent être décrits pour permettre la transition d'état et des gardes peuvent être définis pour restreindre le changement d'état. Chaque State peut définir les comportements qui se produisent à l'entrée, pendant et à la sortie de l' State .



## Éléments

Les principaux éléments pouvant apparaître dans diagrammes Paramétriques sont :

- State
- Statemachine
- Initial
- Final
- Choix
- Histoire
- Fourche et Joindre

Les principaux connecteurs pouvant apparaître dans diagrammes Paramétriques sont :

- Transition
- Dépendance

## Outils



Outre le diagramme Statemachine lui-même, de nombreux outils sont disponibles pour travailler avec les diagrammes Statemachine . Il s'agit notamment de :

- Éditeur Tableau State - Permet de visualiser le diagramme Statemachine dans un tableau qui, pour certains analystes, est plus facile à comprendre qu'un diagramme ; il contient les mêmes informations que le diagramme et peut être visualisé de différentes manières
- Simulation Dynamique - Permet de visualiser Statemachines , montrant comment une entité passe d'un état à un autre
- Statemachines Exécutables - En plus d'utiliser le moteur de simulation et de permettre la visualisation Statemachines , fournir une implémentation complète spécifique au langage qui peut former le « moteur » comportemental pour plusieurs produits logiciels sur plusieurs plates-formes

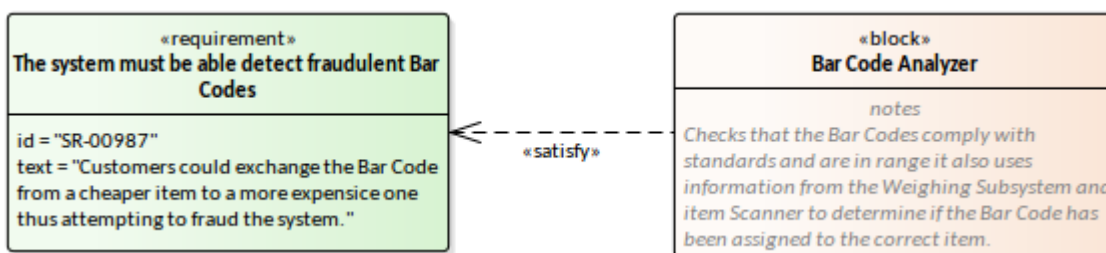
## Usage

Les systèmes complexes sont souvent composés d'entités telles que des blocs qui ont un comportement complexe et peuvent avoir des durées de vie difficiles à comprendre.

Statemachines peuvent être utilisés pour décrire les conditions importantes ( States ) qui se produisent au cours de la vie ou des cycles d'une entité. En règle générale, seules les entités qui ont des étapes importantes dans leur vie sont modélisées avec diagrammes Statemachine . Ces diagrammes donnent un aperçu de la manière dont une entité passe d'un état à un autre, en ignorant les conditions qui ne sont pas importantes pour l'analyse.

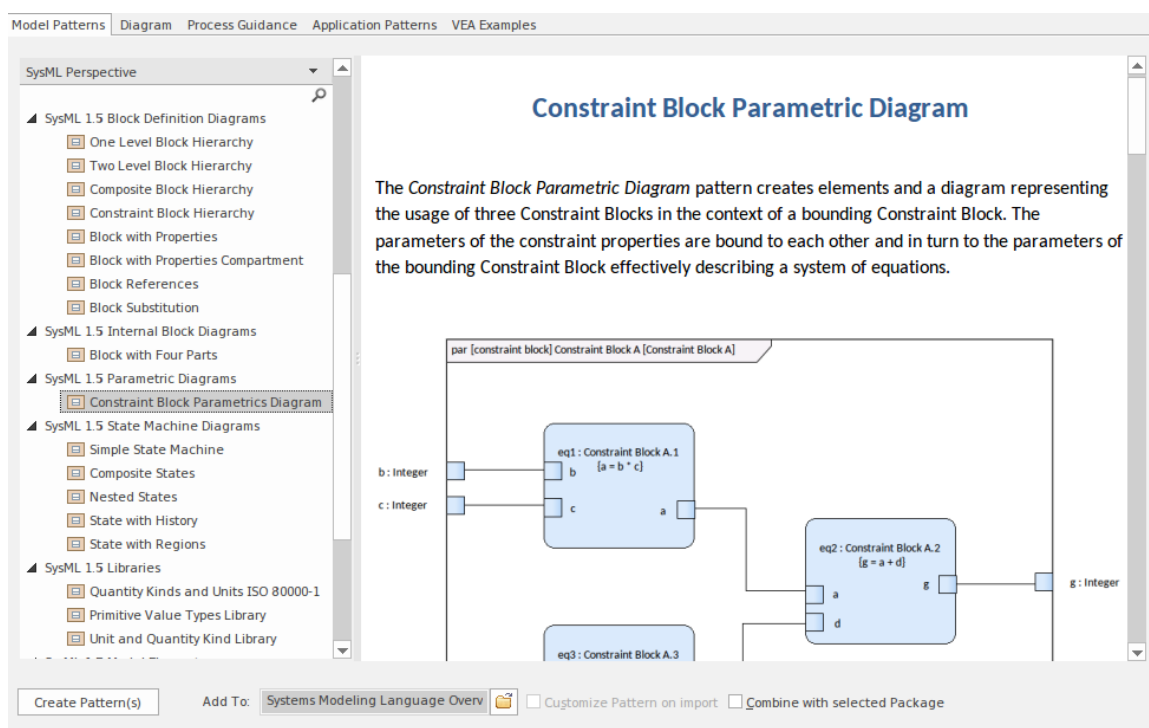
# Présentation Langage Modélisation des Systèmes

L'ingénierie des systèmes basée sur les Modèles annonce une nouvelle ère de communication et de collaboration en temps réel. Elle apporte avec elle le concept du modèle en tant que machine plutôt qu'une armoire pleine de documents. Une machine qui peut effectuer des tâches telles que valider des exigences, générer des simulations paramétriques d'équations mathématiques et physiques complexes, donner vie à State Machines exécutables et simuler la logique métier et décisionnelle, évoluer en réponse aux révisions et créer de la documentation, pour n'en citer que quelques-unes. Ces avantages sont réalisés grâce à la puissance d'Enterprise Architect et parce qu'un langage standard et partagé est utilisé pour créer les modèles - le Langage Modélisation des Systèmes, communément abrégé en SysML.

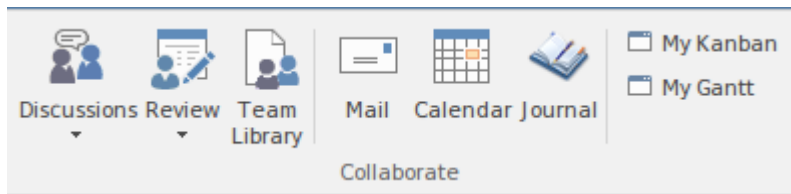


SysML permet aux humains et aux machines de comprendre les modèles : les humains ajoutent de l'ingéniosité, de l'ingénierie et de la conception et les machines effectuent les tâches fastidieuses et sujettes aux erreurs telles que la validation, font le gros du travail comme génèrent des simulations paramétriques et effectuent *des analyses de type « et si »*, et effectuent les tâches les plus banales telles que la recherche et la génération de rapports.

L'acquisition d'une langue n'est pas gratuite, mais elle peut se faire sans douleur ni frustration et sans ce que certains sceptiques appellent un « flair pour les langues ». Enterprise Architect sera également un ami qui vous aidera à formation la langue en fournissant de nombreux outils pour vous aider dans cette formation, ainsi qu'une bibliothèque riche et complète de motifs qui vous aideront à démarrer, en vous assurant de créer des modèles conformes aux meilleures pratiques du secteur.



Lorsque vous commencez votre parcours avec Enterprise Architect vous devenez immédiatement et sans effort membre d'une vaste communauté internationale d'utilisateurs et de praticiens, qui travaillent avec l'outil au quotidien pour spécifier, concevoir, mettre en œuvre et support des modèles d'ingénierie système utilisés pour résoudre des problèmes du monde réel. Bon nombre de ces problèmes et opportunités sont complexes et souvent apparemment insolubles, mais peuvent être résolus par la collaboration de modélisateurs appliquant le SysML pour exprimer et résoudre des problèmes.

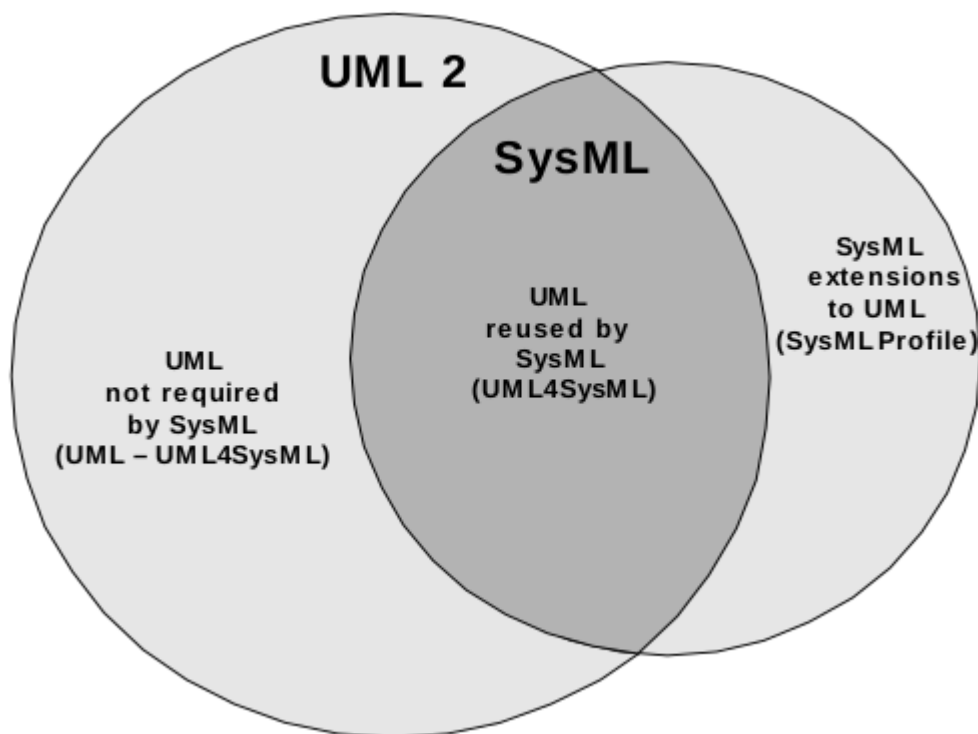


Enterprise Architect facilite cette collaboration de manière transparente grâce à ses riches plates-formes de bureau et Cloud qui contribuent à garantir que les modèles sont à la fois robustes et expressifs, et le résultat non pas d'un seul, mais de plusieurs ingénieurs et autres parties prenantes, travaillant ensemble, quelle que soit la langue naturelle qu'ils parlent, l'appareil qu'ils utilisent ou l'endroit où ils se trouvent dans le monde.

# Architecture du langage

Le Systems Modeling Language (SysML) est fondamentalement un ensemble de symboles conventionnels qui permettent aux humains et aux outils de communiquer sur l'ingénierie des systèmes. Il s'agit d'une norme internationale qui définit et décrit un langage de modélisation à usage général pour l'ingénierie des systèmes. Enterprise Architect est l'un des principaux outils au monde qui implémente cette norme et permet aux ingénieurs systèmes d'appliquer l'approche connue sous le nom d'Ingénierie des Systèmes Modèles Basée. En outre, Enterprise Architect propose des fonctionnalités d'outils qui supportent un large éventail d'aspects auxiliaires de la pratique et de la gestion de l'ingénierie. Nous explorerons ces fonctionnalités d'outils utiles et productives tout au long de ce guide.

Le langage SysML est basé sur un autre standard, le Unified Modeling Language (UML), adopté et utilisé par les ingénieurs logiciels depuis la fin des années 90. C'est important, car de nombreux projets d'ingénierie des systèmes impliquent à la fois des aspects système et logiciel et donc les ingénieurs système et logiciel sont capables de comprendre les modèles des autres, ce qui conduit à une plus grande transparence, à moins de risques d'erreurs et à des constructions de langage mutuellement intelligibles, ce qui se traduit par un système moins susceptible de tomber en panne ou de présenter des défauts. Ce diagramme de Venn montre la relation entre les deux normes. SysML réutilise les Diagrammes de cas d'utilisation, d'activité et Séquence.



## Exigences poussées

La création du Systems Modeling Language (SysML) a été motivée par les besoins des utilisateurs ; la conception du SysML a répondu aux besoins énoncés dans la *demande de proposition pour le Unified Modeling Language pour Ingénierie des Systèmes*. Ce document spécifie une personnalisation d'UML pour l'Ingénierie des Systèmes (SE) et stipule que cette personnalisation doit supporter la modélisation d'une large gamme de systèmes, qui peuvent inclure du matériel, des logiciels, des données, du personnel, des procédures et facilités. Le document stipule :

« La personnalisation d'UML pour l'Ingénierie des Systèmes devrait supporter l'analyse, la spécification, la conception et la vérification de systèmes complexes en :

- Capturer les informations du système de manière précise et efficace pour permettre leur intégration et leur réutilisation dans un contexte plus large

- Analyser et évaluer le système spécifié, pour identifier et résoudre les exigences du système et les problèmes de conception, et pour support les compromis
- Communiquer les informations du système de manière correcte et cohérente entre les différentes parties prenantes et participants

Les concepteurs d' Enterprise Architect ont lu en détail ces documents et la spécification SysML qui en résulte et ont créé un outil sophistiqué et hautement utilisable qui implémente toutes ces exigences et ajoute un riche ensemble de fonctionnalités supplémentaires pour assurer le succès technique et commercial d'une organisation.



# OMG Systems Modeling Language™

## Version 1.5

Pour qu'un langage soit utile et pertinent, il doit évoluer en réponse aux besoins de ses communautés d'utilisateurs. Pour ce faire, la spécification SysML est régulièrement mise à jour et les équipes de Sparx Systems mettent également à jour et étendent Enterprise Architect pour garantir sa conformité avec la norme en constante évolution et, plus fondamentalement, répondre aux divers besoins de sa communauté d'utilisateurs.

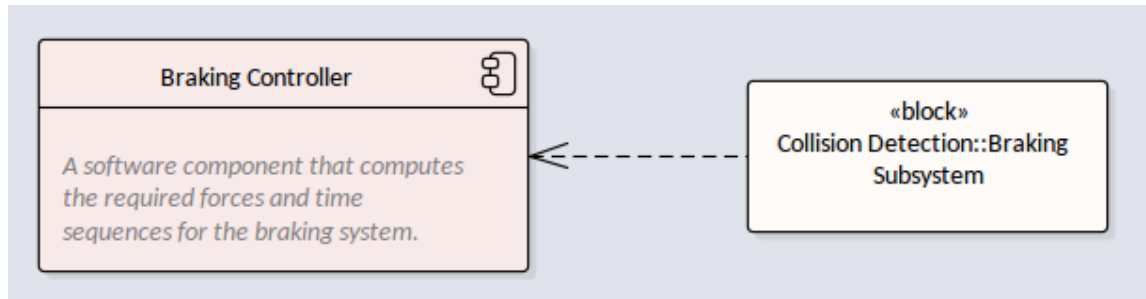
## Réutilisation et extensions Unified Modeling Language

Le Systems Modeling Language (SysML) est construit sur la base du Unified Modeling Language (UML) . L' UML a été ratifié et adopté par l' Object Management Group (OMG), qui continue d'être le gardien de la spécification. En 2005, UML a également été publié par l'Organisation internationale de normalisation (ISO) en tant que norme ISO approuvée. Le langage a fourni une spécification pour les systèmes centrés sur les logiciels modélisation . Le langage SysML remonte à 2001 et trouve ses origines dans une spécification open source, mais lorsque le Conseil international d' Ingénierie des Systèmes (INCOSE) a commencé à travailler avec l'OMG, une version finale du SysML a été adoptée par l'OMG en 2006.

À bien des égards, SysML est théoriquement le langage le plus primitif, car il s'agit d'un langage modélisation à usage général, tandis UML est plus spécialisé, étant conçu pour modélisation de systèmes centrés sur les logiciels. Cependant, l'histoire et la genèse des langages ont inversé cette position. En pratique, SysML a été créé en utilisant le système de profilage UML et est un sous-ensemble étendu de l' UML . Cela signifie que SysML n'a pas repris l'intégralité d' UML et qu'il a également défini des constructions de langage supplémentaires. Le diagramme de Venn que nous avons vu dans une section précédente décrit mathématiquement les deux ensembles de constructions de langage qui se croisent.

L'implémentation de la spécification SysML Enterprise Architect est hautement conforme, les développeurs travaillant en étroite collaboration avec la spécification et en communication constante avec les experts de l'industrie, les leaders d'opinion et les communautés d'ingénierie des systèmes dans un large éventail d'industries. Cela a donné lieu à un outil de classe mondiale qui non seulement implémente la spécification, mais fournit également une large gamme d'outils supplémentaires tels que Statemachines Exécutables , des Simulations Paramétriques , des Diagrammes de Gantt, des

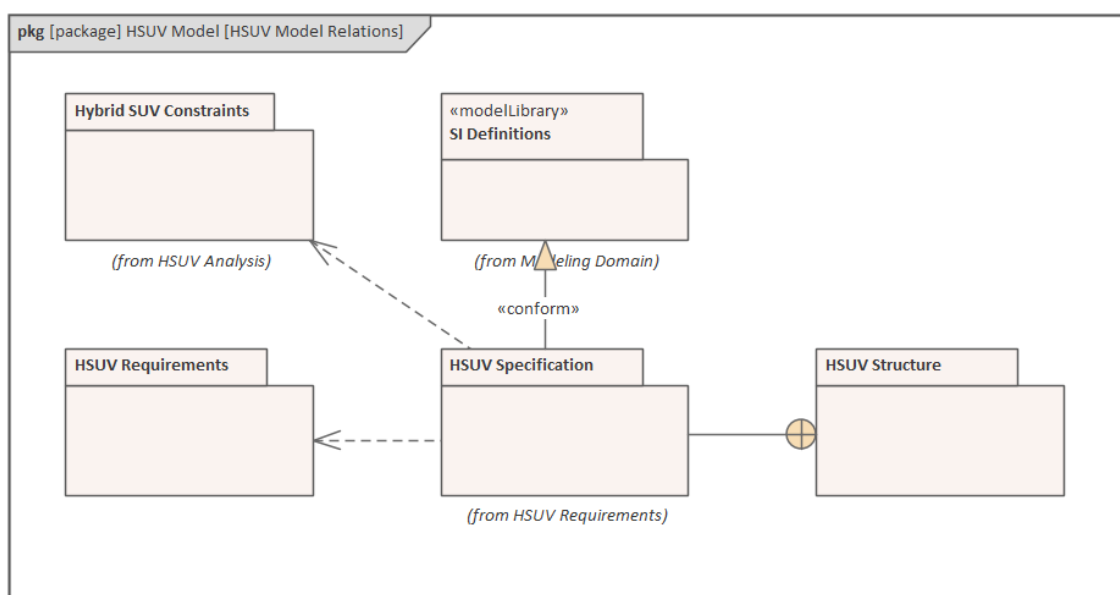
Tableaux Kanban, des Mind Mapping, des Modèles stratégiques et littéralement des centaines d'autres fonctionnalités . En outre, l'interaction entre les problèmes et les solutions d'ingénierie système et logicielle ne cesse de croître dans un large éventail de disciplines, des systèmes ferroviaires aux systèmes aéronautiques, en passant par les systèmes énergétiques et bien d'autres encore. Enterprise Architect occupe une position unique en raison de ses formidables fonctionnalités prenant en charge ces deux disciplines et de ses atouts en tant qu'outil d'architecture.



### Partitionnement avec Paquetages

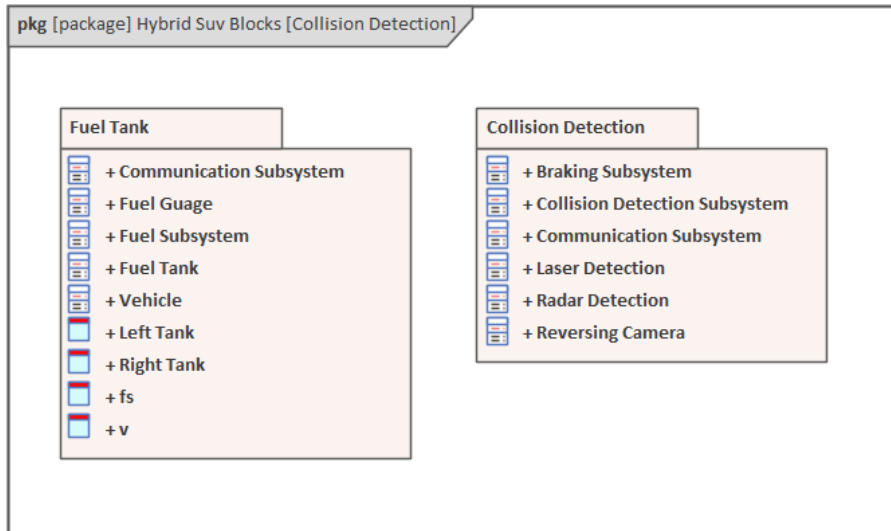
Paquetages constituent l'unité fondamentale de partitionnement du langage et sont conçus pour éviter les dépendances circulaires. Le langage est formellement partitionné en ensembles d'éléments de modèle qui regroupent les éléments de manière logique et permettent à un utilisateur du langage de comprendre les éléments comme une collection d'unités linguistiques.

Ils constituent également l'unité structurelle fondamentale des modèles définis par l'utilisateur et agissent comme un mécanisme à usage général utilisé pour regrouper des éléments en fonction de facteurs définis par l'utilisateur. Formellement, ils peuvent être utilisés pour spécifier un espace de noms, ce qui est important dans certaines constructions modélisation telles que la définition de schémas XML ou la génération de code. Paquetages peuvent être créés et visualisés soit dans la fenêtre Navigateur , soit dans diagrammes , et les deux emplacements offrent différentes façons de travailler avec les Paquetages . Diagrammes sont utiles pour afficher visuellement le contenu des Paquetages ou pour décrire les relations qui existent entre Paquetages .

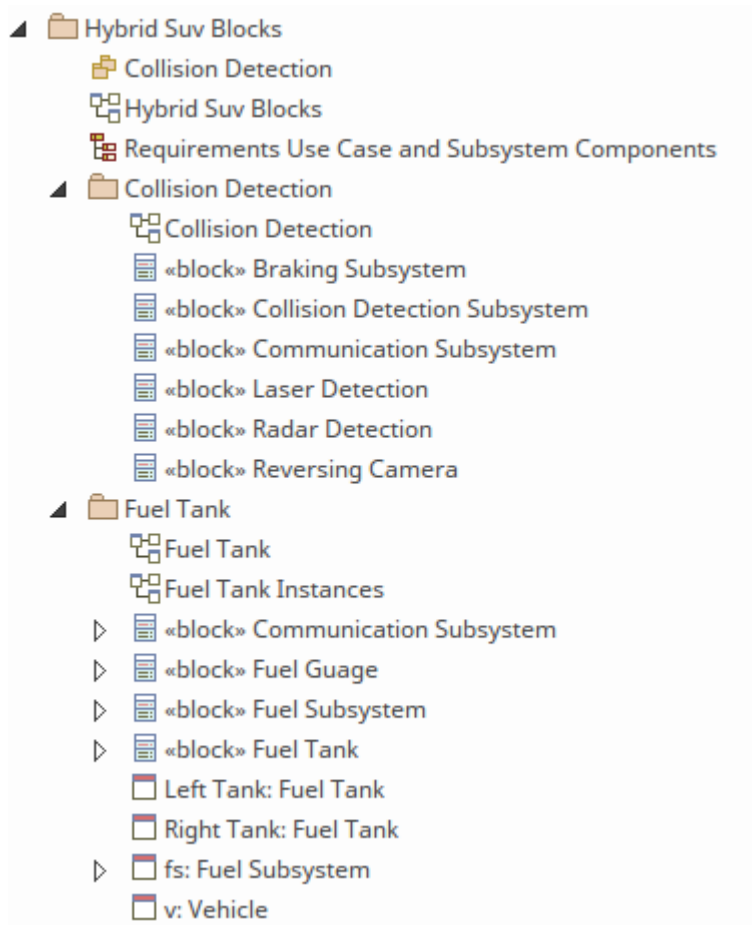


Architect propose de nombreuses façons d'afficher Paquetages dans diagrammes qui aideront les utilisateurs à comprendre les relations structurelles entre Paquetages et les éléments et diagrammes qu'ils contiennent. Lorsqu'un

Paquetage est inclus dans un diagramme , l'outil permet à l'utilisateur de choisir parmi un certain nombre d'options d'affichage et la visibilité du compartiment peut être modifiée pour afficher le contenu Paquetage . Dans ce diagramme l'auteur souhaite afficher le contenu de deux Paquetages qui ont une importance dans le cas peu probable d'une collision. L'option « Afficher le contenu Paquetage » a été sélectionnée dans la visibilité du compartiment d'éléments, ce qui permet de savoir clairement quels éléments sont contenus dans chaque Paquetage .

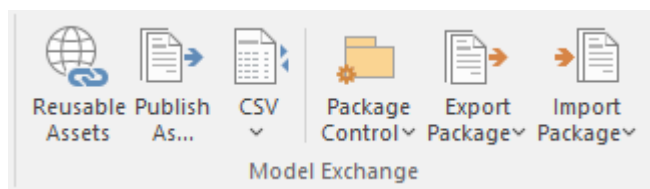


Les mêmes Paquetages et leur contenu peuvent être visualisés dans la fenêtre Navigateur , il est important de se rappeler que s'il est possible d'inclure les diagrammes dans des publications telles que des rapports, le contenu de la fenêtre Navigateur ne serait pas visible dans ces documents.

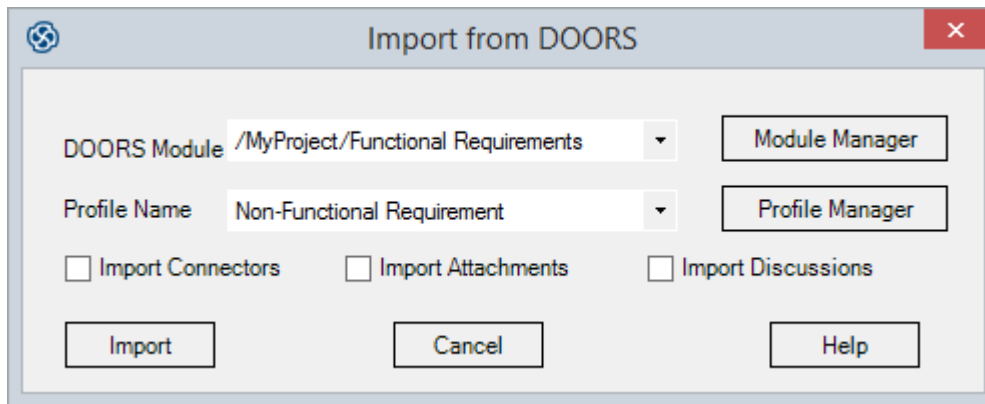


## Interopérabilité et Échange de Modèles

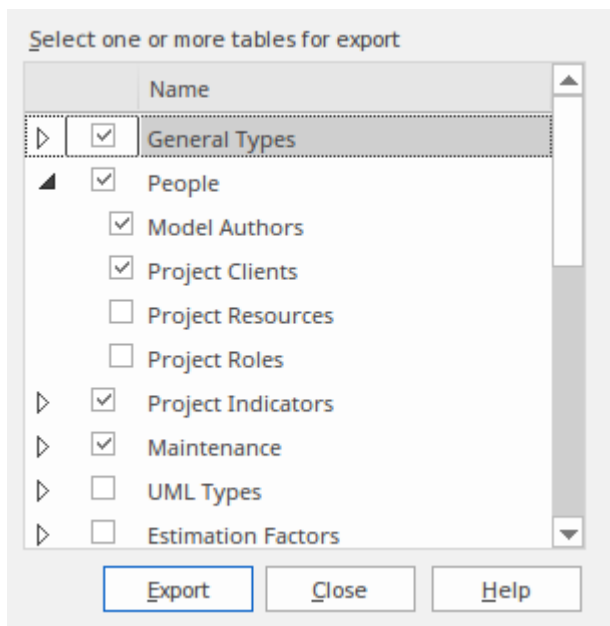
Enterprise Architect est l'un des principaux outils SysML avec un ensemble de fonctionnalités requises, mais les concepteurs sont conscients que les organisations auront besoin d'utiliser une variété d'outils pour accomplir les tâches commerciales et d'ingénierie complexes auxquelles chaque organisation est confrontée au XXI<sup>e</sup> siècle. Pour garantir que les informations d'ingénierie et commerciales importantes sont disponibles pour être échangées avec d'autres outils et plates-formes, il existe support riche pour l'échange de modèles conformément à la norme d'échange de données ISO 10303-233 pour support l'interopérabilité entre d'autres outils d'ingénierie. Ceci est implémenté sur la base de la capacité d'échange UML XMI, qui est prise en charge dans l'outil au niveau Paquetage , permettant à tout Paquetage et à sa hiérarchie contenue d'être échangés avec d'autres outils conformes.



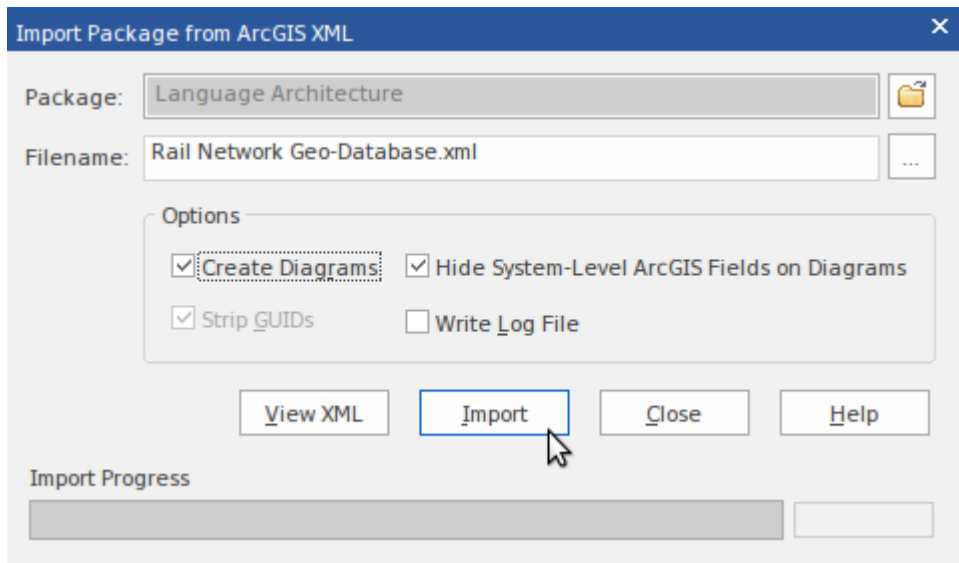




Enterprise Architect va plus loin et propose des mécanismes d'échange avec une grande variété d'outils métier, de gestion de projet, d'analyse et de livraison de projet. Cela se fait au niveau des outils modélisation avec la possibilité d'échanger des données contenues dans des feuilles de calcul au format CSV et du texte dans des traitements de texte. Les données de référence telles que les listes de priorités, de statuts, de complexités, de contraintes et d'autres données telles que les glossaires, les rôles et les auteurs, les calendriers et bien plus encore peuvent toutes être importées et exportées à partir du référentiel.

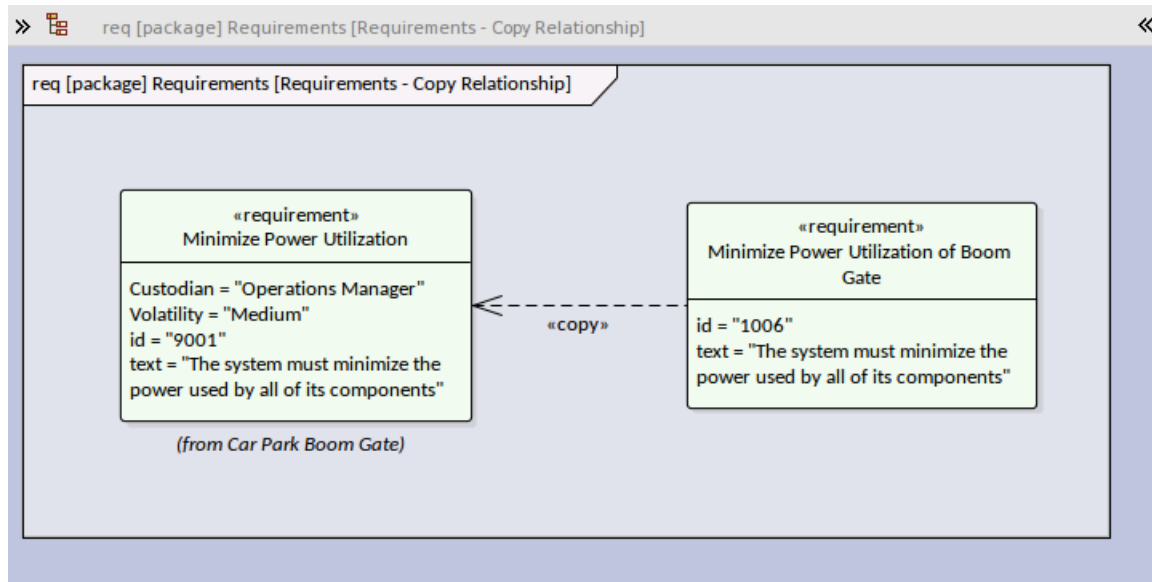


Les informations géospatiales constituent un ensemble de données essentiel dans un monde où la géolocalisation est un aspect important de presque tous les projets et initiatives. Enterprise Architect fournit un échange de données avec les principaux outils modélisation géospatiale, permettant de visualiser et de gérer ensemble deux ensembles de données jusque-là disparates et hétérogènes.



## Concepts grammaticaux clés

Le SysML, comme son parent l'UML, est un langage visuel où diagrammes sont au cœur de la stratégie de communication du langage. Même si l'accent du langage est mis sur cette codification visuelle et la transmission des idées, le langage dispose également de facilités pour exprimer les idées sous forme textuelle, ce qui constitue un complément important aux mécanismes visuels. Un certain nombre d'éléments, tels que l'élément Requirement, ont une forme visuelle, mais les détails de l'Requirement sont écrits dans une propriété appelée Requirement 'text' comme le montre ce diagramme.



Enterprise Architect a également été soigneusement conçu pour respecter la manière dont les différents utilisateurs travaillent avec les informations. L'équipe de conception travaille en étroite collaboration avec sa communauté d'utilisateurs et est consciente que certains utilisateurs travaillent mieux avec la visualisation schématique et d'autres avec du texte. De nombreux outils disponibles dans Enterprise Architect ont été conçus en tenant compte de ces différents types d'utilisateurs. Par exemple, dans la discipline Gestion des Exigences, les utilisateurs sont souvent plus habitués à travailler dans des documents et des feuilles de calcul. Pour répondre à ce besoin, Enterprise Architect propose un certain nombre de vues vers lesquelles les utilisateurs peuvent basculer, qui leur permettent de saisir, de modifier et de gérer Exigences via ces types d'interface. L'un de ces outils est le Gestionnaire de Spécification, qui offre une manière flexible et familière de travailler avec Exigences à la fois à partir d'une vue basée sur des documents et d'une vue sur des feuilles de calcul, permettant de visualiser, de créer et de gérer Exigences en toute simplicité.

Specification Manager: Package: "HSUV Specification" [ Requirement ]				
SysML Example Model ▶ SysML 1.5 ▶ Modeling Domain ▶ HSUV Model ▶ HSUV Requirements ▶ HSUV Specification				Find Package
Requirement	Stereotype	Status	Difficu...	Priority
<input checked="" type="checkbox"/> <b>Capacity</b>	requirement	Proposed	Medium	Medium
<input checked="" type="checkbox"/> <b>CargoCapacity</b> The system will have a Cargo Capacity: Seats folded down of 2 cubic metres (70 cubic feet) and with the seats folded up 1 cubic metre (36 cubic feet). Cargo capacity sometimes also referred to as Cargo volume is the total volume (measured in cubic metres or feet) of space in a car's cargo area. In SUVs, minivans and hatchbacks, there are two operational contexts that must be considered requiring two distinct values: Cargo Capacity Seats Up and Cargo Capacity seats folded down.	requirement	Proposed	Medium	Medium
<input checked="" type="checkbox"/> <b>FuelCapacity</b> The system will have two separate fuel tanks one will be the main fuel storage and will have a capacity of 80 liters (approx. 21.1 US Gallons) and a reserve fuel tank with a storage of 10 Litres (approx. 2.6 US Gallons). Both tanks will be fitted with gauges capable of reporting fuel levels to a 3% accuracy and the volumes value in terms of percentage of tank and number of kilometres (or miles) will be capable of being displayed on the dashboard. Fuel (Tank) Capacity is the volume of fuel that can be stored in the fuel tank system of the Hybrid SUV.	requirement	Proposed	Medium	Medium
<input checked="" type="checkbox"/> <b>PassengerCapacity</b>	requirement	Proposed	Medium	Medium

# Modèles, Diagrammes , éléments et Vues

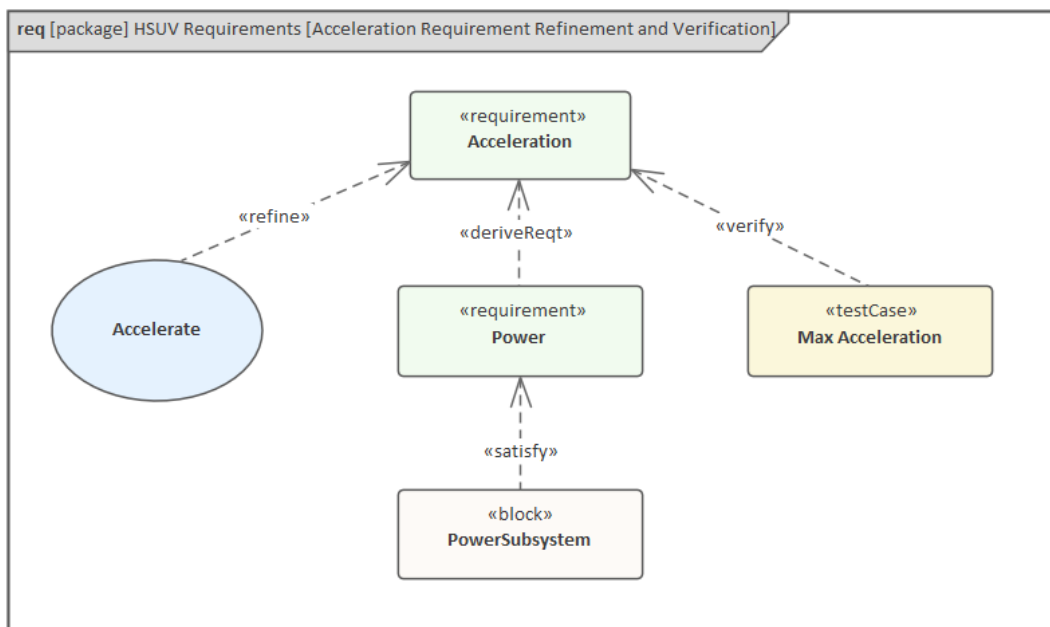
## Modèles

Le terme modèle est largement galvaudé et chargé de sens. Certaines personnes utilisent le terme modèle pour désigner un référentiel entier, tandis que d'autres l'utilisent pour désigner une section d'un référentiel complet. Les modèles sont les divisions structurelles du référentiel.

## Diagrammes

La spécification SysML définit neuf types de diagramme. Il s'agit de la liste canonique et SysML définit en plus les éléments qui sont généralement utilisés sur chaque diagramme. De nombreux nouveaux venus et même certains utilisateurs expérimentés ne savent pas que même si ces listes d'éléments décrivent les éléments couramment utilisés pour un type de diagramme particulier, cela n'empêche pas un modélisateur d'utiliser d'autres éléments sur ces diagrammes. En fait, l'utilisation d'un certain nombre de types d'éléments sur le même diagramme produit un modèle expressif et permet aux parties prenantes et aux ingénieurs de différentes disciplines de comprendre les connexions inter-diagrammes entre les modèles.

Dans cette section, nous apprendrons également qu'il existe un certain nombre d'éléments de modèle « universels » qui, selon la spécification, peuvent être incorporés dans n'importe quel diagramme, notamment les commentaires, les contraintes et les justifications. Ce diagramme montre une gamme de types d'éléments, notamment un Bloc, un cas d'utilisation, une exigence et un cas Test, tous exprimés sur une Interrompre lorsqu'une Variable Change de Valeur (BDD).

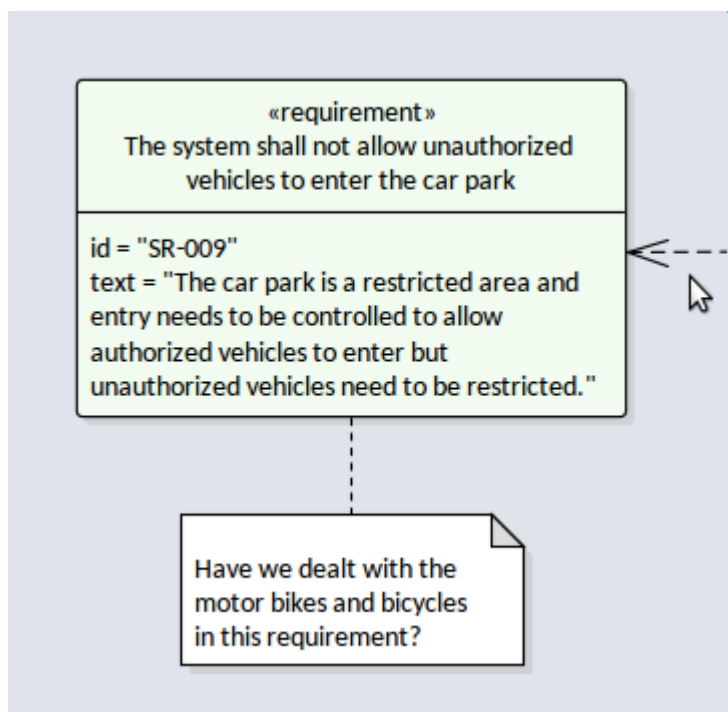


Comme indiqué précédemment, le Langage Modélisation des Systèmes spécifie neuf types différents de diagramme.

## Éléments

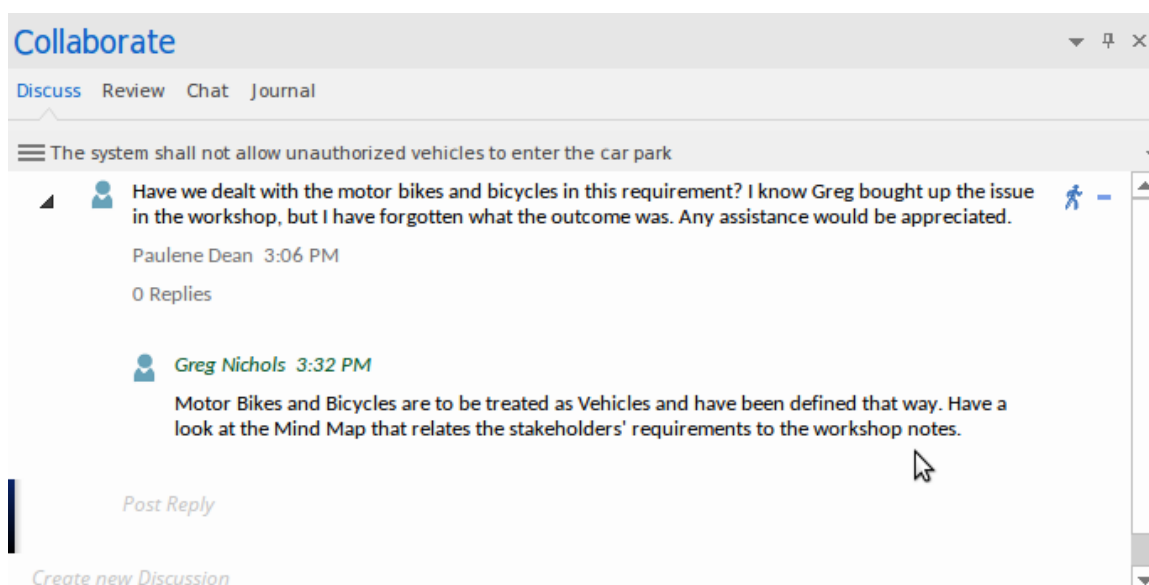
Les neuf diagrammes décrits précédemment sont spécifiés individuellement pour transmettre un aspect particulier d'une

opportunité ou d'une solution d'ingénierie. Par exemple, le diagramme Paramétriques est destiné à montrer comment les équations sont construites. Il existe cependant un certain nombre de types d'éléments qui sont universels pour l'effort de modélisation et peuvent apparaître sur n'importe quel type de diagramme . Bon nombre de ces éléments sont utilisés sur diagrammes pour transmettre des annotations importantes à un modèle ou pour aider à expliquer un aspect particulier du modèle. Ils comprennent des éléments tels que Notes , Contraintes, Rationales et Vues . Dans ce diagramme une partie prenante qui a visualisé le modèle a ajouté un commentaire pour remettre en question une partie du modèle.



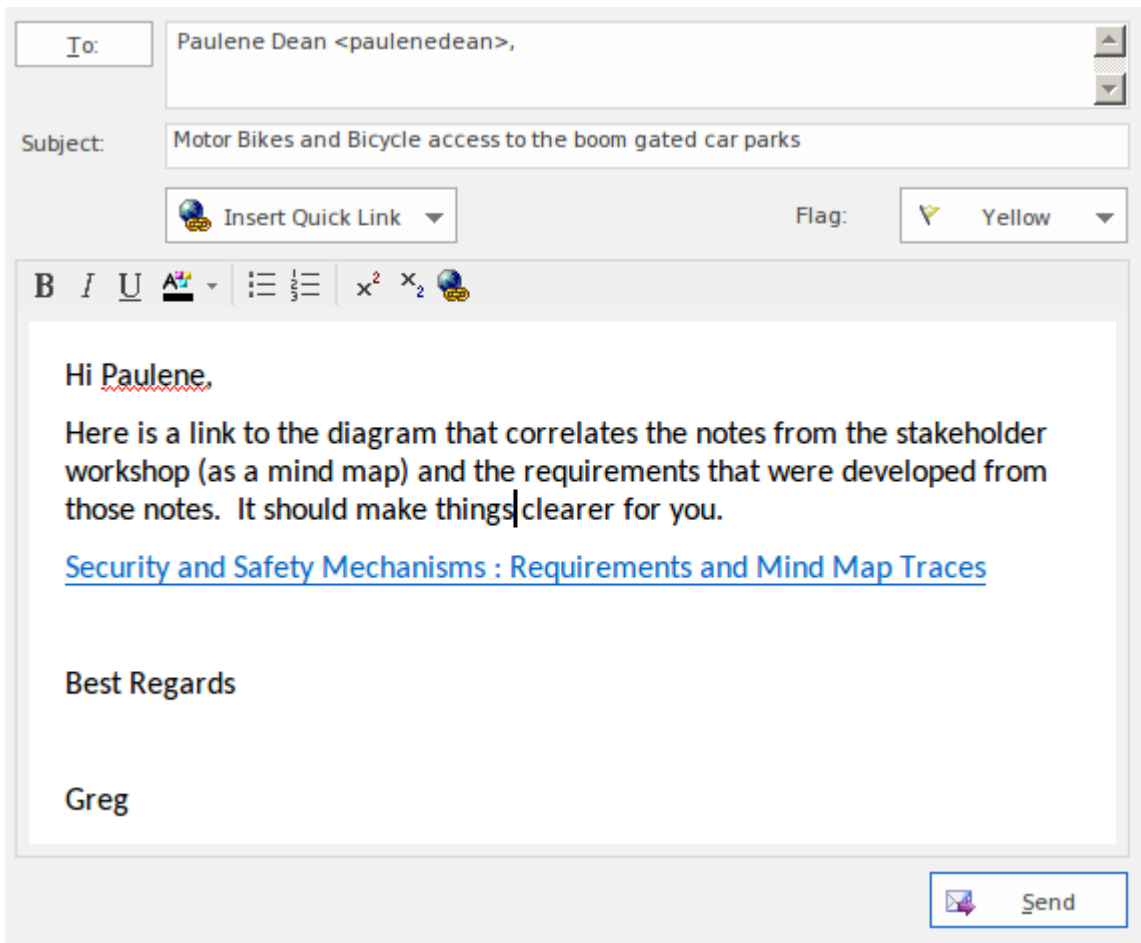
Pour plus d'informations, consultez la rubrique d'aide [Common Pages](#) .

Bien qu'Enterprise Architect soit hautement conforme à la spécification SysML, il dispose d'un certain nombre de fonctionnalités collaboratives qui permettent de gérer de tels commentaires, notamment via sa fonctionnalité Discussion. Cela permet de séparer les discussions des éléments propres qui composent le modèle. Cette image d'écran montre le même commentaire ajouté à l'aide de la fonctionnalité Discussion, qui permet des réponses et une gamme d'autres dispositifs collaboratifs.



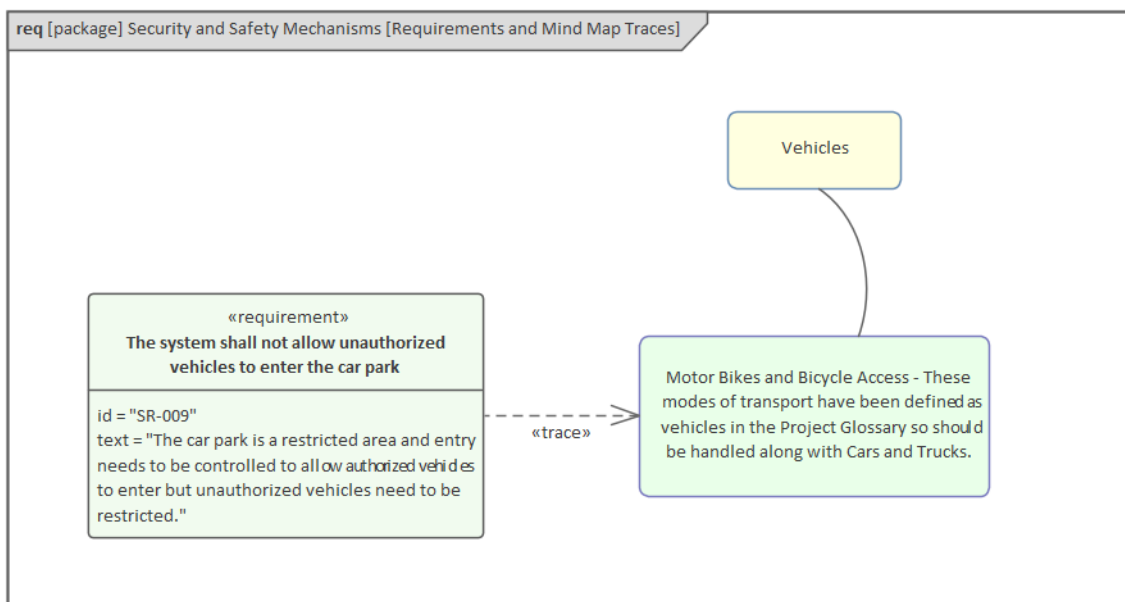
Pour plus d'informations, consultez la rubrique d'aide [Discussions](#) .

Le répondant aurait également pu envoyer un message Mail de Modèle , qui pourrait inclure un lien vers un élément ou diagramme Enterprise Architect comme indiqué.



Pour plus d'informations, consultez la rubrique d'aide [Model Mail](#) .

Lorsque l'utilisateur ouvre le message électronique, il peut suivre le lien et ouvrir le diagramme référencé dans le lien. Il s'agit d'un mécanisme utile permettant de référencer et d'accéder à des vues dynamiques et en temps réel des informations modélisation plutôt qu'à une image envoyée dans un document statique. Cette image montre le diagramme qui s'afficherait dans l'outil lorsque le lien est ouvert.



La capacité à relier des éléments de différents domaines modélisation est l'un des atouts majeurs d' Enterprise Architect en tant que plate-forme unificatrice pour les équipes et les disciplines, et cela est particulièrement vrai dans la relation

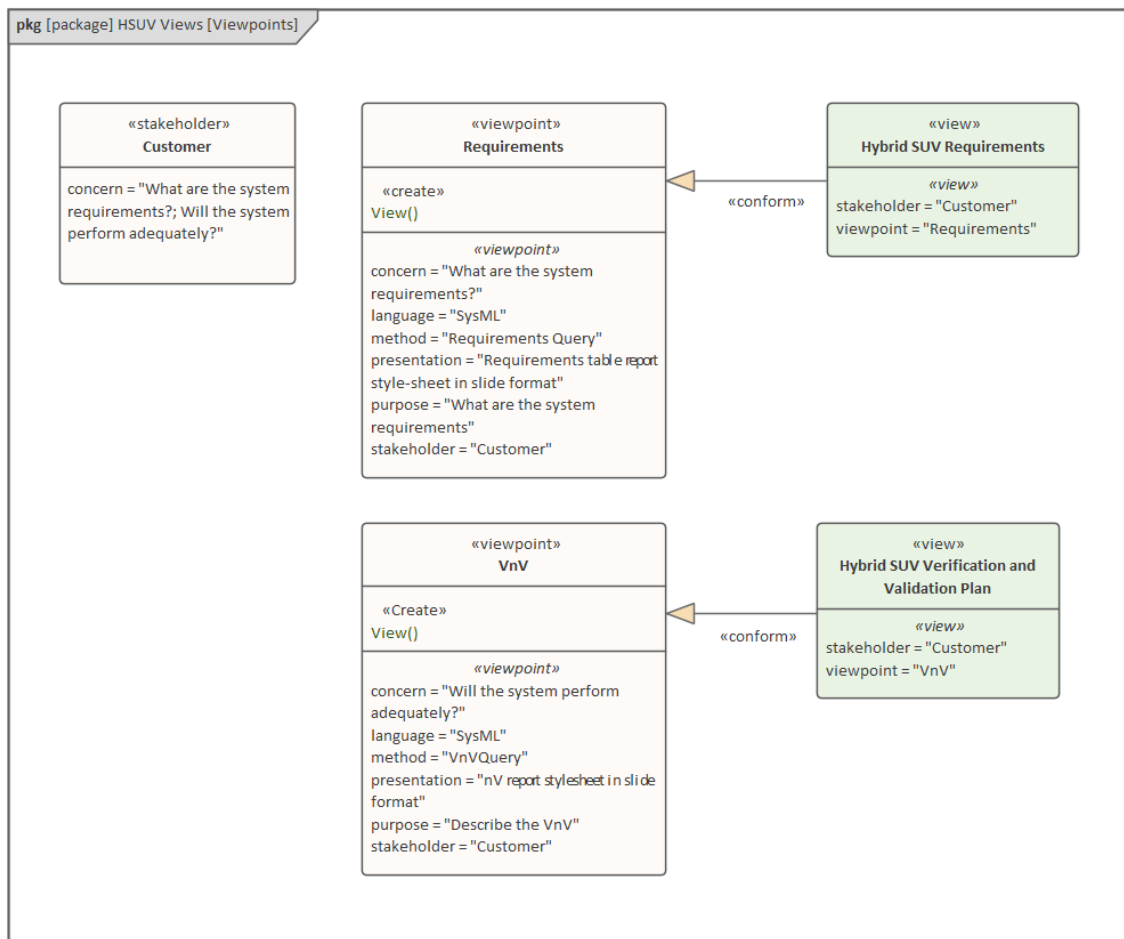
entre la stratégie d'entreprise et l'ingénierie, et entre le développement logiciel et l'ingénierie. Le résultat est un modèle cohérent et harmonisé dans lequel la possibilité d'erreurs résultant de jonctions entre différentes équipes est considérablement réduite. Pour plus d'informations, consultez la rubrique d'aide [Traceability Window](#).

## Vues

Fondamentalement, un système est conçu, analysé, conçu et construit pour ses parties prenantes. Les ingénieurs systèmes recueillent les préoccupations et les intérêts des parties prenantes et appliquent l'analyse pour créer des exigences et des contraintes. Celles-ci sont utilisées comme données d'entrée pour l'analyse et la conception et avant que le système ne soit livré pour validation et vérification. Les parties prenantes doivent être en mesure de visualiser la manière dont leurs intérêts sont pris en compte à différentes étapes du processus d'ingénierie et cette visualisation peut être fournie par des vues et des points de vue. Les concepts de point de vue et de vue sont articulés dans la norme ISO-42010 (anciennement IEEE-1471) et la spécification SysML a été rédigée pour être cohérente avec la norme ISO-42010. Il existe un certain nombre de points de vue couramment utilisés, notamment :

- Opérationnel
- Performance
- Fabrication
- Sécurité

Un point de vue est une prescription permettant de construire une vision qui répondra aux besoins, aux intérêts et aux préoccupations d'une partie prenante donnée. Une vision est ce que la partie prenante voit d'un point de vue donné et doit lui permettre de visualiser les parties du système qui la concernent tout en laissant de côté ou en occultant les aspects du système qui ne l'intéressent pas.



En plus du mécanisme formel décrit par la spécification SysML sous la forme d'éléments Vue et Viewpoint, Enterprise



Architect dispose d'une large gamme d'outils pour aider à la création et à la gestion des points de vue, des vues et des représentations. Plusieurs outils peuvent être utilisés pour créer différentes vues des éléments du référentiel, notamment les Ensembles de travail et les Modèle Vues . Ensembles de travail permettent d'enregistrer et de rouvrir sous forme d'ensemble une collection de diagrammes , de matrices, de bibliothèques Modèle et d'autres éléments, ce qui est utile lorsque vous travaillez avec différents groupes de parties prenantes. Pour plus d'informations, consultez la rubrique d'aide [Working Sets](#)

Modèle Vues permet de créer des vues d'éléments regroupés indépendamment de leur emplacement dans la fenêtre Navigateur . Il existe également plusieurs outils permettant de masquer ou d'obscurcir des parties d'un diagramme afin de le rendre plus attrayant pour un public particulier. L'apparence des diagrammes peut être modifiée en modifiant l'apparence des éléments, notamment en utilisant une image, et Filtres de Diagramme peuvent masquer ou cacher des éléments de la vue. Pour plus d'informations, consultez la rubrique d'aide relative [Visual Filters](#) . Le moteur de documentation peut créer des publications de haute qualité directement à partir du modèle aux formats PDF ou DocX. Pour plus d'informations, consultez la rubrique d'aide [Documentation](#) .

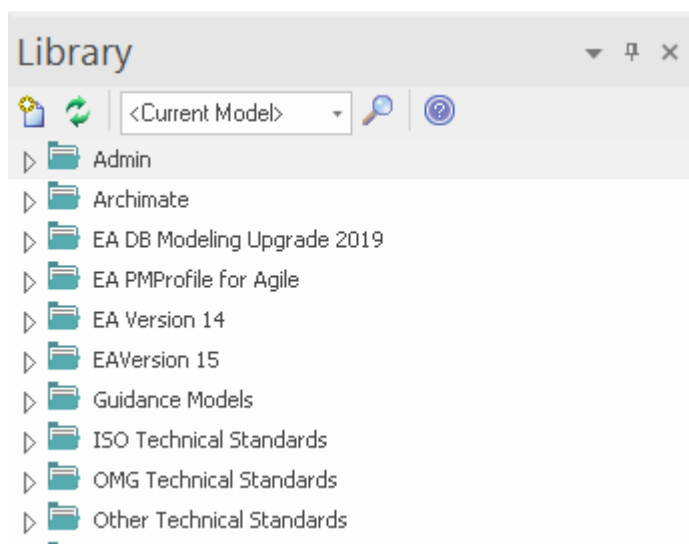
## Collaborer en tant qu'équipe d'ingénierie

Une équipe d'ingénieurs est multidisciplinaire et se compose de stratèges, de gestionnaires, d'ingénieurs système, d'ingénieurs logiciels, de testeurs et d'autres. Les pressions commerciales pour lancer un produit ou fournir une solution signifient que les équipes doivent travailler de manière plus intelligente et plus cohérente pour garantir les résultats d'ingénierie. Enterprise Architect a été conçu dès le départ comme une plate-forme collaborative, non seulement pour les ingénieurs mais pour toutes les disciplines. Il facilite la collaboration des individus et des équipes et le partage d'informations, de modèles, de conceptions et de solutions avec une gamme complète d'outils allant des discussions, révisions, d'une bibliothèque d'équipe et d'un chat au Contrôle de Version et Lignes de base.

## Référentiel partagé central

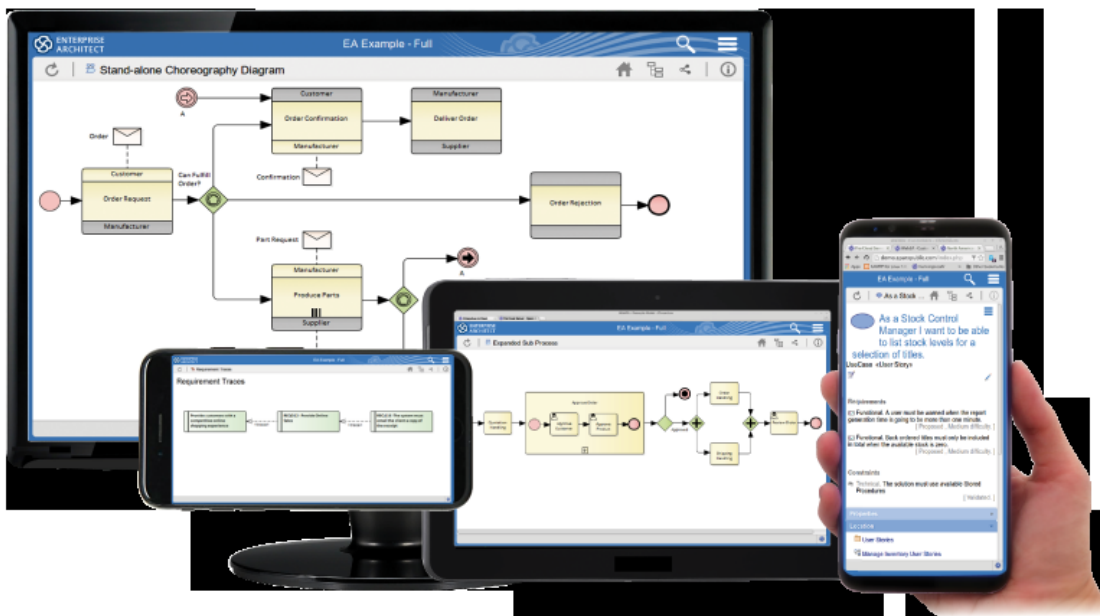
Enterprise Architect n'est pas seulement un outil de dessin sophistiqué, mais une plate-forme modélisation qui permet de valider, de simuler, de transformer et de manipuler des modèles par programmation. Cela est possible car l'outil ne stocke pas les diagrammes au format raster ou vectoriel, mais plutôt les images sont codées dans une base de données relationnelle avec toutes les données de référence et autres métadonnées du référentiel. La plate-forme utilise cette base de données relationnelle pour stocker toutes les informations et métadonnées modélisation et cela fournit le stockage back-end auquel accèdent les outils client et Web. Pour plus d'informations, consultez le Sujet d'aide [The Model Repository](#) .

Cette architecture de produit permet aux utilisateurs de partager des modèles, diagrammes et d'autres informations du référentiel telles que des données de référence, des images dans la Bibliothèque d'images et des documents dans la Bibliothèque d'Équipe et de nombreuses autres fonctionnalités d'outils qui facilitent la collaboration et le travail en équipe au sein d'une équipe colocalisée ou distribuée. Pour plus d'informations, consultez la rubrique d'aide [The Modeling Team](#) .



# Cloud Computing

Le référentiel central décrit dans la rubrique précédente est accessible directement depuis le client de bureau Enterprise Architect, en utilisant les couches de connectivité de base de données appropriées (ODBC) et les pilotes associés. Il existe cependant un autre moyen plus simple d'accéder à un référentiel hébergé dans un Cloud, via les services Cloud utilisant les protocoles http ou https. Le Cloud peut être sur site ou hors site et il existe de nombreuses façons de Cloud configurer. Il existe également un serveur comprenant un certain nombre de modules pouvant être achetés, appelé Pro Cloud Server, qui fournit une variété d'outils, facilités et de moyens de visualiser les modèles via un navigateur sur un ordinateur ou un appareil portable, tel qu'une tablette ou un smartphone. Pour plus d'informations, consultez la rubrique [Pro Cloud Server Repositories](#).

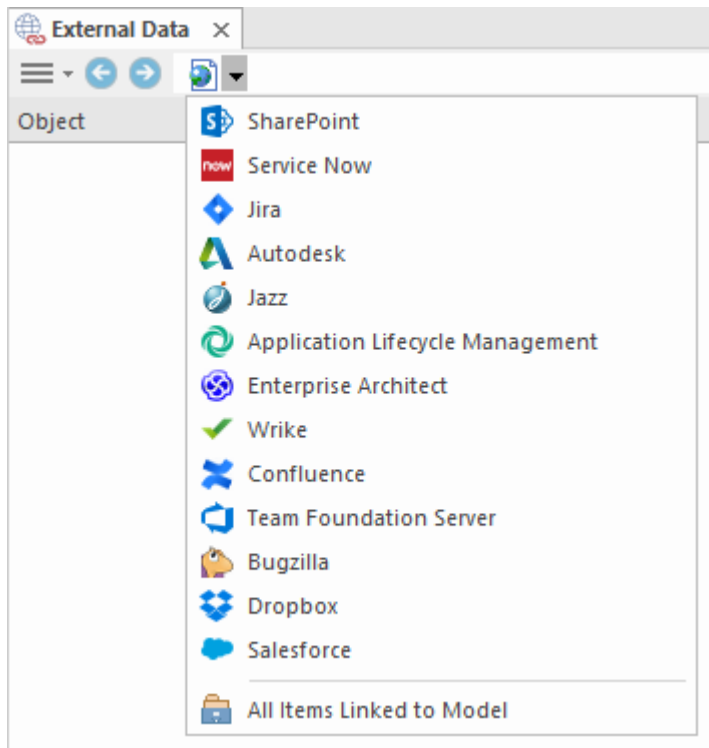


Cette plateforme permet de travailler avec une grande variété d'intervenants, des responsables d'ingénierie aux ingénieurs-conseils, qu'ils soient locaux ou répartis dans le monde entier. La puissance de cette méthode de travail peut être exploitée avec une configuration minimale et l'avantage de permettre à tous les membres de l'équipe et aux intervenants de visualiser et de contribuer aux mêmes modèles ne peut être sous-estimé.

Des gains de productivité peuvent être obtenus en étant capable d'obtenir des commentaires précieux et opportuns, ou de discuter d'un diagramme avec un consultant en ingénierie alors qu'il voyage en train vers un salon professionnel, ou avec le responsable de l'ingénierie pendant une pause d'un symposium - le tout en temps réel et dans le modèle à partir d'un smartphone ou d'une tablette.

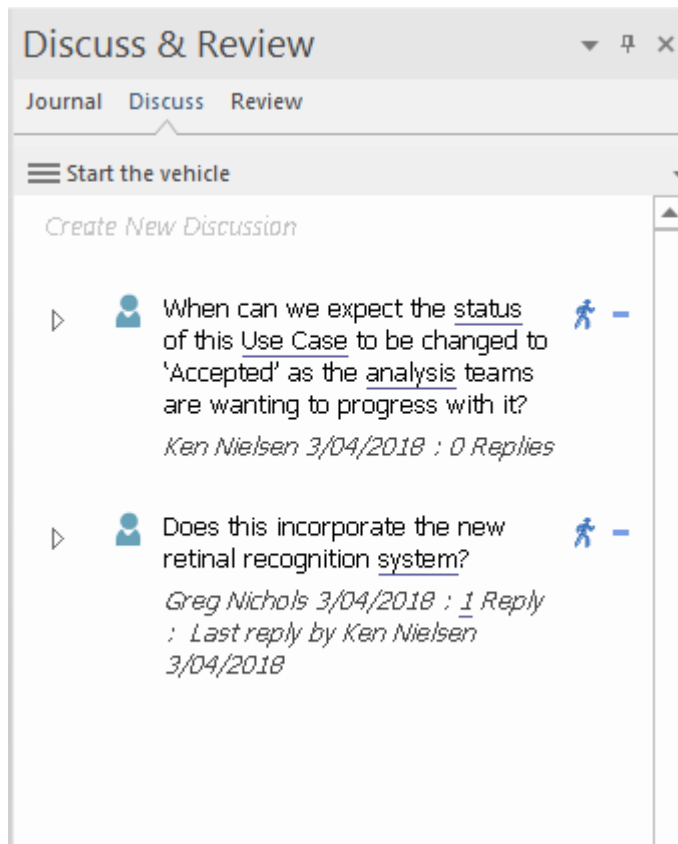
Le Pro Cloud Server offre également une connectivité à une large gamme d'autres outils et plates-formes, couvrant les systèmes Gestion des Exigences, les outils de gestion de configuration, les systèmes de suivi des problèmes, les systèmes Gestion de Projet et plus encore.

Cette capture d'écran montre une liste de certaines des intégrations disponibles. Cette liste est régulièrement complétée. Il est donc utile de vérifier si des intégrations supplémentaires ont été incluses. Pour plus d'informations, consultez la rubrique [Integrate Data from External Providers](#).



## Discussions et chat

Au cœur de la notion de collaboration se trouve la capacité d'un modélisateur à discuter et à discuter avec des collègues ou des spécialistes de l'industrie et des normes à propos d'un problème ou d'une solution. Enterprise Architect permet aux ingénieurs, aux responsables et à d'autres personnes de participer à des discussions sur les éléments, diagrammes et les connecteurs. Un message peut être créé pour lancer un fil de discussion ou une conversation dans laquelle d'autres modélisateurs peuvent ensuite participer en répondant. Les discussions sont conservées séparément des méta-informations sur les éléments et diagramme ce qui permet de faire des commentaires riches et constructifs sans affecter la documentation ou les rapports générés à partir des modèles. Les discussions et le chat sont deux des options disponibles, les discussions de la fenêtre Discuss & Révision et les chats de la fenêtre Chat & Mail.

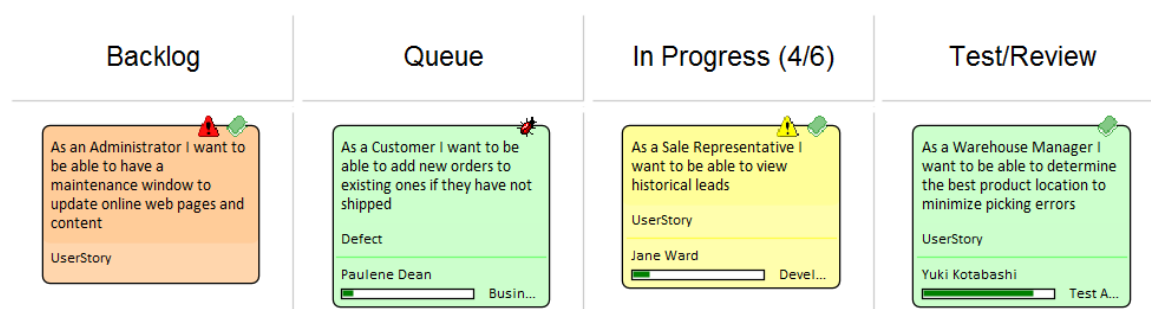


Le chat est utile pour une communication rapide et réactive avec des collègues ou des experts qui ont été définis comme faisant partie d'un groupe d'utilisateurs défini par la sécurité. Les chats ne sont pas liés aux éléments du modèle comme le sont les discussions, mais sont plutôt globaux et lorsque la fenêtre Chat & Mail est ouverte et qu'un groupe est sélectionné, les éléments sont répertoriés par ordre de date et d'heure. Pour plus d'informations, consultez la rubrique d'aide [Teams & Collaboration](#).

## Ressources et calendriers Kanban

La technique Kanban a été implémentée dans Enterprise Architect de manière à améliorer considérablement la productivité de votre équipe et la gestion de projets ou d'ensembles de tâches d'ingénierie logicielle et système. Dans Enterprise Architect il s'agit d'une fonctionnalité simple à utiliser, qui vous permet de gérer les éléments d'un backlog et de les déplacer vers un nombre quelconque de voies, voire vers d'autres tableaux, représentant les étapes d'un processus. La facilité peut être intégrée dans des processus d'ingénierie ou de développement logiciel existants ou nouveaux, ce qui se traduit par des gains d'efficacité sans précédent.

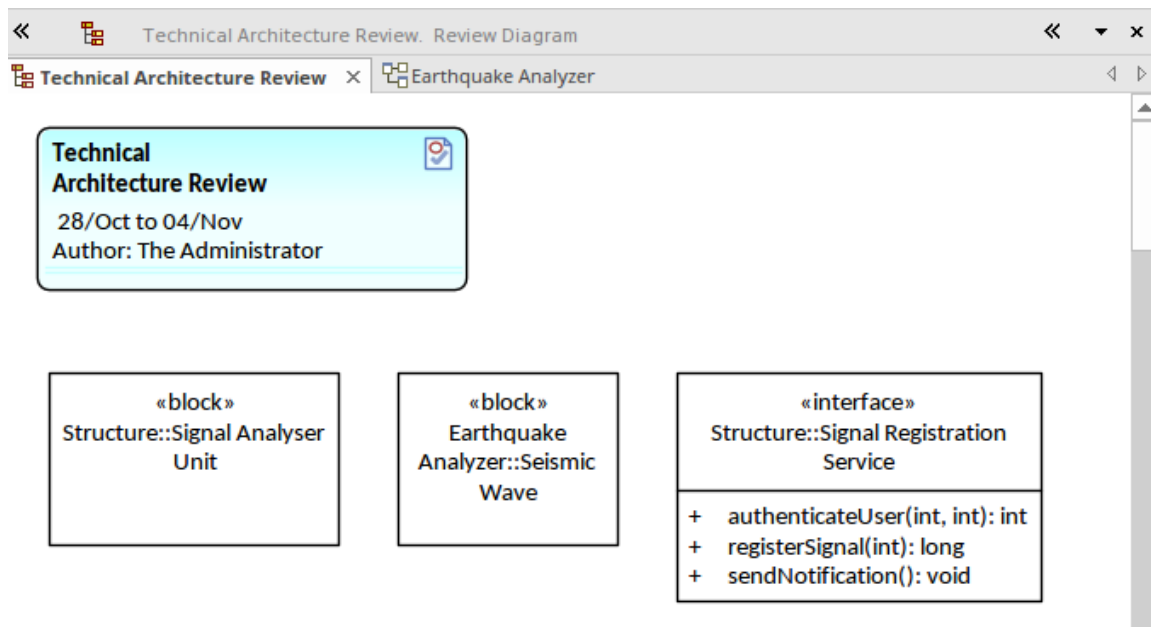
L'un des grands avantages de cette fonctionnalité est que les éléments qui participent aux diagrammes Kanban peuvent être liés à d'autres éléments du référentiel, ce qui permet une traçabilité complète depuis, par exemple, une exigence jusqu'à une intention stratégique ou un composant d'une conception jusqu'à un élément d'un produit commercialisé. Kanban peut être utilisé pour visualiser les ressources, l'effort et l'état d'avancement des éléments à mesure qu'ils se déplacent dans un nombre quelconque de tableaux Kanban liés. Il s'agit d'un exemple de processus de développement logiciel lié à un projet d'ingénierie de systèmes d'entrepôt.



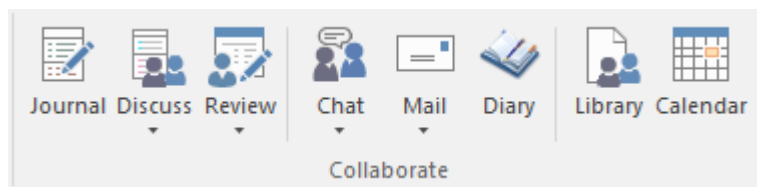
Pour plus d'informations, consultez les rubriques d'aide [Kanban Boards](#) et [Resources & Work Items](#).

## Révision de Modèles

Révision de Modèles permet aux parties prenantes du projet de collaborer formellement à l'évaluation ( révision ) du contenu du modèle, y compris les éléments et diagrammes . Cet outil de collaboration pratique utilise un certain nombre de fonctionnalités intégrées - telles que la vue Révision - pour gérer le processus de révision et pour visualiser les discussions et les contributions à la révision . Un diagramme révision fournit un mécanisme permettant aux participants d'ajouter des éléments et diagrammes à la révision . Ce diagramme contient un certain nombre d'éléments liés au sujet de la révision .



Un nombre illimité de révisions peuvent être créées et les modélisateurs peuvent rejoindre et participer aux révisions . La rampe de lancement du facilité révision est le panneau « Collaborer » du ruban Démarrer .

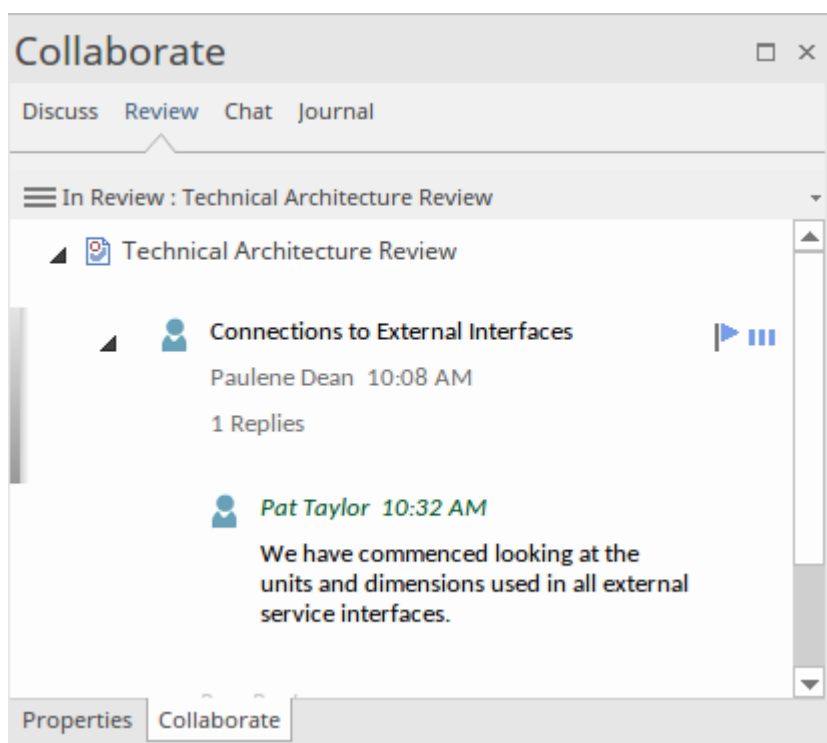


La sélection de l'option « Révision > Gérer Révisions » ouvrira la vue Révisions , où toutes révisions existantes seront répertoriées et - avec un clic droit de la souris - de nouvelles révisions pourront être créées. La vue affiche les métadonnées Révision dans le panneau de gauche (y compris les dates de début et de fin) et les détails révision dans le panneau de droite, y compris l'élément et diagrammes qui font partie de la révision et les messages et réponses pour chaque élément.

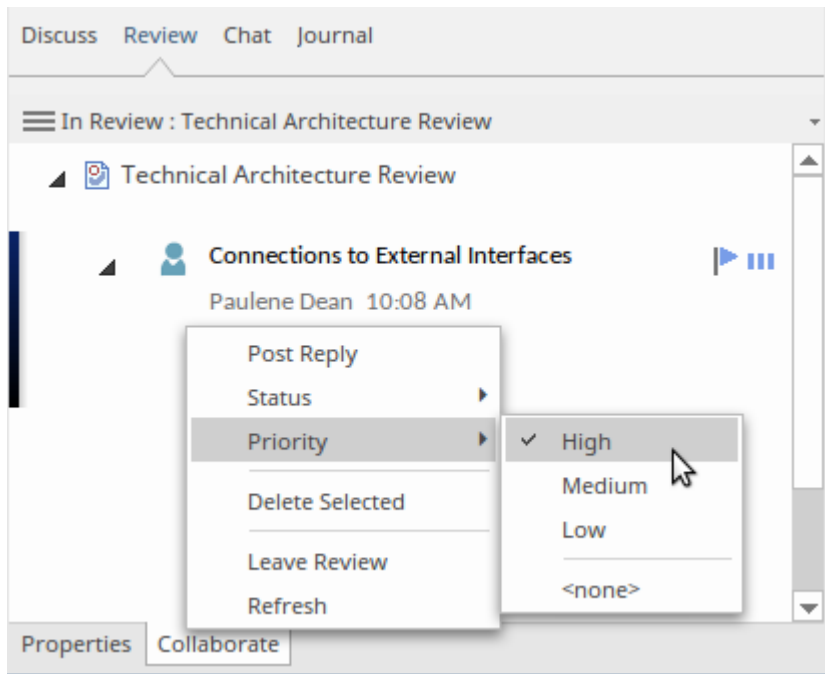


Element	Topics Open	Complete	Posts (Recent)
<ul style="list-style-type: none"> <li>▲  <b>Technical Architecture Review</b> Approved. 28 Oct to 04 Nov</li> <li> <ul style="list-style-type: none"> <li>▲  Set : Technical Architecture Review                             <ul style="list-style-type: none"> <li> Seismic Wave</li> <li> Signal Analyser Unit <span style="float: right;">1</span> <span style="float: right;">1</span></li> <li> Signal Registration Service <span style="float: right;">1</span> <span style="float: right;">1</span></li> </ul> </li> <li>▲  Other Items                             <ul style="list-style-type: none"> <li> Technical Architecture Review <span style="float: right;">1</span> <span style="float: right;">2</span></li> </ul> </li> </ul> </li> </ul>			

Les commentaires révision peuvent être saisis et visualisés dans l'onglet « Révision » de la fenêtre Discussion et Révision . Ils conserveront un décompte continu de tous les messages et réponses annotant le message, ainsi que des réponses avec l'auteur et la date. De cette manière, des résultats hautement collaboratifs peuvent être obtenus et - grâce au produit WebEA disponible sur les smartphones et les tablettes - les parties prenantes telles que les consultants externes et les experts du secteur peuvent contribuer à la révision sans avoir à utiliser l'application de bureau Enterprise Architect .



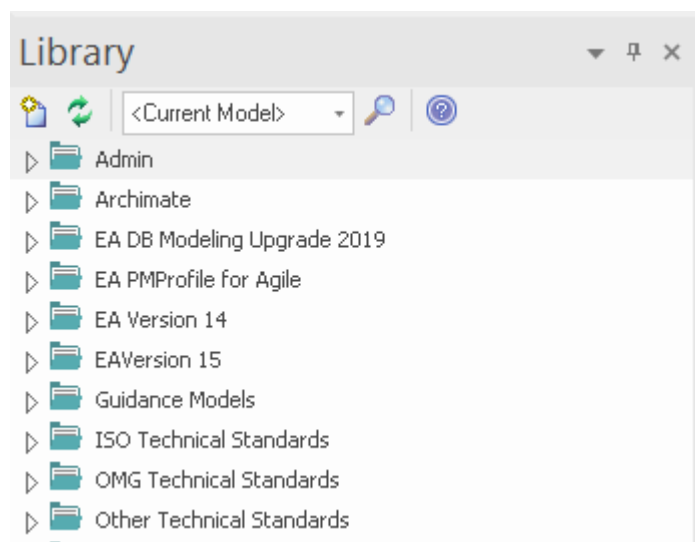
Chacun des éléments de la vue Révision peut avoir un certain nombre de propriétés définies, y compris le statut et la priorité de l'élément révision ; ceux-ci peuvent être vus comme deux petites icônes (un drapeau et une icône de quantité) à droite de l'élément.



## Partager des ressources dans la Bibliothèque d'Équipe

Les équipes qui utilisent des processus centrés sur Ingénierie Systèmes Modèles Basée s'appuieront invariablement sur une vaste gamme de documents sous forme de politiques, méthodes, instructions, descriptions de processus, documents guidage , normes et autres types de documentation d'ingénierie ou de projet. Certains d'entre eux seront basés sur des documents et d'autres seront basés sur des ressources et disponibles sur un réseau interne, un système partagé, un Intranet ou un Extranet ou plus largement sur l'Internet public.

Quel que soit l'emplacement des documents ou des pages, ils peuvent être importés dans Enterprise Architect ou référencés en tant que ressources externes via une URL. Ils peuvent être inclus dans un diagramme en tant qu'artefact interne ou externe, mais il est plus pratique de les importer ou de les référencer à partir de la Bibliothèque d'Équipe .



Il pourrait également être envisagé d'importer certains ou tous les éléments de ces documents en tant qu'éléments de modèle de première classe. Par exemple, une politique pourrait être créée en tant qu'élément de métamodèle et la liste des politiques pourrait être importée, ce qui permettrait de relier les politiques individuelles à des composants système particuliers. Pour plus d'informations, consultez la [The Model Library](#) Rubrique d'aide.

## Affichage des modèles sur les appareils mobiles

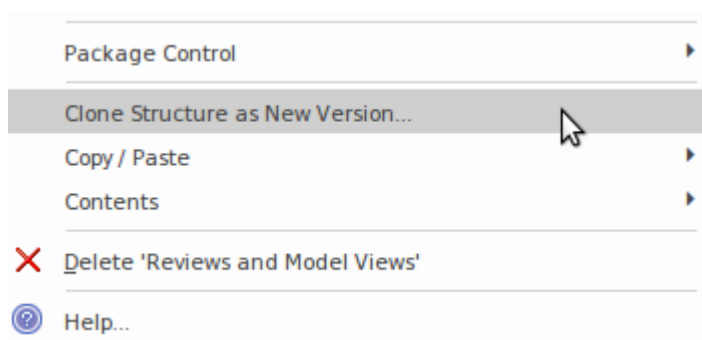
Le monde du travail moderne a beaucoup changé ces dernières années, les organisations encourageant la flexibilité sous forme de bureaux partagés et de travail à domicile, ce qui conduit de plus en plus de personnes à travailler sur des appareils portables. En outre, le rythme du changement dans notre monde moderne s'accélère chaque année, poussé par l'innovation et les perturbations - par exemple, une pandémie touchant tous les pays dans lesquels des fournisseurs, des clients ou des collègues sont basés, obligeant les gens à s'absenter des bureaux partout et à ne pas pouvoir se déplacer pour consulter ou fournir une expertise directement. Les stratèges, les technologues et les ingénieurs doivent collaborer pour obtenir des résultats techniques et, dans une main-d'œuvre dispersée, cela signifie généralement qu'ils doivent contribuer aux modèles à partir d'appareils mobiles, à la fois en déplacement et dans des conditions d'isolement restreint.

Le contenu du référentiel Enterprise Architect peut être visualisé en temps réel via un navigateur sur un appareil mobile tel qu'une tablette ou un smartphone. Cela permet aux ingénieurs, aux responsables et à d'autres personnes de collaborer entre deux réunions, lors d'inspections hors site, dans les transports en commun ou partout où ils se trouvent. Cela n'a jamais été possible auparavant, et maintenant la vitesse d'un projet n'a pas besoin d'être ralentie en attendant révisions ou le retour des personnes au bureau ; les modèles peuvent être mis à jour à la vitesse d'Internet depuis n'importe où. Pour plus d'informations, consultez la rubrique d'aide [Pro Cloud Server Repositories](#) .

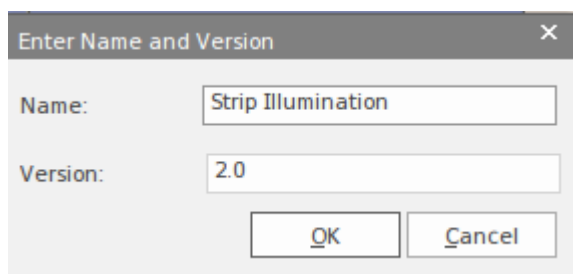
## Modélisation de l' Avenir

Le monde est animé par un appétit presque insatiable pour le changement et l'innovation, ce qui a pour conséquence que les ingénieurs systèmes doivent travailler plus intelligemment et plus rapidement et trouver des moyens astucieux de résoudre les problèmes. *Modélisation Consciente du Temps* est un facilité modélisation unique qui permet aux ingénieurs et autres parties prenantes de modéliser un nombre quelconque d'états futurs. Cette introduction de l'état dans les modèles fournit un mécanisme permettant de visualiser à quoi pourrait ressembler une solution proposée et permet aux ingénieurs de comparer un certain nombre de solutions proposées. Une solution donnée peut être analysée et raisonnée et les pièges et problèmes potentiels peuvent être identifiés dans les modèles. Révisions et des examens peuvent être effectués par un nombre quelconque de parties prenantes et ceux-ci peuvent être utilisés pour déterminer laquelle des nombreuses solutions est appropriée.

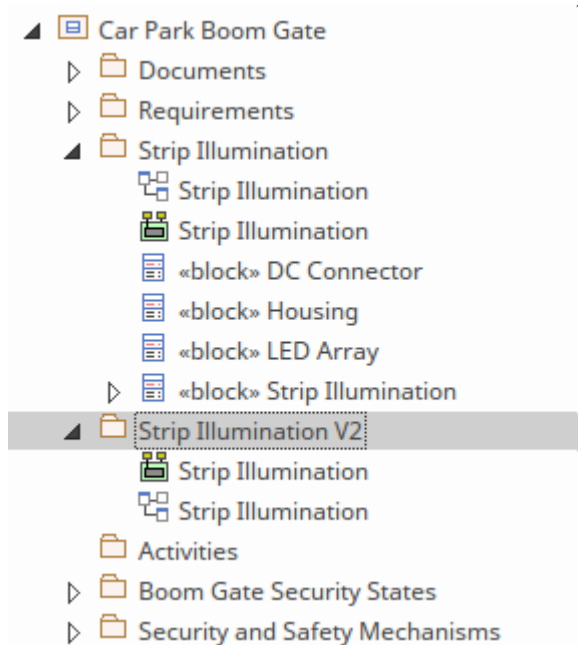
Le processus peut être lancé en créant un clone d'un Paquetage pour lequel vous avez besoin d'un modèle d'état futur. Cela peut être fait en utilisant les options du ruban ou depuis le menu contextuel de la fenêtre Navigateur .



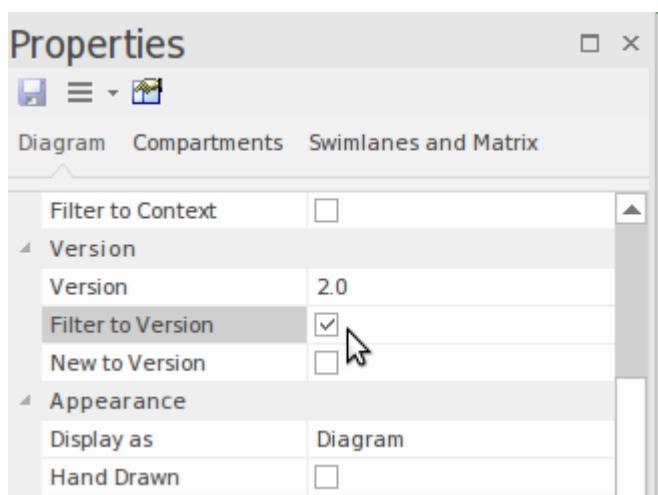
L'outil affiche une prompt permettant à l'ingénieur de spécifier un *nom* et un *numéro de version* pour la structure clonée. Le *numéro de version* est essentiel au fonctionnement de la fonctionnalité et est utilisé par Enterprise Architect comme moyen de suivi des éléments et diagrammes qui font partie de cette future version avec le numéro de version spécifié. L'outil prompt également un emplacement dans le référentiel pour la structure Paquetage clonée. En général, on peut lui donner un nom qui inclut sa version et le parent contenu dans le Paquetage en cours de clonage ; il peut également, cependant, être conservé dans une partie distincte du référentiel destinée aux versions d'état futures.



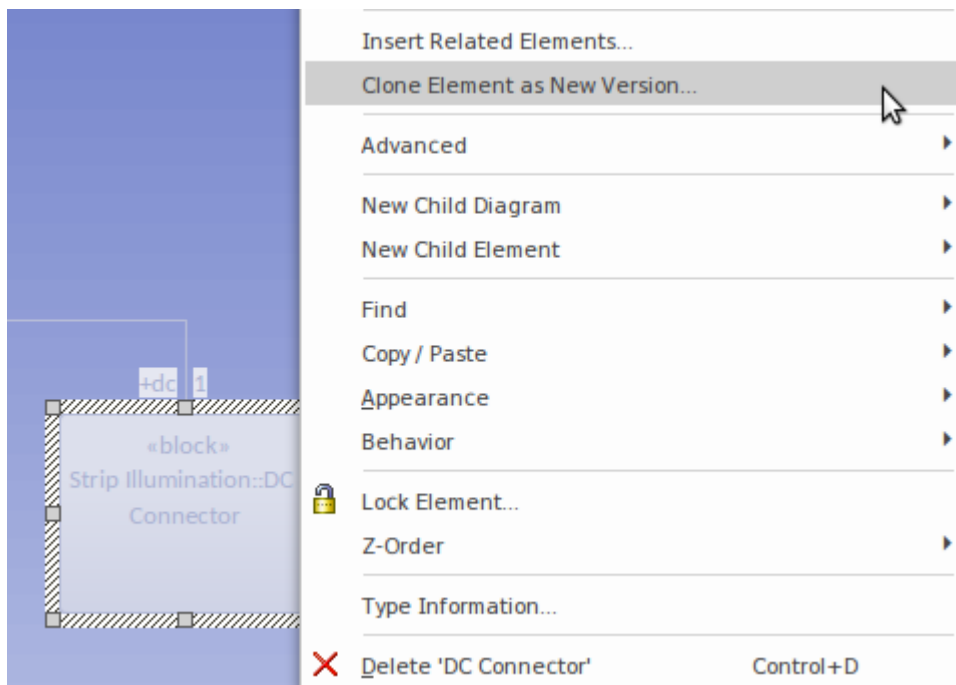
L'étape qui vient d'être réalisée configure simplement la structure pour le clonage et ne crée pas elle-même les états futurs de l'élément ; elle fait cependant une copie de tous diagrammes contenus dans le Paquetage . Cette illustration montre un Paquetage qui a été cloné, contenant deux diagrammes qui sont des copies des diagrammes du Paquetage d'origine.



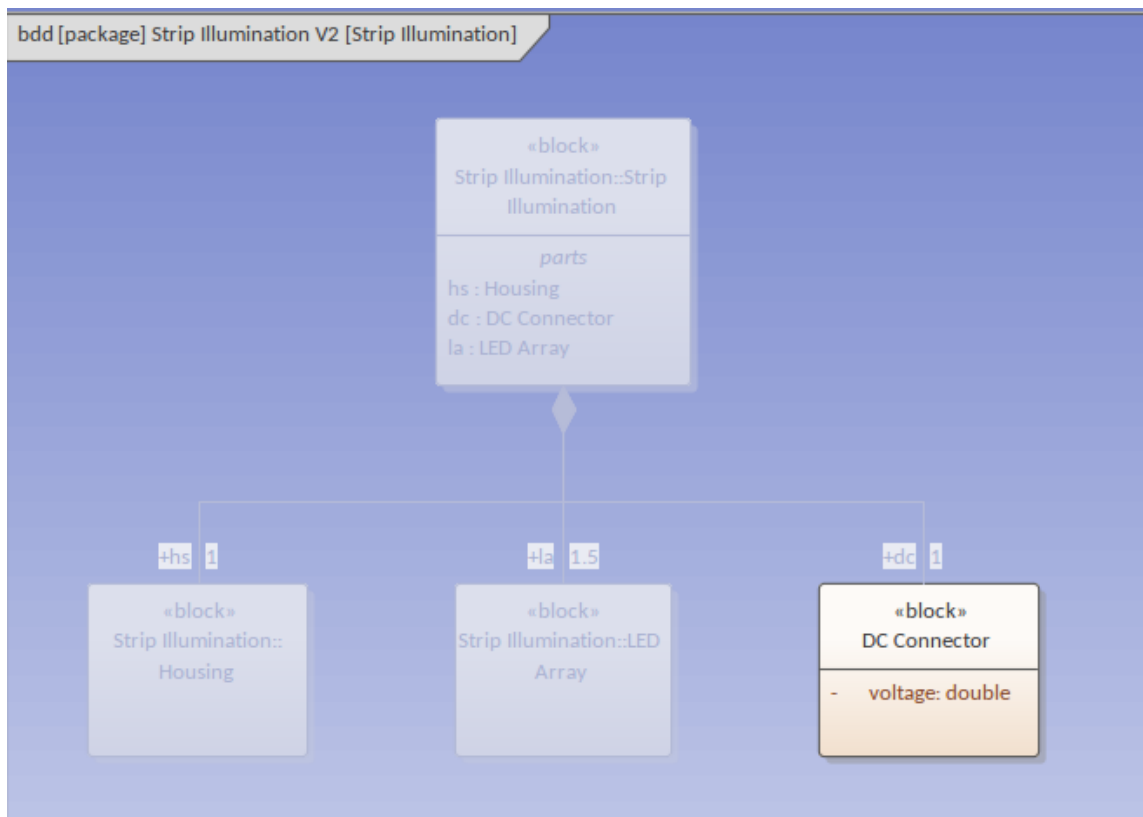
Le diagramme est affecté du numéro de version spécifié dans la dialogue de version, défini par l'utilisateur à l'étape précédente. Le clonage d'éléments individuels s'effectue en sélectionnant un élément sur un diagramme, mais tant que cette opération n'est pas effectuée, tous les éléments du diagramme seront de la version précédente. Enterprise Architect dispose d'une facilité disponible dans la feuille de propriétés diagramme qui permet au modélisateur de filtrer les éléments du diagramme en fonction de la version du diagramme.



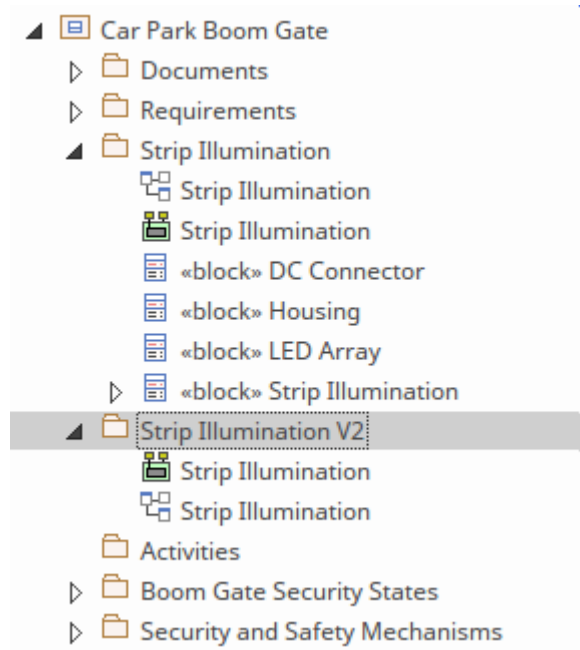
La définition de cette propriété sur un diagramme nouvellement cloné affichera tous les éléments en niveaux de gris, car ils proviennent tous de la version précédente. À partir de ce point, les éléments individuels peuvent être clonés et Enterprise Architect fera une copie de l'élément sélectionné, créant des copies de tous ses connecteurs. Ceci est important car cela permettra à l'élément d'être promu comme version actuelle mise à jour une fois la modification implémentée. Les éléments individuels diagramme peuvent être clonés en sélectionnant l'élément dans le diagramme et en affichant le menu contextuel, comme indiqué dans cette illustration.



Une fois encore, vous serez invité à entrer le numéro de version. Par commodité, l'outil utilisera par défaut celui choisi pour le Paquetage cloné. Une fois ce numéro accepté, Enterprise Architect créera une copie de l'élément sélectionné et, comme le diagramme est toujours *filtré par version*, cet élément apparaîtra normalement dans le diagramme, les autres éléments étant toujours affichés en niveaux de gris.



Le Navigateur affichera l'élément nouvellement créé (version 2) qui sera collationné avec le diagramme. De cette façon, le Paquetage cloné ne contiendra que les éléments et diagrammes portant le nouveau numéro de version. Pour plus d'informations, consultez la [Time Aware Modeling](#) Aide sujet.





# Contrôle de Version et Lignes de base

Nous vivons et travaillons dans un monde qui évolue à la vitesse d'Internet et, par conséquent, les problèmes et les opportunités techniques évoluent à la même vitesse. Le défi des ingénieurs consiste à enregistrer, analyser, évoquer et mettre en œuvre des solutions techniques dans ce laps de temps. Cela signifie que presque avant qu'un ingénieur ait décrit complètement un problème, celui-ci aura changé ou le contexte commercial ou technique du problème aura changé. La plupart du temps, la nouvelle direction ou les changements seront décrits et adoptés, mais d'autres fois, un ingénieur sera obligé de revenir en arrière et de revenir à une version antérieure du problème, de l'opportunité ou de la solution. Enterprise Architect dispose d'outils sophistiqués pour effectuer ce retour en arrière.

Enterprise Architect dispose de deux outils fondamentaux pour travailler avec des versions antérieures du contenu modélisation .

- *Contrôle de Version* - Une fois configuré, tout point de changement peut être retourné ; les utilisateurs extraient les fragments de modèle, apportent des modifications et archivent les fragments, créant ainsi des versions - pour plus d'informations, consultez la rubrique d'aide [Version Control](#)
- *Lignes de base* - Créées à des moments clés ou importants du développement d'un modèle ; un utilisateur crée une ligne de base, puis à un moment ultérieur, le modèle évolué est comparé à la ligne de base à n'importe quel niveau de granularité, et le contenu antérieur peut être restauré - pour plus d'informations, voir la [Baselines](#) Rubrique d'aide.

Il existe une différence importante entre les deux méthodes : Lignes de base doivent être créées intentionnellement. Par exemple, si une équipe d'ingénieurs crée un modèle et qu'il est validé par un responsable produit, puis que le travail se poursuit sur la phase deux, si pour une raison quelconque l'équipe souhaite revenir en arrière et revenir au modèle de la phase un, sans une ligne de base en place, cela ne sera pas possible. En revanche, une fois Contrôle de Version configuré, la même équipe pourrait facilement revenir au jalon de la phase un à condition de connaître la date à laquelle il a été terminé.

## Lignes de base

Lignes de base sont un moyen efficace de garantir qu'une équipe peut revenir en arrière sur un modèle jusqu'à un jalon ou un point important de l'évolution du modèle. Elles offrent un moyen de gestion des changements piloté par l'utilisateur et rassurent les équipes modélisation en leur indiquant qu'elles peuvent facilement revenir à un point antérieur si elles s'écartent de la piste pour une raison ou une autre ou si une dimension du problème, de l'opportunité ou de la solution change et qu'elles doivent le faire. Il est important de se rappeler qu'une ligne de base est simplement un instantané d'un Paquetage (incluant potentiellement des sous-Packages) dans le référentiel, mais elle doit être créée intentionnellement et doit être créée au moment où elle représente. Pour plus d'informations, consultez la rubrique d'aide [Baseline Tool](#) .

Model Elements	Status	Property	Model	Baseline
Robotic Manufacturing		Abstract	false	false
Production Line		Alias		
«block» Automated Assembly Unit	Changed	Author	Theresa Morantini	Yuki Kotabashi
«block» Production Line	Changed	Date	27/03/2018 10:02:01 AM	27/03/2018 10:02:01 AM
Robots in Manufacturing	Changed	Date	7/04/2019 10:28:17 AM	27/03/2018 10:09:09 AM
Visual Elements		Complexity	3	1
«requirement» Automation	Changed	Filename		
«block» Spot Weld Body R...	Changed	Language	Java	C#
«block» Production Line	Changed	IsLeaf	false	false
Production Line	Changed	IsSpec	false	false
Visual Connectors		IsRoot	false	false
Dependency	Changed	Keywords		
Association	Changed	Multiplicity		
Aggregation	Changed	Name	Automated Assembly Unit	Automated Assembly Unit
		Notes		
		Parent	Production Line	Production Line
		Persistence		
		Phase	2.0	1.0
		Scope	Public	Public
		Status	Validated	Proposed
		Stereotype	block	block
		Type	Class	Class
		Version	3.2	1.0

## Contrôle de Version

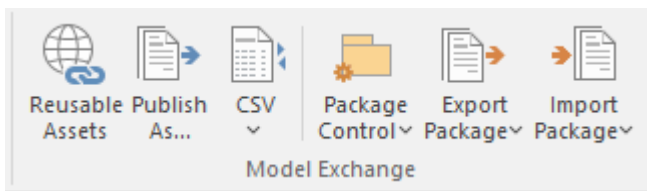
Enterprise Architect permet à une équipe d'ingénieurs de gérer les modifications et les révisions des projets en plaçant Paquetages de modèle individuels, des nœuds d'affichage ou des nœuds racines sous Contrôle de Version . Contrôle de Version est configuré dans Enterprise Architect via un certain nombre d'applications de contrôle de code source tierces qui gèrent l'accès et stockent les révisions des Paquetages contrôlés. Une fois le logiciel Contrôle de Version installé et configuré, une équipe peut enregistrer un historique des modifications apportées aux Paquetages , afficher et récupérer les révisions antérieures du travail, extraire et archiver le contenu au fur et à mesure de son élaboration, etc. Cette facilité permet à une équipe de travailler de manière collaborative tout en offrant aux ingénieurs un moyen isolé de travailler sur des parties particulières du modèle. Pour plus d'informations, consultez la rubrique d'aide [Version Control](#) .

## Serveur d'actifs réutilisables

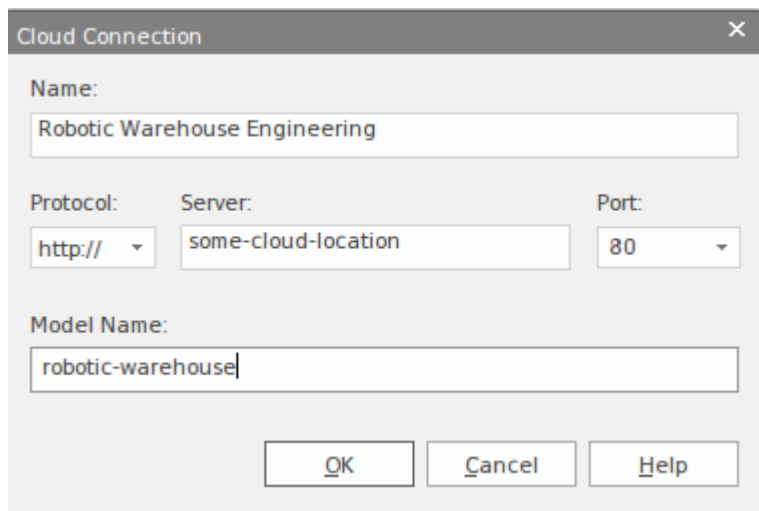
Le serveur d'actifs réutilisables (RAS) est une fonctionnalité de productivité d'équipe qui permet aux équipes de stocker le contenu de modélisation dans un emplacement accessible par des groupes de modélisateurs distribués pour réutilisation. Toute équipe ou organisation peut configurer un RAS, stocker le contenu et, grâce aux paramètres de sécurité, le rendre disponible. L'unité atomique de stockage est un actif qui peut être à la fois des informations basées sur la modélisation et sur des fichiers :

- Paquetages contenus dans un référentiel et visualisables dans le Navigateur
- Fichiers dans une gamme de formats texte, code et graphiques, y compris les fichiers .qea

Le serveur d'actifs réutilisables est disponible à partir du ruban « Publier » à l'aide du panneau « Échange de Modèles ». Cela met ce service à portée de main.



Les ressources sont stockées dans le Cloud et nécessitent une connexion à un modèle de service Cloud Pro configuré à cet effet. En règle générale, cette tâche est effectuée par la section infrastructure d'un service informatique, et les détails de la connexion sont simplement fournis à l'équipe d'ingénierie. Cette capture d'écran montre les détails requis pour établir la connexion basée sur le Cloud.



Une fois qu'un serveur a été configuré, il est possible d'y ajouter n'importe quel élément. Formellement, le serveur est un registre et le contenu est configuré pour être enregistré sur le serveur.

New Storage

Name :

Type :

Storage Access Password

Complete Access :

Confirm Password :

Set this password to allow access to the Storage for registering, retrieving and deleting packages.

Read-Only Access :

Confirm Password :

Set this password to restrict access to the Storage for retrieving packages only.

Le serveur d'actifs réutilisables peut être utilisé pour stocker des informations et des actifs modélisation entre les projets, et est particulièrement utile pour stocker des informations en vue de leur réutilisation entre des projets ou des programmes. Lorsqu'un projet a apporté sa valeur à l'entreprise, il est assez courant que les artefacts du projet soient archivés et rendus inaccessibles aux autres équipes. Le serveur d'actifs réutilisables est un endroit pratique pour stocker ces artefacts afin qu'ils puissent être réutilisés par d'autres équipes. Par exemple, un projet qui a développé des modèles pour un hôpital nouveau ou modernisé pourrait stocker ces précieux artefacts modélisation dans le RAS, puis chaque fois qu'un autre hôpital est en cours de construction, ils pourraient être réutilisés, économisant ainsi potentiellement des milliers d'heures de travail. Cette image montre les détails d'un Paquetage enregistré dans le RAS, y compris le numéro de version, l'identifiant utilisateur global et les commentaires qui aideront un ingénieur à rechercher du contenu réutilisable pour trouver les actifs qu'il recherche.

Package Details

Name :

GUID :

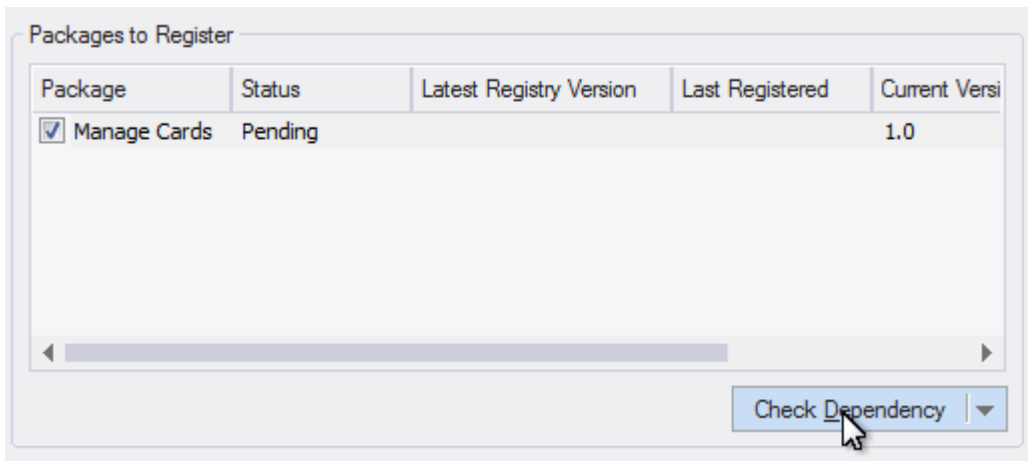
Current Version :

Comments :

Notes :

L'un des principaux avantages de cette fonctionnalité est que le RAS conserve les actifs dans une arborescence de dépendances, ce qui permet à un utilisateur potentiel de l'actif de comprendre les Paquetages dont il dépend. Il s'agit d'un

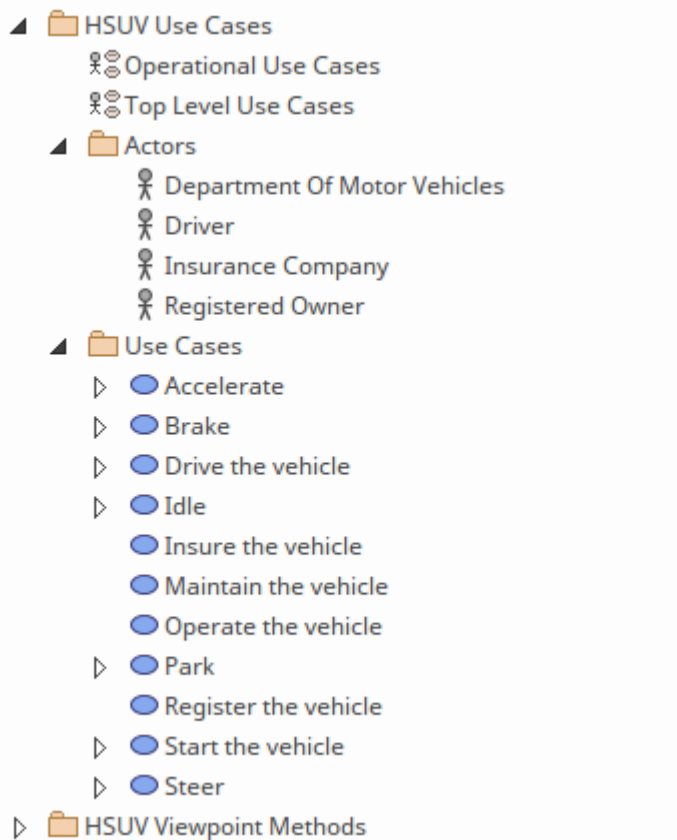
mécanisme analogue utilisé par les programmes d'installation de logiciels qui déterminent si un élément logiciel sélectionné pour l'installation dépend d'autres éléments qui ne sont pas présents sur la machine cible, et si ces éléments ont à leur tour d'autres dépendances. Le Service d'Actifs Réutilisables effectue ce travail pour l'ingénieur et effectue une traversée des graphes de dépendances, permettant à l'utilisateur de comprendre de quoi dépend l'actif requis. Cette capture d'écran montre comment les dépendances peuvent être gérées dans le RAS.



Pour plus d'informations, consultez la rubrique d'aide [Reusable Asset Service \(RAS\)](#).

## Utiliser Paquetages pour Structurer le Référentiel

Le Paquetage est l'un des éléments les plus fondamentaux et les plus importants de SysML. Il fonctionne comme un conteneur et, vu simplement, il ressemble à un dossier dans votre logiciel d'exploration de fichiers préféré pour votre ordinateur. Ainsi, de cette façon, il s'agit d'abord d'un conteneur qui regroupe d'autres éléments, y compris d'autres Paquetages, mais nous verrons dans cette rubrique qu'il a également d'autres fonctions importantes dans Enterprise Architect.

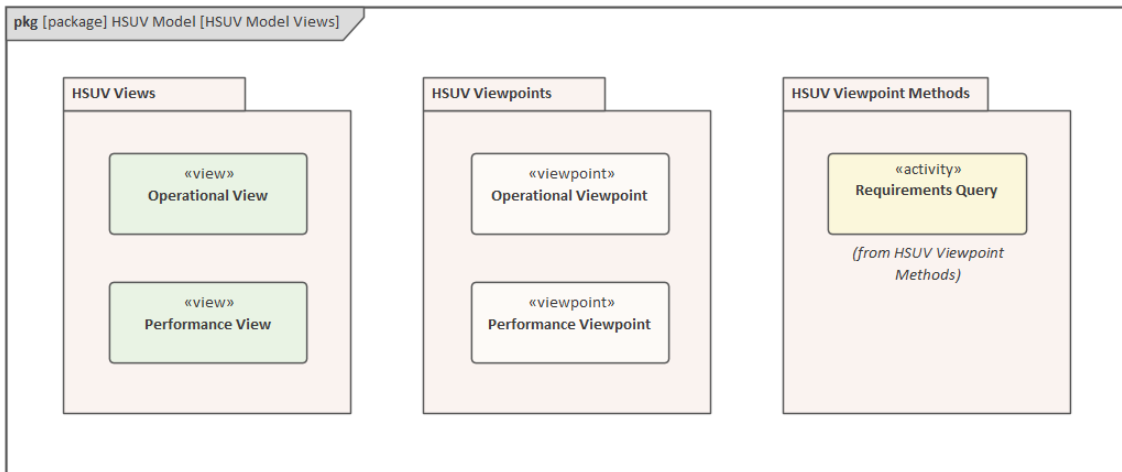


Dans un sens plus profond, un Paquetage est un espace de noms qui permet d'identifier de manière unique tout élément d'un référentiel, de manière similaire au fonctionnement d'une URL. Le chemin affiché ici a été extrait automatiquement d' Enterprise Architect ; il vous permet de visualiser l'espace de noms.

Exemple SysML. Modèle HSUV. Cas d'utilisation HSUV. Cas d'utilisation. Conduire le véhicule

La mise en place de la structure Paquetage est une tâche importante et souvent redoutée, mais heureusement, Enterprise Architect élimine une grande partie de l'angoisse que ressentent les nouveaux venus (et les modélisateurs expérimentés) lorsqu'ils abordent cette tâche.

Les prochains sujets vous présenteront les meilleures pratiques pour configurer une structure Paquetage, ainsi que certains des outils et facilités supplémentaires qui rendront le travail avec Paquetages beaucoup plus facile.



## The Function of Packages

En plus de la discussion précédente, Paquetages sont un élément important dans l'utilisation d' Enterprise Architect car ils sont utilisés comme base pour un certain nombre de facilités de l'outil, notamment :

- Conteneur pour éléments,
- Définition Namespace ,
- Sécurité,
- Contrôle de Version ,
- Lignes de base ,
- Importation et exportation,
- Documentation,
- Audit,
- Modélisation Consciente du Temps , et bien plus encore.

Ces éléments doivent tous être pris en compte lors du choix de la structure des Paquetages . Le conteneur et l'espace de noms sont les plus critiques, mais toutes les autres fonctions doivent être prises en compte lors du choix d'une structure de modèle initiale ou lorsque la structure du modèle est en cours de refactorisation. Il arrive souvent que certaines fonctions ne soient pas utilisées initialement et ne soient mises en jeu que lorsque le référentiel a atteint un certain degré de maturité. C'est souvent un déclencheur pour que le référentiel soit réorganisé, mais heureusement, en raison de la facilité du glisser-déposer, cela peut être fait facilement et efficacement et n'est pas un exercice chronophage.

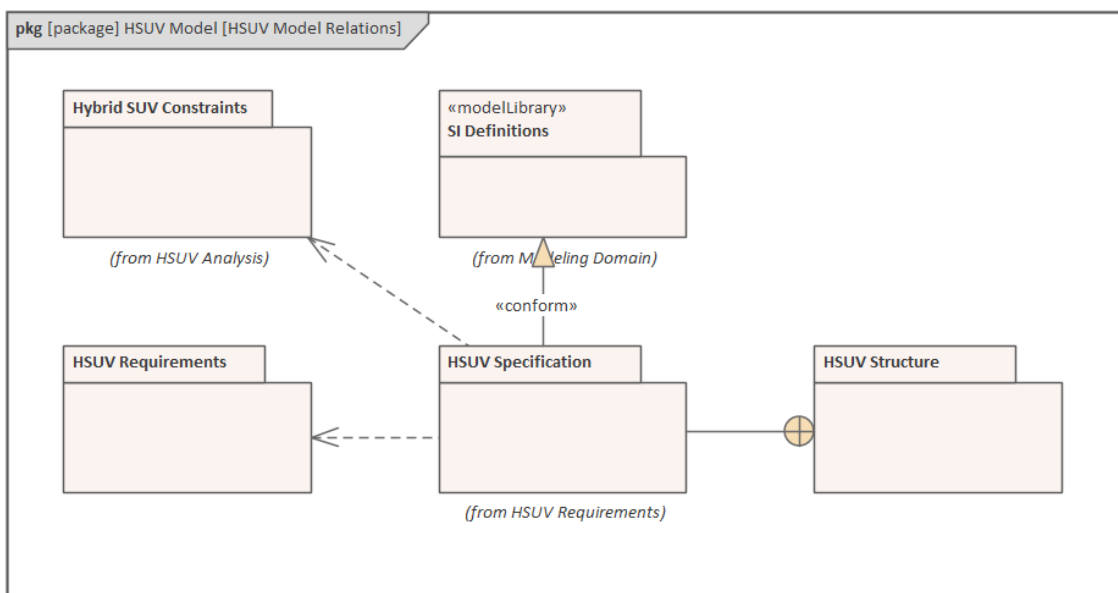


## Présentation Diagrammes Paquetage

Le diagramme Paquetage est un diagramme simple qui décrit visuellement la structure du référentiel, y compris les relations entre Paquetages et d'autres Paquetages et éléments. Les diagrammes Paquetage semblent assez simplistes avec un petit nombre d'éléments :

- Modèle
- Bibliothèque d'Équipe
- Paquetage
- Vue
- Point Vue
- Partie prenante

Ceux-ci sont reliés par une série de relations.

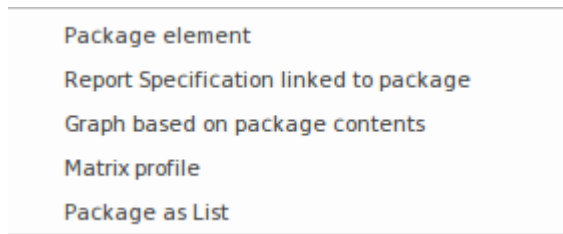


Encore une fois, le nombre de relations est assez limité, mais chacune a une signification spécifique dans le diagramme .

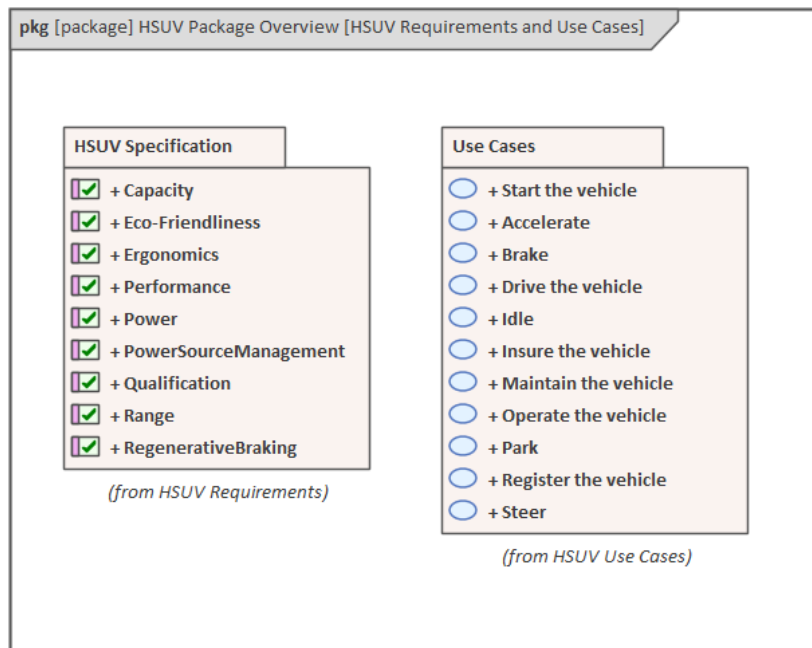
- Se conformer
- Dépendance
- Importer
- Endiguement
- Réalisation
- Affiner
- Exposer

Comme pour tous les éléments SysML, les éléments ont à la fois un aspect graphique et textuel. Les notes peuvent être ajoutées à chacun des Paquetages et aux relations pour clarifier le but de l'élément ou du connecteur. Le diagramme Paquetage peut contenir n'importe quel type d'élément de modèle, mais il contient généralement des Paquetages . Enterprise Architect étend la spécification SysML en fournissant un certain nombre de façons différentes et innovantes de visualiser des Paquetages et leur contenu sur un diagramme Paquetage . Ces options sont visibles dans le menu qui s'affiche lorsqu'un

Paquetage est glissé de la fenêtre Navigateur vers un diagramme .



Dans ce diagramme nous pouvons voir que le modélisateur a choisi l'option « Élément Paquetage » et a défini la visibilité des compartiments de l' objet diagramme Paquetage pour afficher le contenu Paquetage . Les options de visibilité des compartiments sont disponibles dans le menu contextuel d'un objet diagramme pour n'importe quel élément, et Enterprise Architect modifie dynamiquement les options en fonction du type d'élément et des compartiments disponibles.



Dans le diagramme suivant, l'ingénieur a choisi l'option « Paquetage as List », qui s'appuie sur un code SQL généré automatiquement mais configurable par l'utilisateur pour créer une liste dynamique d'éléments basée sur l'une des métadonnées de l'élément Paquetage . Nous voyons ici la même liste d'exigences, mais cette fois-ci, un certain nombre de propriétés sont également affichées.

pkg [package] HSUV Package Overview [HSUV Requirements List]

List of Elements in Package HSUV Specification			
	NAME	TYPE	STATUS
	Decomposition of Performance Requirement	Class	Proposed
	Tree of Performance Requirements	Class	Proposed
	Acceleration	Requirement	Validated
	Braking	Requirement	Implemented
	Capacity	Requirement	Proposed
	CargoCapacity	Requirement	Proposed
	Eco-Friendliness	Requirement	Proposed
	Emissions	Requirement	Proposed
	Ergonomics	Requirement	Proposed
	FuelCapacity	Requirement	Approved
	FuelEconomy	Requirement	Validated

Showing 1 - 10 of 20 items

N'importe quel nombre de diagrammes Paquetage peut être créé pour définir ou aider à visualiser la structure du référentiel. Pour plus d'informations, consultez la rubrique d'aide [Package Diagram](#).

## Création de Diagrammes Paquetage

Un diagramme Paquetage peut être créé à partir de plusieurs emplacements dans l'Interface Utilisateur, en sélectionnant :

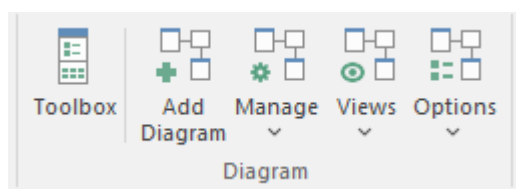
- Ruban de conception - *Ajouter* une icône sur le panneau *Diagramme*
- Barre d'outils de la fenêtre Navigateur - Icône *Nouveau Diagramme*
- Menu contextuel de la fenêtre Navigateur - *Ajouter Diagramme*

Les options d'accès afficheront toutes la même dialogue, ce sont simplement des points d'entrée différents vers les mêmes fonctionnalités de l'outil. Nous utiliserons le ruban Conception pour créer un diagramme Paquetage.

Sélectionnez d'abord l'emplacement dans la fenêtre Navigateur où vous souhaitez placer le diagramme Paquetage. Il peut s'agir soit d'un Paquetage, soit d'un élément, mais il est courant d'insérer diagrammes Paquetage dans un Paquetage.

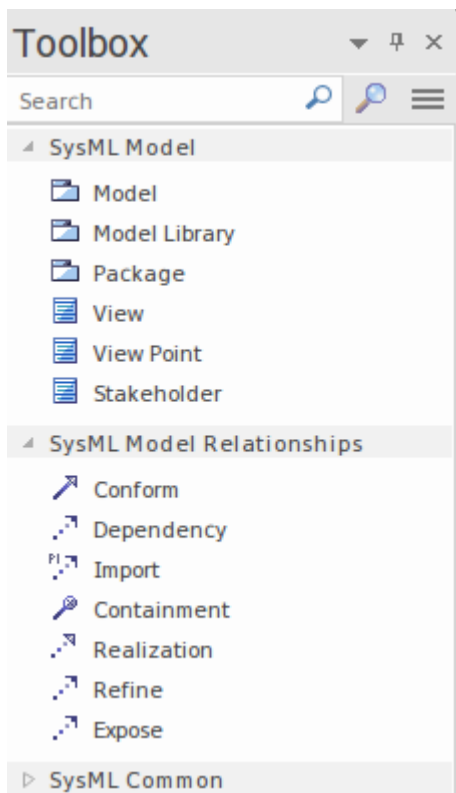
Une fois l'emplacement Paquetage sélectionné dans la fenêtre Navigateur, sélectionnez l'option du ruban :

Conception > Diagramme > Ajouter Diagramme



La sélection de cette option ouvre la dialogue *Nouveau Diagramme*, dans laquelle vous nommez le diagramme ; le nom par défaut est le nom du Paquetage ou de l'élément qui contient le diagramme. Avec la perspective SysML choisie et la version de SysML sélectionnée, une liste de diagrammes s'affiche à partir de laquelle vous sélectionnez le diagramme Paquetage. Cliquez sur le bouton OK pour créer un nouveau diagramme Paquetage à l'emplacement sélectionné dans la fenêtre Navigateur. La Vue Diagramme s'ouvre, vous permettant de commencer à ajouter des éléments et des connecteurs qui décrivent la structure du système et sa division en ces groupes structurels. Enterprise Architect affiche également les pages de la boîte à outils « Paquetage » qui contiennent les éléments et les relations tels que définis par la spécification SysML à appliquer pour la construction diagrammes Paquetage. N'importe quel nombre d'autres pages de

la boîte à outils peut être ouverte, si nécessaire, en plus des pages « Éléments communs » et « Relations communes » qui seront toujours disponibles.



Les éléments et connecteurs les plus importants utilisés avec le diagramme Paquetage sont :

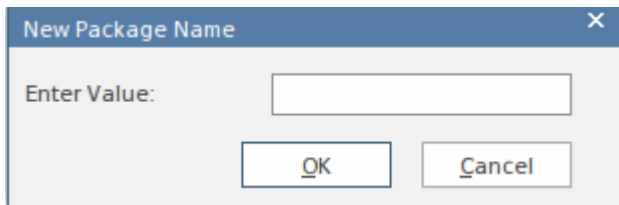
### Éléments

- Modèle - utilisé pour définir une partie de haut niveau du système
- Bibliothèque d'Équipe - utilisée pour définir un ensemble d'éléments réutilisables
- Paquetage - utilisé pour créer une unité structurelle de base
- Vue - utilisé pour définir ce qu'une partie prenante verra lors de la visualisation
- Point de vue - utilisé pour définir un point de référence pour une vue
- Partie prenante – utilisé pour décrire les personnes ou les parties ayant un intérêt matériel

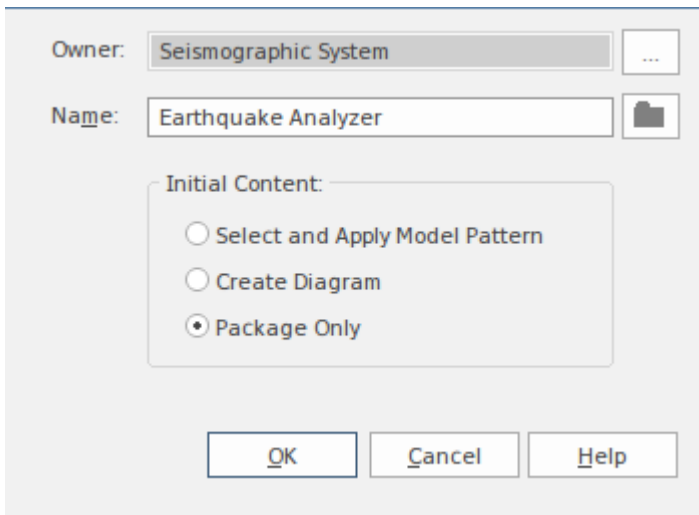
### Connecteurs

- Se conformer
- Dépendance
- Importer
- Endiguement
- Réalisation
- Affiner
- Exposer

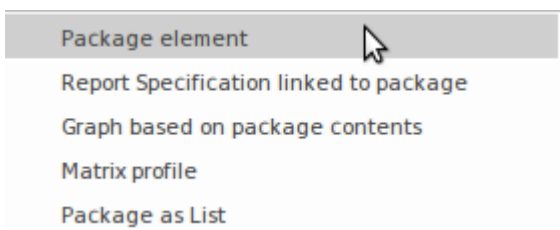
Les éléments peuvent être ajoutés au diagramme en les faisant glisser de la boîte à outils vers la Vue Diagramme . Il est considéré comme une bonne pratique de commencer par définir Modèle et les bibliothèques Modèle . Lorsqu'une icône Modèle ou Bibliothèque d'Équipe est glissée de la boîte à outils vers un diagramme , le modélisateur sera invité à saisir un nouveau nom Paquetage à l'aide de cette dialogue :



Une fois le nom Paquetage saisi, un nouvel élément Paquetage sera créé sur le diagramme . Lors de la création d'un Paquetage à partir de la Boîte à outils ou de l'insertion d'un Paquetage dans la fenêtre Navigateur , le modelleur dispose d'un certain nombre d'options comme indiqué dans le dialogue de cette capture d'écran.



Comme décrit précédemment, lorsqu'un Paquetage existant est glissé depuis la fenêtre Navigateur (ou copié depuis un diagramme existant), Enterprise Architect vous permet de le créer de différentes manières, comme indiqué dans cette capture d'écran :



## Régimes d'organisation Paquetage

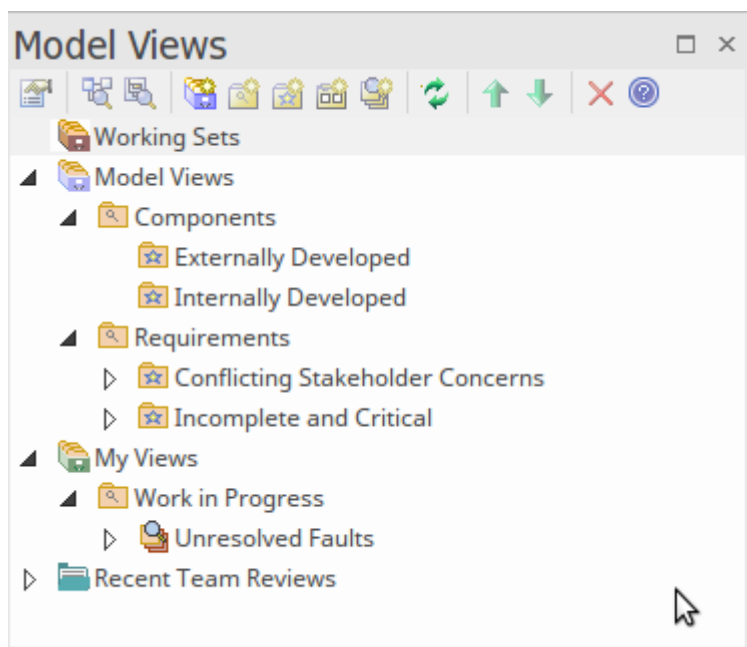
Comme nous l'avons déjà indiqué, les bibliothécaires, les administrateurs ou les ingénieurs impliqués dans la mise en place du référentiel peuvent se retrouver en conflit quant à la direction à prendre, car il existe un large éventail de principes d'organisation qui peuvent être utilisés pour structurer le contenu du référentiel. En voici quelques-uns :

- Une structure de décomposition - Systèmes | Sous-systèmes | Composants | Pièces
- Équipes d'ingénierie travaillant sur différents aspects d'un système - Équipe 1 | Équipe 2
- Programmes de travail et projets - Programme 1 | Projet 1, Projet 2
- Divisions au sein d'une méthode - Architecture | Exigences | Design
- Sécurité et contrôle d'accès
- Facilité de navigation

Chacun de ces principes ou une combinaison de ceux-ci peut être utilisé pour structurer le référentiel, et ils peuvent être modifiés au fil du temps pour s'adapter à l'évolution des pratiques d'ingénierie, de l'utilisation du modèle et de l'expérience des utilisateurs. Le principe le plus difficile est peut-être celui de rendre le référentiel convivial pour ses utilisateurs, afin de garantir une navigation aisée pour qu'ils puissent trouver facilement ce qu'ils recherchent. Enterprise Architect dispose de facilités utiles pour réduire cette tension, permettant d'utiliser d'autres mécanismes pour la navigation et de libérer la conception du référentiel pour qu'elle se développe sur la base d'un certain nombre d'autres principes plus importants. Certaines de ces fonctionnalités d'outils sont répertoriées ici.

## Modèle Vues

Propose un mécanisme flexible et efficace permettant à un ingénieur ou à une équipe de créer n'importe quelle vue du modèle qu'ils jugent utile. Grâce à cette facilité les modélisateurs et les ingénieurs n'ont plus besoin d'accéder à la fenêtre Navigateur, car ils peuvent localiser les éléments qui les intéressent via la fenêtre Modèle Vues.



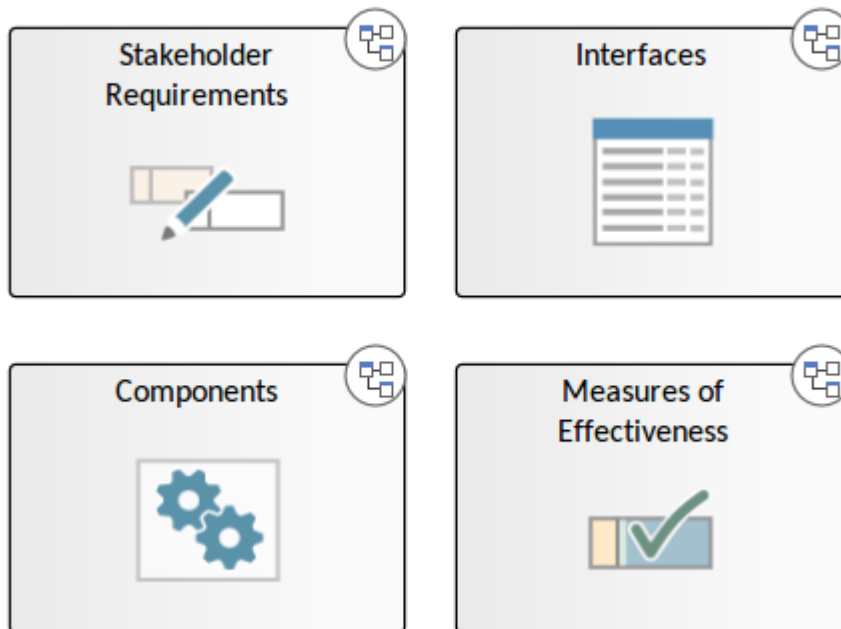
Par exemple, des vues peuvent être créées à partir d'une recherche qui renvoie des éléments de n'importe quelle partie du référentiel ; un ingénieur peut définir une vue qui renvoie toutes les exigences hautement prioritaires, avec le statut « Approuvé » et marqué comme « Difficile », quel que soit le projet auquel elles appartiennent ou leur emplacement dans la hiérarchie Paquetage. Alternativement, un modélisateur peut simplement sélectionner des éléments et diagrammes particuliers qui lui sont importants et les inclure dans une Vue Favoris, ou créer une vue basée sur des Composants nouvellement créés. Cette facilité fournit un mécanisme très flexible pour accéder aux parties importantes du référentiel, et les vues peuvent être créées au niveau du modélisateur ou de l'équipe. Nous reviendrons sur la facilité Modèle Vues

dans une rubrique ultérieure, car c'est une partie extrêmement utile de l'outil.

## Diagramme des cellules de navigation

Enterprise Architect a facilité la navigation des utilisateurs dans un référentiel en fournissant un mécanisme diagramme permettant de créer un lien hypertexte vers n'importe quel diagramme du référentiel.

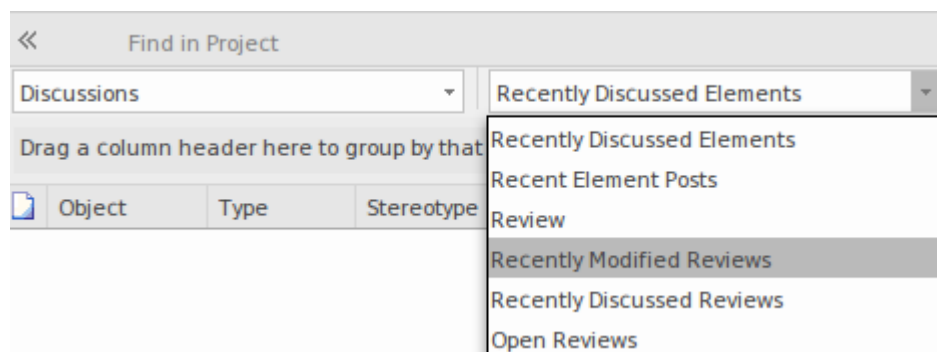
### Retail Self Checkout Project- Home



Cela permet aux bibliothécaires et même aux modélisateurs eux-mêmes de créer un nombre illimité de diagrammes qui servent de rampes de lancement et qui amèneront le lecteur vers diagrammes qui l'intéressent, l'empêchant ainsi de devoir connaître la structure du référentiel. Ces diagrammes sont consultables via le navigateur Internet et les produits Cloud et offrent une expérience convaincante aux utilisateurs occasionnels et aux non-modélisateurs.

## Facilité de recherche

Il s'agit d'une fonctionnalité puissante qui fournit des recherches intégrées et définies par l'utilisateur pour récupérer une liste d'éléments ou diagrammes qui répondent à un ensemble de critères spécifié. La quantité d'informations contenues dans un référentiel peut croître de manière exponentielle à mesure que davantage de personnes contribuent aux modèles et que des informations sont importées à partir de sources externes telles que Risques, Politiques, Rules, Principes, etc. Il existe un ensemble riche et utile de recherches définies dans le cadre du produit et, dans de nombreuses circonstances, une de ces recherches intégrées suffira à un modélisateur ou à un ingénieur pour localiser les éléments ou diagrammes qu'il recherche. Ces recherches peuvent être pilotées par des paramètres, fournissant un mécanisme permettant de réutiliser une recherche pour trouver une variété d'éléments. Par exemple, une recherche peut être écrite avec un paramètre d'entrée utilisateur de Status permettant aux utilisateurs de saisir un statut, par exemple « Proposé » au moment de l'exécuter de la recherche.

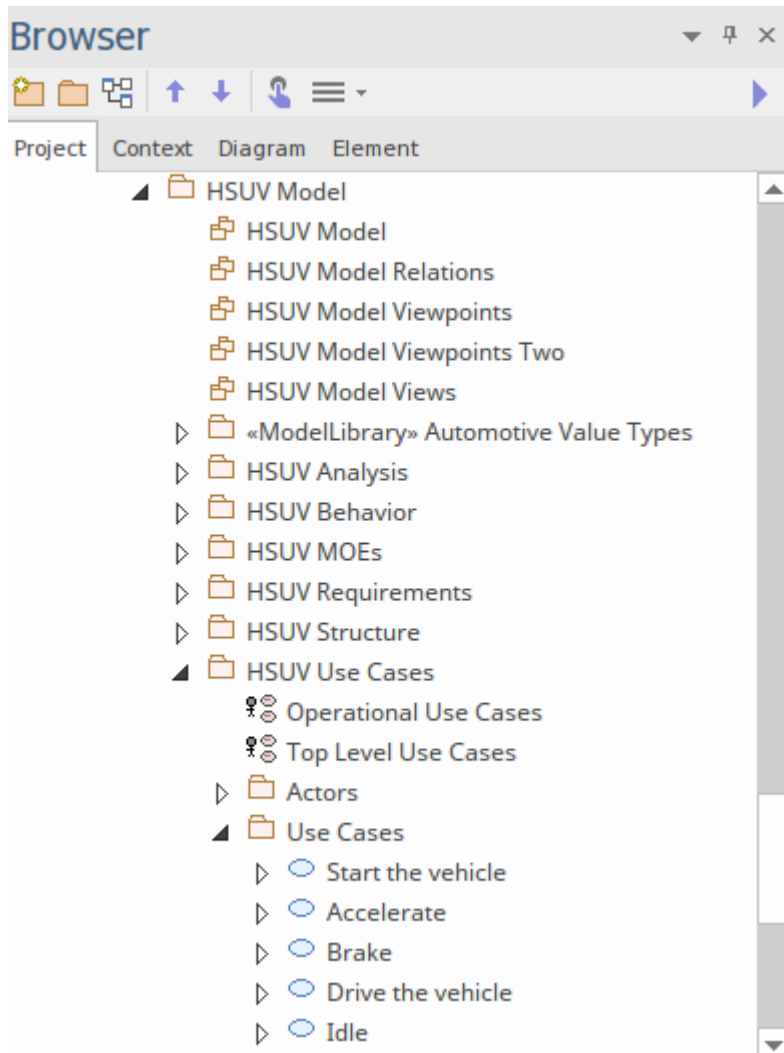


Les recherches peuvent être créées par du personnel non technique à l'aide du générateur Query intuitif, mais il existe également un certain nombre d'autres moyens de créer des recherches, notamment des requêtes basées sur SQL qui nécessitent une connaissance des tableaux de base de données et des requêtes complémentaires qui nécessitent qu'une personne technique crée un programme qui définit la recherche. Ces recherches peuvent être utilisées par un certain nombre d'autres facilités notamment, comme indiqué précédemment, Modèle Vues .



## La fenêtre Navigateur

La fenêtre Navigateur est l'outil principal de structuration et de navigation dans le référentiel, en utilisant des nœuds d'arborescence extensibles et rétractables. L'élément structurel clé est le Paquetage, qui est un élément de type dossier pouvant contenir d'autres éléments et diagrammes, y compris d'autres Paquetages. Les éléments à leur tour peuvent contenir d'autres éléments, fonctionnalités et diagrammes, mais pas Paquetages.

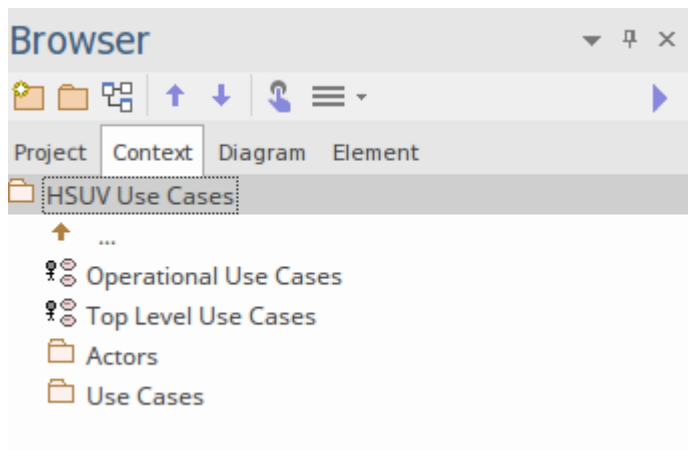



Les nœuds racines sont les nœuds les plus hauts de l'arbre ; ces Paquetages racines peuvent contenir Vues qui peuvent à leur tour contenir n'importe quel niveau de Paquetages et d'éléments. Les nœuds de l'arbre comprenant Paquetages, des éléments, Fonctionnalités et diagrammes peuvent être copiés et collés entre les emplacements, ou glissés et déposés vers de nouveaux emplacements. De nombreux outils, fonctions et fenêtres importants sont appliqués au niveau du Paquetage, tels que l'importation ou l'exportation du contenu du modèle, la documentation et le Contrôle Paquetage, y compris Lignes de base. Pour plus d'informations, consultez la rubrique [The Browser Window](#).

### Diagramme de contexte et navigateurs d'éléments

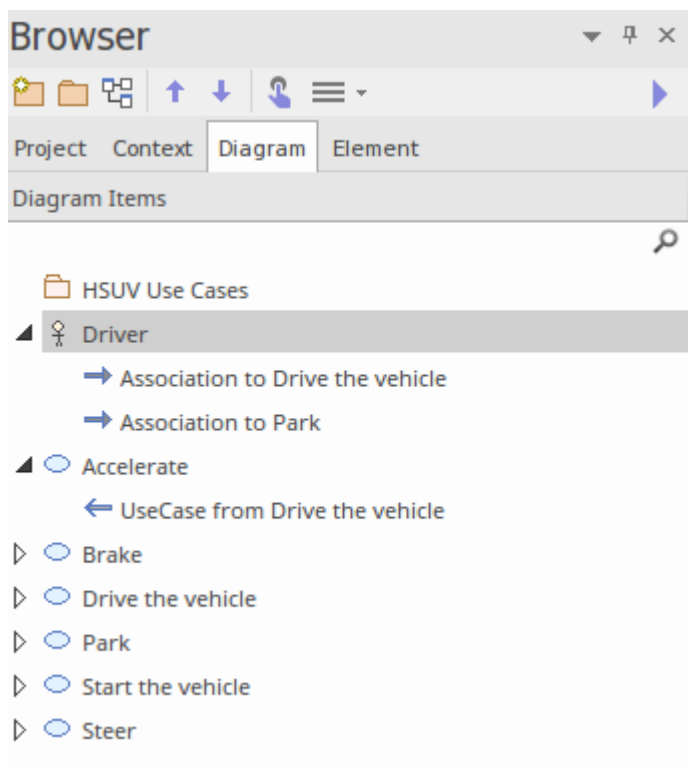
Enterprise Architect propose un certain nombre de navigateurs supplémentaires qui permettent à un ingénieur ou à un modélisateur de se concentrer sur un sous-ensemble du contenu du référentiel. Ces navigateurs peuvent être sélectionnés sous forme d'onglets dans la fenêtre principale Navigateur.

Le *Navigateur de contexte* fournit une vue filtrée d'une branche particulière du modèle, pour travailler uniquement sur la section du référentiel pertinente à un moment donné. Cette vue en vedette supprime le bruit et la complexité de la vue Projet et affiche uniquement la partie du modèle qui l'intéresse, permettant à l'ingénieur de la visualiser de manière isolée.

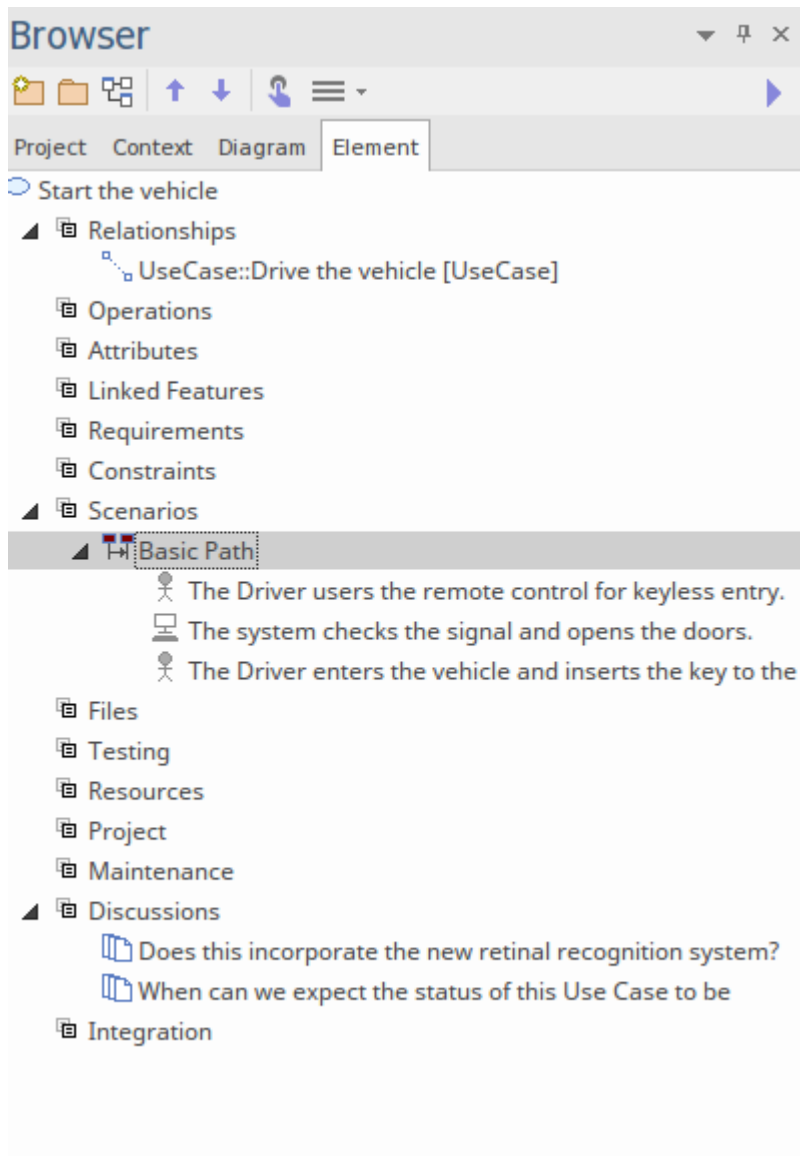


Dans notre exemple, le modélisateur souhaite concentrer son attention sur le Paquetage du cas d'utilisation. En utilisant le bouton , vous pouvez remonter dans l'arborescence, ou en cliquant sur un Paquetage ou un élément qui contient d'autres éléments, ceux-ci peuvent être affichés, mais la vue reste toujours à un seul niveau de hiérarchie.

Le *Diagramme Navigateur* liste les objets présents sur le diagramme actif. Les connecteurs de chaque objet peuvent également être affichés, ce qui permet d'avoir une vue intéressante du diagramme .

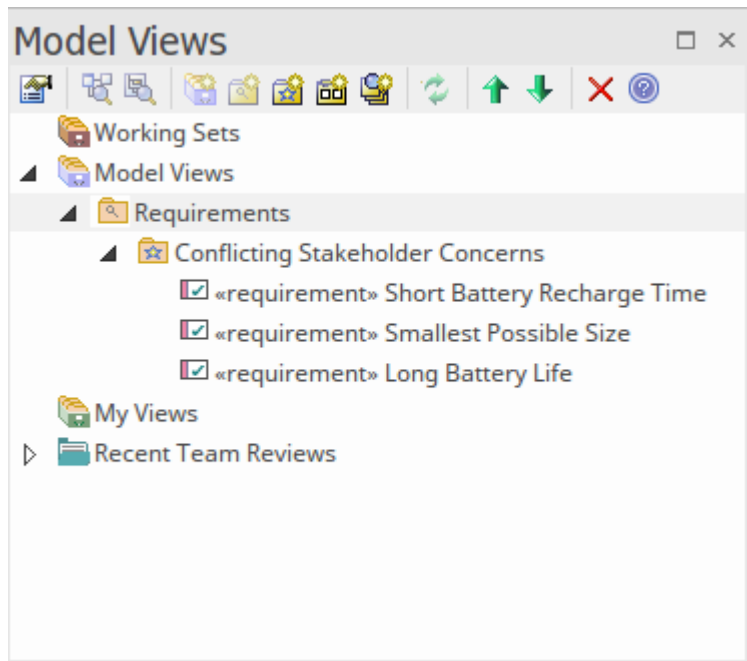


Le *Navigateur d'Élément* affiche des informations sur l'élément actuellement sélectionné. Il permet de visualiser toutes les relations entre les éléments, les scénarios, les exigences, fonctionnalités et bien plus encore.



## Accéder au Référentiel à l'aide de Modèle Vues

Modèle Vues offrent des vues alternatives des éléments du Référentiel . Alors que la fenêtre Navigateur est conçue pour organiser les Paquetages et les éléments de manière structurale, la facilité Modèle Vues permet au modélisateur de créer un certain nombre de vues qui peuvent regrouper les éléments et diagrammes différemment. Il s'agit d'une facilité pratique qui peut être utilisée par des particuliers et des équipes d'ingénierie pour voir le contenu du référentiel dans un certain nombre de vues propriétaires conçues pour présenter uniquement les informations importantes ou pertinentes. Elle vous permet de créer des fenêtres permettant de visualiser le référentiel de manière unique et convaincante, ce qui apportera un aperçu et une clarté, permettant au modélisateur de voir des choses qui n'auraient peut-être pas été possibles avec la fenêtre Navigateur .



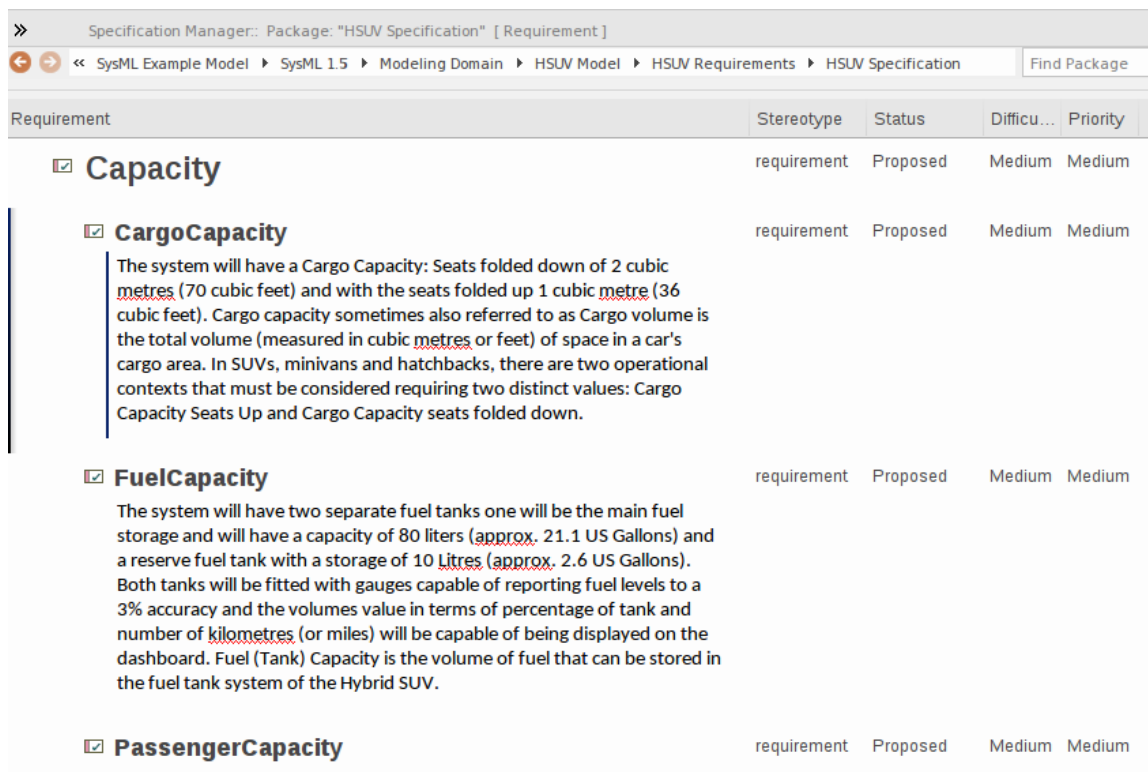
Les vues peuvent être basées sur un large éventail de critères, notamment des dossiers Favoris contenant des éléments d'intérêt sélectionnés à la main, et des dossiers basés sur une recherche telle que « tous les éléments créés la semaine dernière qui ont un statut Proposé » ou « tous les composants fournis par un fournisseur d'ingénierie particulier ». Pour plus d'informations, consultez le [Model Views](#) Rubrique d'aide.

## Définition et gestion des besoins

L'ingénierie des exigences est l'une des disciplines les plus critiques du cycle de vie du développement de solutions et a un impact avéré sur la réussite des projets. Selon les mots du célèbre physicien du XXe siècle, Albert Einstein :

« Si vous définissez un problème avec suffisamment de soin, la solution vous sautera aux yeux. »

Enterprise Architect dispose d'une gamme inégalée d'outils pour développer, gérer, visualiser et documenter les exigences, y compris des outils permettant d'importer ou d'intégrer et de synchroniser avec des systèmes de gestion des exigences externes.



Requirement	Stereotype	Status	Difficu...	Priority
<input checked="" type="checkbox"/> <b>Capacity</b>	requirement	Proposed	Medium	Medium
<input checked="" type="checkbox"/> <b>CargoCapacity</b> The system will have a Cargo Capacity: Seats folded down of 2 cubic metres (70 cubic feet) and with the seats folded up 1 cubic metre (36 cubic feet). Cargo capacity sometimes also referred to as Cargo volume is the total volume (measured in cubic metres or feet) of space in a car's cargo area. In SUVs, minivans and hatchbacks, there are two operational contexts that must be considered requiring two distinct values: Cargo Capacity Seats Up and Cargo Capacity seats folded down.	requirement	Proposed	Medium	Medium
<input checked="" type="checkbox"/> <b>FuelCapacity</b> The system will have two separate fuel tanks one will be the main fuel storage and will have a capacity of 80 liters (approx. 21.1 US Gallons) and a reserve fuel tank with a storage of 10 Litres (approx. 2.6 US Gallons). Both tanks will be fitted with gauges capable of reporting fuel levels to a 3% accuracy and the volumes value in terms of percentage of tank and number of kilometres (or miles) will be capable of being displayed on the dashboard. Fuel (Tank) Capacity is the volume of fuel that can be stored in the fuel tank system of the Hybrid SUV.	requirement	Proposed	Medium	Medium
<input checked="" type="checkbox"/> <b>PassengerCapacity</b>	requirement	Proposed	Medium	Medium

Ces outils implémentent tous les aspects des exigences définies dans la spécification SysML, mais les fonctionnalités de l'outil vont bien au-delà pour créer une plate-forme d'exigences sophistiquée remplie d'outils pour toutes les disciplines associées à la gestion et à la définition des exigences. Non seulement les outils sont utiles aux ingénieurs ou aux gestionnaires travaillant directement avec les exigences, mais il existe une gamme de facilités telles que la fenêtre de traçabilité qui aideront n'importe quelle discipline et qui peuvent être utilisées par les équipes Architecture et de conception qui sont chargées de s'assurer que les exigences sont intégrées dans les conceptions et par conséquent mises en œuvre dans le produit ou le service livré. Pour plus d'informations, consultez la rubrique d'aide [The Requirements Model](#).

## Développer Exigences

Le développement des exigences comprend toutes les activités et tâches associées à la découverte, l'évaluation, l'enregistrement, la documentation et la validation des exigences pour un projet particulier. Exigences sont découvertes, analysées, spécifiées et vérifiées. Enterprise Architect dispose d'une large gamme d'outils et fonctionnalités pour aider l'ingénieur système à développer les exigences. La pièce maîtresse du développement des exigences est le Gestionnaire de Spécification, grâce auquel l'ingénieur peut saisir, visualiser et gérer les exigences sous forme de texte comme dans une feuille de calcul ou un document. Les propriétés des exigences telles que le statut, la priorité et l'auteur peuvent être modifiées en ligne et des filtres peuvent être appliqués pour limiter l'affichage à des exigences particulières.

Item	SysML1.4::text	Status
<input checked="" type="checkbox"/> <b>Operational Visibility</b>	The boom must be visible in all operating conditions including weather events such as fog and low light conditions such as at night.	Approved
<input checked="" type="checkbox"/> <b>Fog and Rain Visibility</b>	The boom must be visible in any weather conditions including Fog and Rain and there must be enough time in these conditions for a driver to stop at the control unit.	Validated
<input checked="" type="checkbox"/> <b>Low Light Visibility</b>	The boom must be 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.	Proposed
<input checked="" type="checkbox"/> <b>Vehicle Height</b>	The boom must allow tall vehicles such as trucks or pantechs to enter and exit the carpark without restriction.	Approved

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.

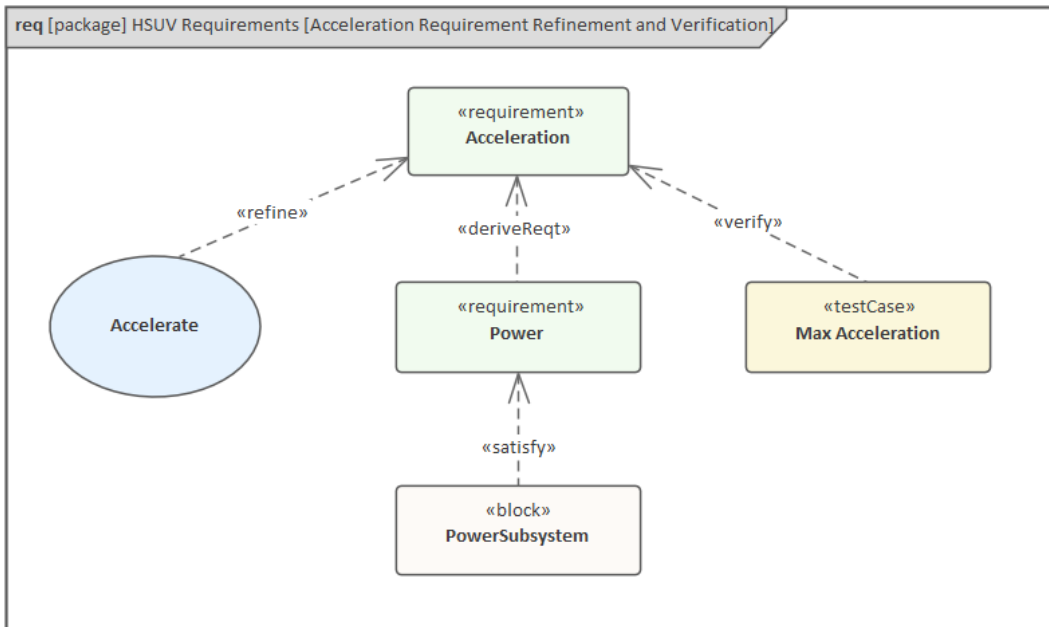
#### 1.2.1 REQ022 - Order Books

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

#### 1.2.2 REQ021 - List Stock Levels

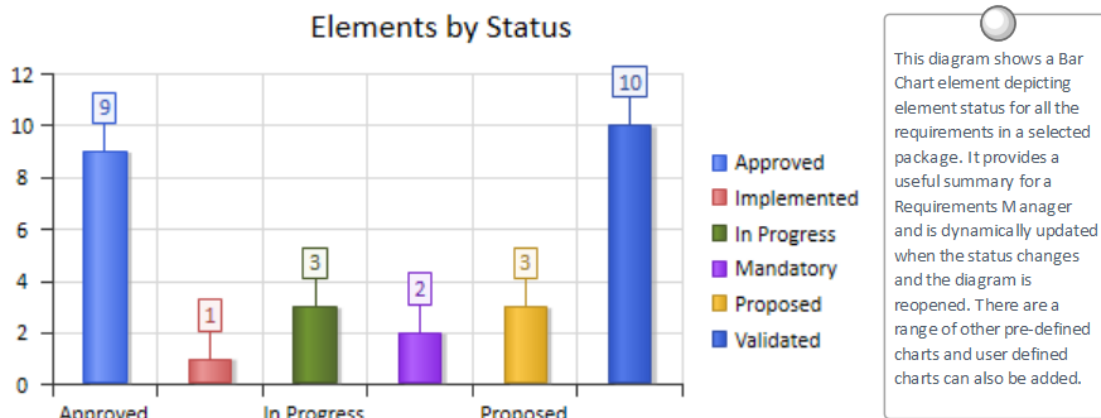
A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.

Le Gestionnaire de Spécification peut être utilisé conjointement avec une plateforme d'autres outils tels que diagrammes , la fenêtre de Traçabilité et le facilité de Discussions.



## Gérer Exigences

Il s'agit des activités visant à maintenir un ensemble d'exigences qui représentent un accord entre l'équipe de projet et le client. Il vise également à garantir que les exigences sont acceptables pour les équipes de conception et de développement et qu'elles sont suffisamment spécifiques pour être mises en œuvre dans des systèmes opérationnels, logiciels ou matériels. Enterprise Architect est une plate-forme sophistiquée de gestion des exigences et, quel que soit le domaine, la taille du projet ou la méthode suivie, Enterprise Architect fournit des outils qui facilitent la gestion des plus grands référentiels d'exigences dans des projets complexes.



## Relations d'exigences

Il existe un ensemble riche de relations d'exigences qui permettent de connecter les éléments d'exigences à d'autres éléments modélisation, y compris d'autres Exigences. Les relations comprennent :

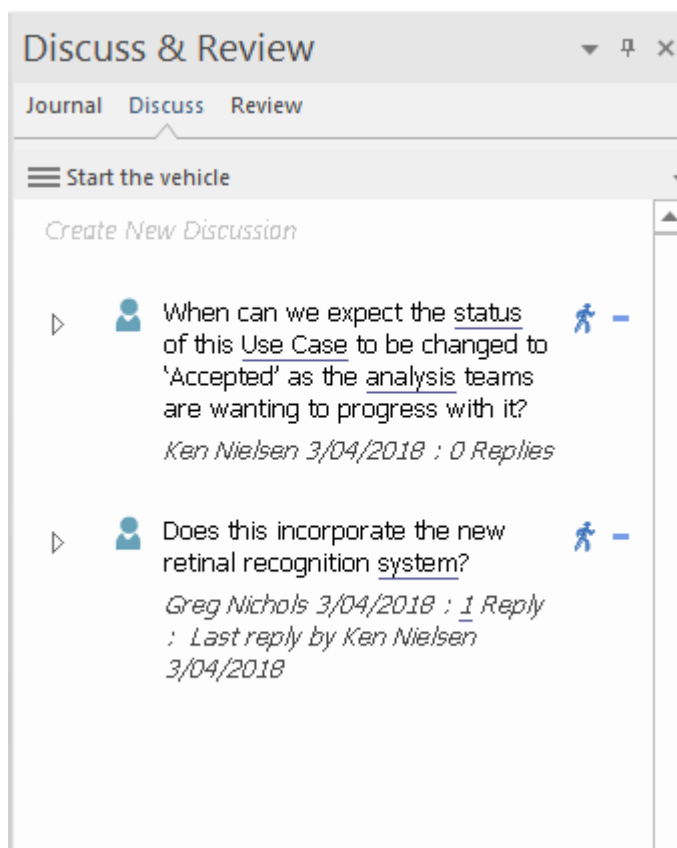
- Endiguement

- Tracer
- Copie
- Dériver
- Vérifier
- Affiner
- Satisfaire

Nous explorerons ces relations en détail dans le corps de ce sujet.

## Visualiser Exigences

La visualisation des exigences est un aspect important du processus d'élaboration des exigences, car il est essentiel que le catalogue d'exigences puisse être consulté par toutes les parties prenantes au fur et à mesure de leur spécification, de leur analyse, de leur développement et de leur gestion. Les exigences représentent l'interprétation par un ingénieur des discussions, observations et articulations faites par les parties prenantes concernant le problème ou l'opportunité en question. Enterprise Architect dispose d'un large éventail de mécanismes permettant non seulement de présenter ces exigences à la communauté des parties prenantes, mais également de permettre leur discussion, leur examen et leur organisation.



## Documenter Exigences

Il existe un certain nombre de documents qui sont généralement produits dans le cadre de la discipline d'ingénierie des exigences, tels que les Spécification Exigences du système et Rapports de cas d'utilisation, et qui peuvent être générés automatiquement à partir d'un modèle d'exigences à l'aide gabarits intégrés ou gabarits définis par l'utilisateur. En outre,



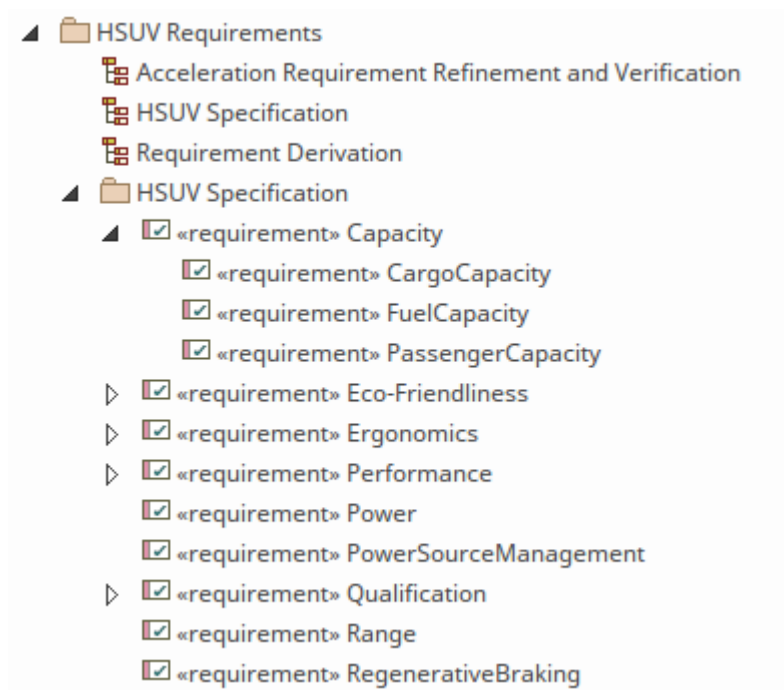
une large gamme d'autres documents peuvent être produits à l'aide gabarits intégrés ou personnalisés.



Il est également possible de visualiser les modèles dans un navigateur Web sur un appareil portable tel qu'un téléphone, une tablette ou un PC. Cette facilité est disponible dans le cadre du produit Pro Cloud Server et offre une alternative à la production de documentation statique et permet à une équipe d'ingénierie de communiquer et de collaborer avec un public étendu en dehors de l'environnement modélisation sans avoir besoin d'installer ou de configurer un logiciel.

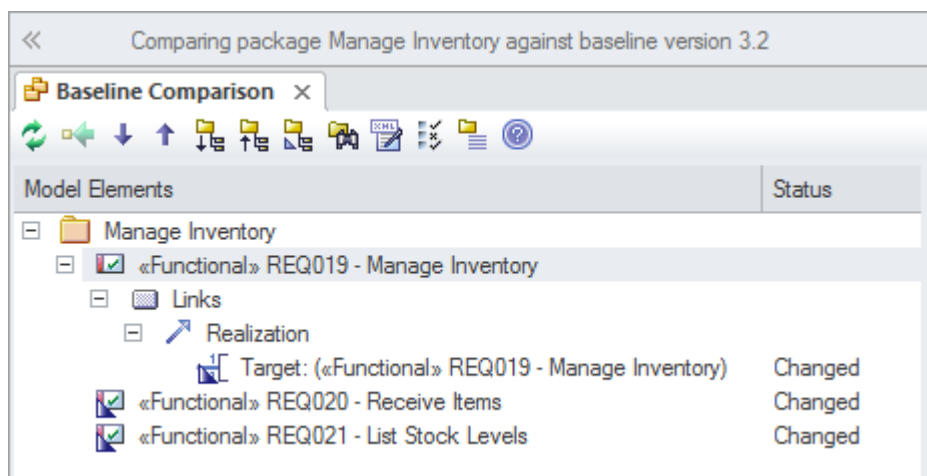
## Exigences des citoyens de première classe

Enterprise Architect propose une large gamme de facilités qui peuvent être utilisées pour le développement, la visualisation, la gestion et la documentation des Exigences en tant qu'éléments de première classe. Les lecteurs de manuels SysML généraux repartiront souvent avec l'idée que Exigences sont exprimées sur diagrammes , mais Enterprise Architect propose une large gamme d'autres moyens de visualiser Exigences qui aideront l'ingénieur lorsqu'il travaillera avec elles en tant qu'éléments textuels, notamment en étant capable de les visualiser dans une hiérarchie dans la fenêtre Navigateur .



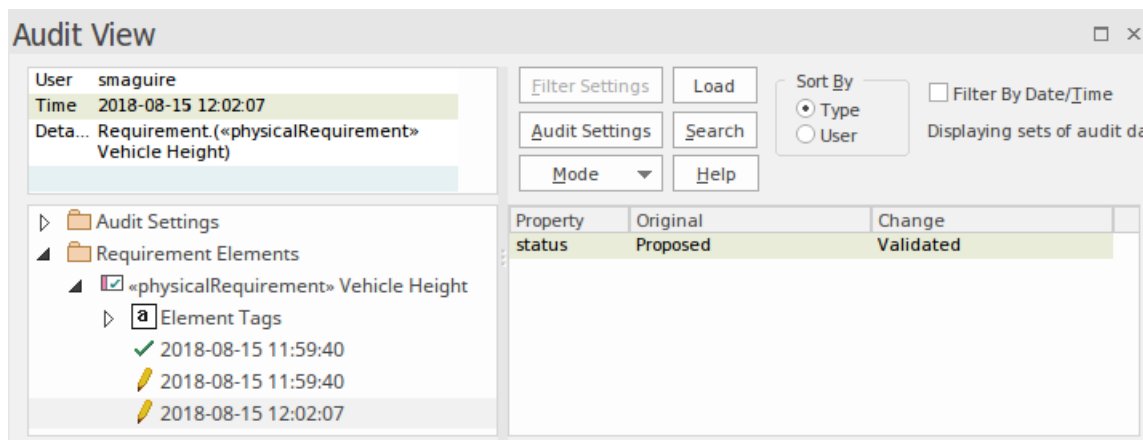
Exigences peuvent être créées dans le cadre d'un cahier des charges ou d'un appel d'offres, ou faire partie d'un document contractuel, auquel cas elles peuvent être facilement importées dans Enterprise Architect . Cependant, il est plus courant qu'elles soient développées dans le cadre d'un effort d'élicitation généralement mené dans le cadre d'ateliers et révisions . Enterprise Architect dispose d'un certain nombre de fonctionnalités qui peuvent être utilisées pour enregistrer le résultat de ces réunions, telles que diagrammes Mind Map. Une fois les ateliers terminés, les idées enregistrées lors de ces réunions peuvent être converties en Exigences ou mappées aux éléments de la réunion d'une manière qui permet leur développement collaboratif.

Les Exigences font souvent partie d'une relation contractuelle entre organisations, ou d'un accord entre différentes sections d'une même organisation, et doivent donc être maintenues et gérées avec rigueur. Enterprise Architect fournit une large gamme de facilités pour aider à cette rigueur, notamment Lignes de base , des outils d'audit, Contrôle de Version et bien plus encore.



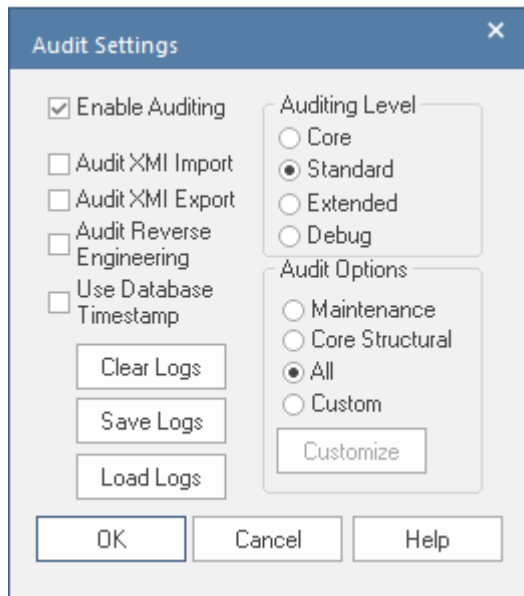
## Audit Exigences

L'audit peut être activé dans un modèle et peut suivre les détails d'une modification d'exigence, y compris le moment où elle a été modifiée, la personne qui l'a modifiée et le delta - avant et après la modification. L'audit peut être utilisé pour suivre ce qui a été modifié dans un modèle, qui l'a modifié et quand. Il existe un certain nombre de modes et un administrateur de référentiel peut utiliser les paramètres pour spécifier ce qui est enregistré dans l'audit. Alors qu'une ligne de base peut être utilisée pour montrer la différence entre un modèle et un instantané à un moment donné, l'outil d'audit enregistre chaque modification individuelle ; il ne peut cependant pas être utilisé pour revenir à un état antérieur (l'outil de ligne de base serait utilisé si cela était nécessaire).



Il s'agit d'une fonctionnalité particulièrement utile dans Ingénierie des Systèmes , lorsqu'il existe des aspects réglementaires ou de conformité dans un processus ou lorsque des défauts doivent être retracés jusqu'à leur conception ou leur spécification d'exigences. L'audit est généralement configuré et administré par un bibliothécaire ou une fonction administrative au sein de l'équipe. L'audit peut être activé, configuré et visualisé à l'aide de l'option du ruban « Paramètres > Modèle > Audit ».

L'audit est désactivé par défaut et doit être activé (activé) avant que le système ne commence à conserver un log d'audit. Cette option, ainsi qu'une série d'autres options, est disponible dans la fenêtre Paramètres d'audit.



Pour plus d'informations, consultez la rubrique d'aide [Auditing](#) .

## Exigences Lignes de base

Lignes de base sont Instantanés d'un Paquetage dans un modèle initiés par l'utilisateur. La ligne de base fait effectivement une copie d'une branche de la hiérarchie Paquetage et de son contenu. À un moment ultérieur, le modèle peut être comparé à la ligne de base et, si le modèle a changé, ces changements seront présentés dans un outil de visualisation, permettant à un utilisateur de visualiser chaque partie du modèle qui a changé, y compris le contenu qui existe dans la ligne de base et le modèle. Il est ensuite possible d'injecter le contenu de la ligne de base dans le modèle au niveau d'un changement discret.

Comparing package Stakeholder against baseline version 2.3.5

Model Elements	Status	Property	Model	Baseline
Stakeholder		Abstract	false	false
«physicalRequirement» All Visibility Operation	Changed	Alias		
«physicalRequirement» Vehicle Height	Changed	Author	Paulene Dean	Paulene Dean
		Date	15/08/2018 12:26:50 PM	15/08/2018 12:26:50 PM
		Date	15/08/2018 12:30:20 PM	15/08/2018 12:26:50 PM
		Complexity	2	1
		Filename		
		Language	<none>	<none>
		IsLeaf	false	false
		IsSpec	false	false
		IsRoot	false	false
		Keywords		
		Multiplicity		
		Name	All Visibility Operation	All Visibility Operation
		Notes		
		Parent	Stakeholder	Stakeholder
		Persistence		
		Phase	1.0	1.0
		Scope	Public	Public
		Status	Approved	Proposed
		Stereotype	physicalRequirement	physicalRequirement
		Type	Requirement	Requirement
		Version	1.2	1.0
		Classifier		
		Visibility		
		Concurrency		
		Cardinality		
		Style		

Lignes de base offrent aux équipes Ingénierie des Systèmes un moyen pratique de s'assurer que les modèles évoluent dans le bon sens, et lorsqu'une direction de modèle doit être rétablie vers une version antérieure, elles peuvent être utilisées pour rétablir des parties atomiques du modèle. Lignes de base peuvent être configurées et visualisées en

appuyant sur Ctrl+Alt+B ou à partir de l'emplacement du ruban :

Ruban : Conception > Paquetage > Gérer > Gérer Lignes de base

Comme indiqué précédemment, Lignes de base sont créées par l'utilisateur et sont stockées dans le référentiel. Par conséquent, si le référentiel est copié, les Lignes de base le seront également. Il est assez courant que la plupart des utilisateurs soient autorisés à créer Lignes de base, mais la possibilité de restaurer à partir d'une Ligne de base est généralement réservée à un rôle de bibliothécaire ou d'administrateur. Pour plus d'informations, consultez la rubrique d'aide [Baseline Tool](#).

## Contrôle de Version

Contrôle de Version permet de versionner Paquetages d'un modèle. Pour commencer à travailler sur une partie du modèle, un utilisateur doit extraire un Paquetage (y compris ses sous-paquetages) puis travailler sur une copie locale. Une fois le travail terminé ou à tout moment, un utilisateur peut archiver le Paquetage ce qui permet aux autres utilisateurs du modèle de voir les modifications.

Contrôle de Version fournit une méthode sophistiquée et robuste de travail avec les modèles et, contrairement à Lignes de base ne nécessite pas qu'un utilisateur lance une version autre que l'extraction du Paquetage. Le système crée automatiquement une version en arrière-plan au fur et à mesure que le travail est effectué et que des modifications sont apportées. Contrôle de Version peut être configuré et visualisé à partir de ces options du ruban.

Paramètres > Contrôle de Version > Projet-VC, Package-VC

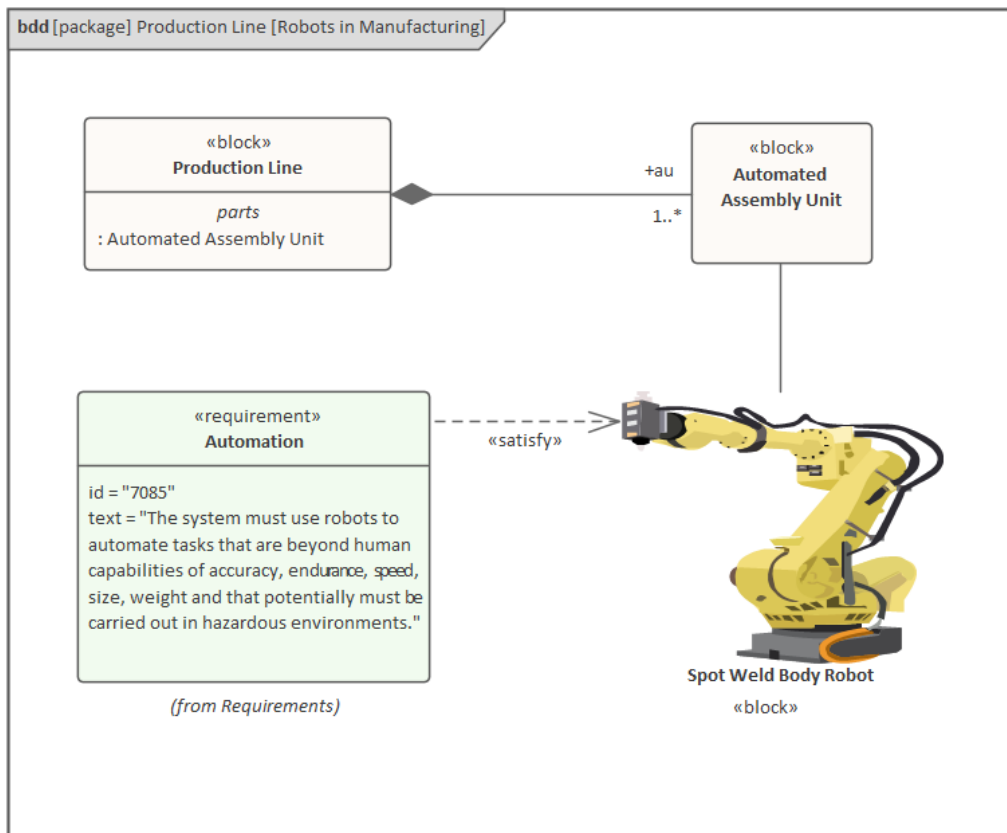
Contrôle de Version fournit un mécanisme efficace de gestion du contenu des modèles et permet à un utilisateur ou à une équipe de garder un contrôle précis sur la façon dont un Paquetage et son contenu évoluent au fil du temps. Pour plus d'informations, consultez le [Version Control of Model Data](#) Rubrique d'aide.

## Présentation Diagrammes d'exigences

Un diagramme d'exigences permet de visualiser Exigences et leurs connexions. Il ne s'agit pas seulement de la relation entre deux Exigences, mais également des relations entre Exigences et d'autres types d'éléments tels que les cas d'utilisation, les activités et Bloc qui peuvent être visualisés sur ces diagrammes. Deux des éléments fournis dans la boîte à outils sont :

- Exigence
- Cas Test

Ces éléments peuvent être connectés entre eux ou à d'autres éléments, créant ainsi des expressions riches.



Dans ce diagramme nous voyons une exigence qui a été connectée à un Bloc à l'aide d'une relation *de satisfaction*, qui décrit comment d'autres éléments garantiront que l'intention de l'exigence est satisfaite. Le Bloc a une image alternative définie, celle-ci étant l'image d'un robot.

Encore une fois, le nombre de relations est assez limité, mais chacune a une signification spécifique dans le diagramme.

- Endiguement
- Tracer
- Copie
- Dériver
- Vérifier
- Affiner
- Satisfaire

Comme pour tous les éléments SysML, ces éléments présentent à la fois un aspect graphique et un aspect textuel. L'exigence a deux propriétés définies :

- id - un identifiant unique pour l'exigence
- Texte - une description textuelle de l'exigence

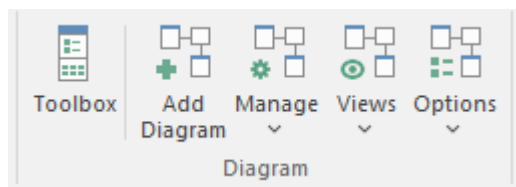
Il est possible de créer un nombre illimité de diagrammes d'exigences pour décrire les besoins et les préoccupations des parties prenantes et autres. Pour plus d'informations, consultez la rubrique [SysML Requirements Modeling](#).

## Création Diagrammes d'exigences

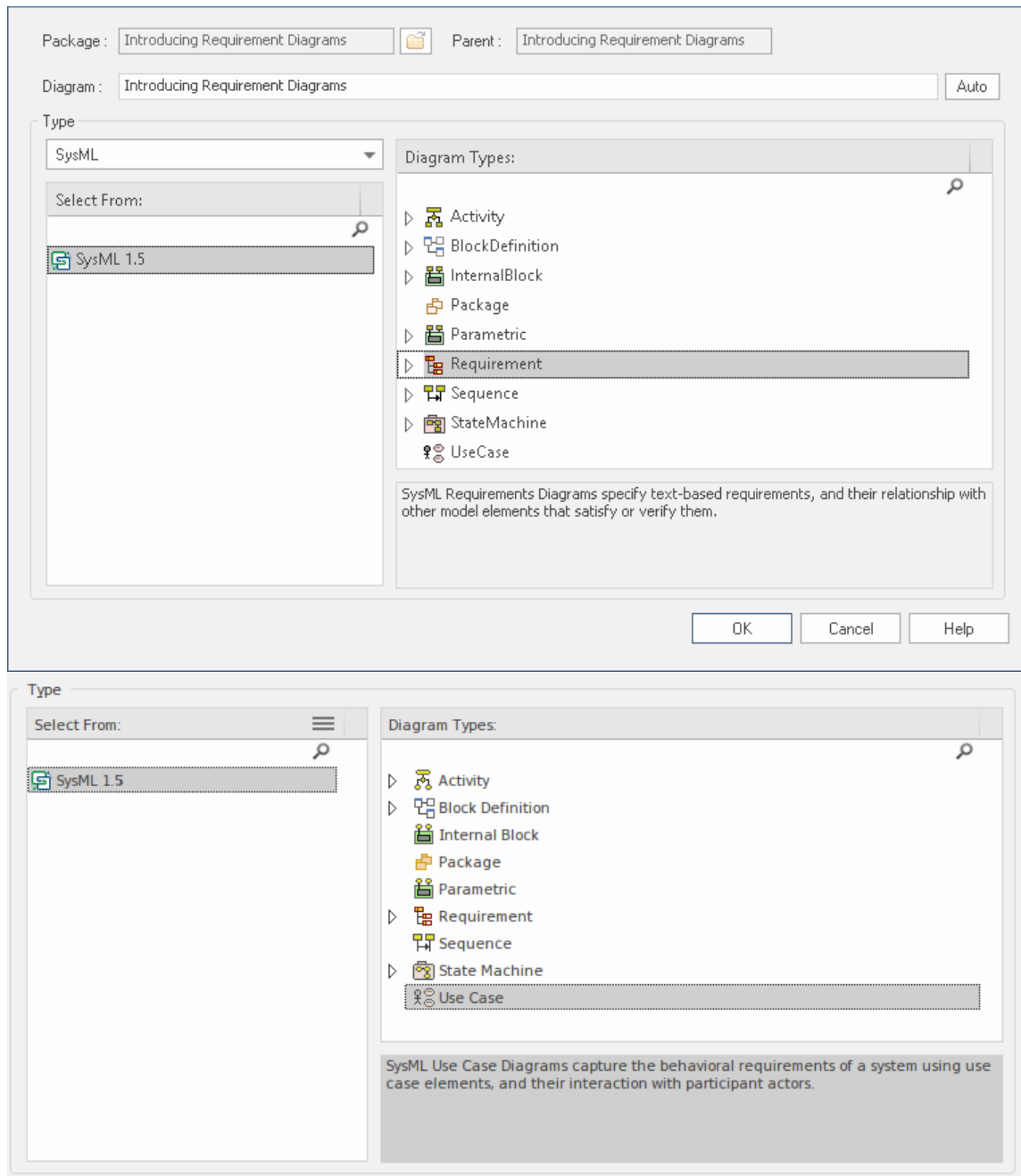
Un diagramme d'exigences peut être créé à partir de plusieurs emplacements dans l' Interface Utilisateur , tels que :

- Ruban de conception - Icône « Ajouter Diagramme » sur le « Diagramme » panneau
- Barre d'outils de la fenêtre Navigateur - Icône 'Nouveau Diagramme '
- Menu contextuel de la fenêtre Navigateur - option « Ajouter Diagramme »

Nous allons utiliser le ruban Conception pour créer un diagramme d'exigences. Tout d'abord, sélectionnez l'emplacement dans la fenêtre Navigateur où vous souhaitez créer le diagramme d'exigences. Comme pour tous diagrammes , il peut s'agir d'un Paquetage ou d'un élément, mais il est courant d'insérer diagrammes d'exigences dans un Paquetage . Une fois l'emplacement Paquetage sélectionné dans la fenêtre Navigateur , cliquez sur l'option du ruban « Conception > Diagramme > Ajouter Diagramme ».



Cette option ouvre la dialogue « Nouveau Diagramme », qui vous permet de renommer le diagramme . Le nom par défaut est celui du Paquetage ou de l'élément qui contient le diagramme . Lorsque vous sélectionnez la Perspective SysML et la version de SysML, une liste de diagrammes s'affiche, vous permettant de sélectionner le diagramme d'exigences. Cliquez sur le bouton OK pour créer un nouveau diagramme d'exigences à l'emplacement spécifié dans la fenêtre Navigateur . La Vue Diagramme s'ouvre, vous permettant de commencer à ajouter des éléments et des connecteurs qui décrivent les Exigences et leurs relations. Enterprise Architect affiche également la page « Exigences » de la Boîte à outils, qui contient les éléments et les relations définis par la spécification SysML comme applicables à la construction diagrammes d'exigences. N'importe quel nombre d'autres pages de la Boîte à outils peut être ouvert selon les besoins, en plus des pages de la Boîte à outils « Éléments communs » et « Relations communes » qui sont affichées par défaut.



Les éléments et connecteurs les plus importants utilisés avec le diagramme des exigences sont :

### Éléments

- Exigence - utilisé pour définir une exigence
- Cas Test - utilisé pour décrire un Test

### Connecteurs

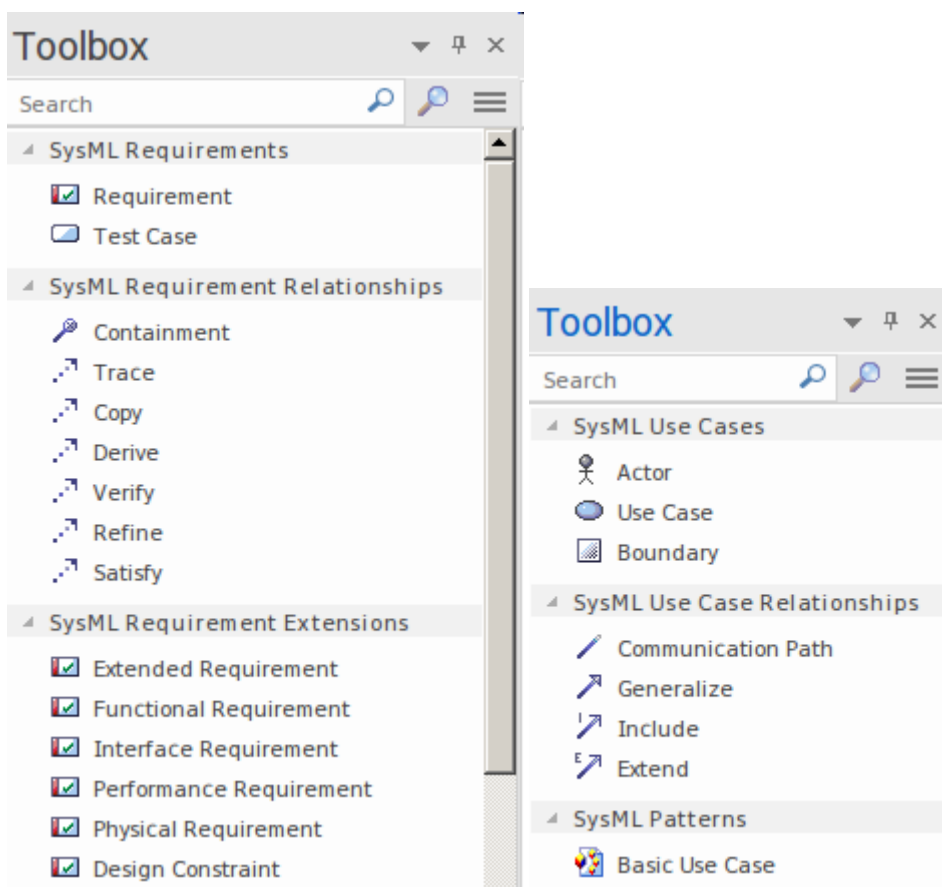
- Confinement - utilisé pour fournir des informations supplémentaires qui aident à clarifier l'exigence
- Trace - utilisé pour connecter une exigence à tout autre élément modélisation
- Copie - utilisé pour montrer qu'une exigence est une copie d'une autre
- Dériver - utilisé pour décrire le fait qu'une exigence est basée sur ou est une extension ou une dérivation d'une autre exigence



- Vérifier - utilisé pour indiquer qu'une exigence a été remplie
- Affiner - utilisé pour ajouter un raffinement ou des informations supplémentaires qui aident à clarifier l'exigence
- Satisfaire - utilisé pour montrer qu'un ou plusieurs éléments du modèle dans l'architecture ou la conception répondent à la notion exprimée dans l'exigence

### Extensions des exigences

- Exigence étendue - utilisée pour Exigences étendues
- Exigence fonctionnelle - utilisée pour Exigences liées à la fonction
- Exigence d'interface - utilisée pour Exigences liées aux interfaces
- Exigence de performance - utilisée pour Exigences liées à la performance
- Exigence physique - utilisée pour Exigences liées aux aspects physiques d'un système
- Exigence de conception - utilisée pour Exigences liées à la conception



Les éléments peuvent être ajoutés au diagramme en les faisant glisser de la boîte à outils vers la Vue Diagramme . Relations peuvent être créées en sélectionnant d'abord la relation requise dans la boîte à outils, puis en effectuant un glisser-déposer entre un élément source et un élément cible. Il est courant de ne pas créer diagrammes d'exigences qui répertorient simplement les Exigences , mais plutôt de créer diagrammes qui montrent les relations entre deux Exigences quelconques ou les relations que les Exigences entretiennent avec d'autres éléments tels que les cas d'utilisation, les activités et les blocs.

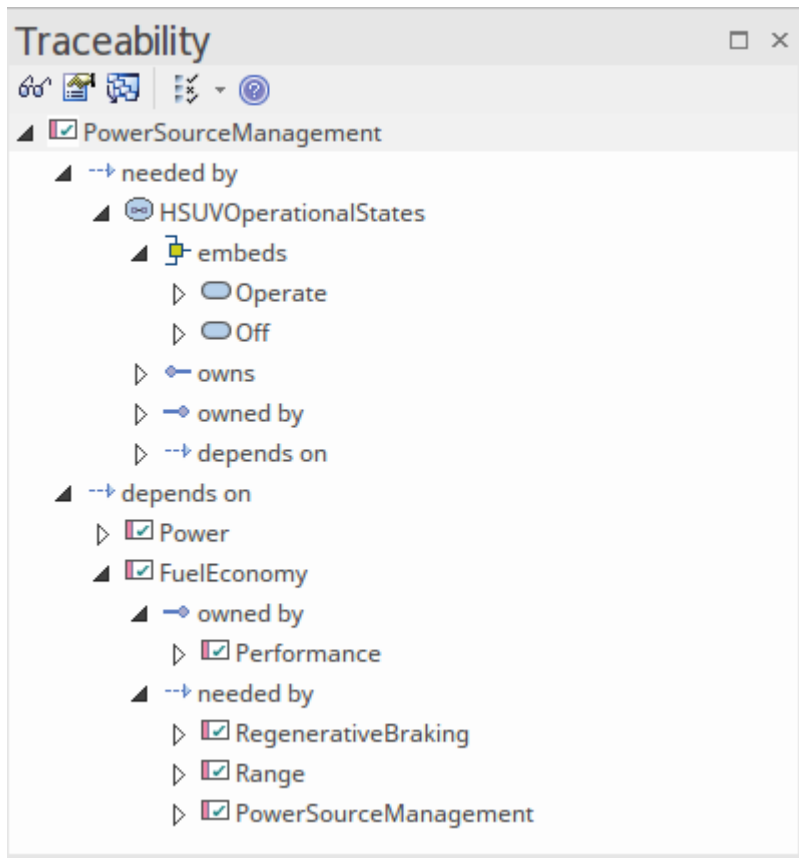
## Développer Exigences

Le développement des exigences comprend toutes les activités et tâches associées à la découverte, l'évaluation, l'enregistrement, la documentation et la validation des exigences pour un projet ou un programme de travail particulier. Exigences sont découvertes, analysées, spécifiées et vérifiées, et Enterprise Architect dispose d'une large gamme d'outils et fonctionnalités pour aider l'ingénieur système à développer les exigences. La pièce maîtresse du développement des exigences est le Gestionnaire de Spécification, qui permet à l'ingénieur des exigences de saisir, d'afficher et de gérer les exigences sous forme textuelle dans un format de feuille de calcul.

Item	SysML1.4::text	Status
<input checked="" type="checkbox"/> <b>Operational Visibility</b>	The boom must be visible in all operating conditions including weather events such as fog and low light conditions such as at night.	Approved
<input checked="" type="checkbox"/> <b>Fog and Rain Visibility</b>	The boom must be visible in any weather conditions including Fog and Rain and there must be enough time in these conditions for a driver to stop at the control unit.	Validated
<input checked="" type="checkbox"/> <b>Low Light Visibility</b>	The boom must be 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.	Proposed
<input checked="" type="checkbox"/> <b>Vehicle Height</b>	The boom must allow tall vehicles such as trucks or pantechs to enter and exit the carpark without restriction.	Approved

Add New ▾

Le Gestionnaire de Spécification peut être utilisé conjointement avec une plateforme d'autres outils, tels que diagrammes, la fenêtre de traçabilité et le facilité de discussions. Ces fenêtres fournissent d'autres vues des exigences, offrant au modélisateur et au visualiseur une compréhension approfondie de la manière dont une exigence est liée à d'autres parties du référentiel, et fournissant des détails non visibles via le Gestionnaire de Spécification.



# Élicitation

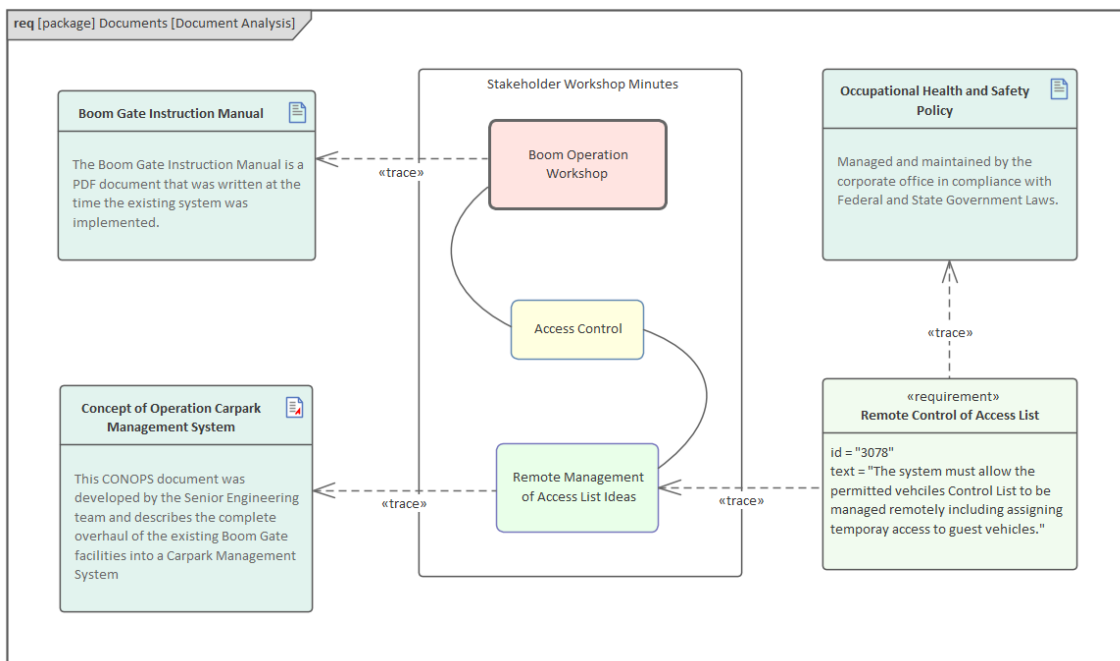
L'élicitation est le processus de découverte d'informations. Les informations recueillies au cours de ce processus constitueront les précurseurs des exigences. Les informations seront généralement brutes et souvent hétérogènes, et ce n'est qu'au cours de la phase d'analyse des exigences que les véritables exigences pourront en être déduites. L'élicitation prendra de nombreuses formes et toutes les compétences de l'ingénieur des exigences seront nécessaires pour déterminer quels documents, machines, outils, personnes et processus examiner pour découvrir les informations.

## Sources des documents

Exigences peuvent souvent provenir d'un large éventail d'endroits, y compris de documents tels que :

- Affaire Métier
- Concept d'opération
- Spécifications des exigences (d'un système existant)
- Manuel d'utilisation
- Document de normes
- Document de politique
- Document réglementaire ou de conformité

Bien que tous ces documents puissent être développés dans Enterprise Architect à l'aide de la facilité Document Artifact, ils sont généralement développés dans d'autres outils et se trouvent en dehors du référentiel. Ils peuvent être glissés dans un diagramme et importés dans le référentiel ou enregistrés comme référence ou substitut du document externe.



Une autre option, peut-être plus utile, consiste à les ajouter à la Bibliothèque d'Équipe, qui est une bibliothèque de documents et de pages Web utilisée pour créer un catalogue d'éléments pouvant être référencés par des exigences. Il est également intéressant d'envisager de réviser le contenu des documents et d'incorporer les informations en tant qu'éléments de modèle. Cela présente l'avantage pour un ingénieur de pouvoir créer des relations de traçabilité entre des éléments tels que les motivations Métier et les éléments de problème et de solution tels que Exigences, les cas d'utilisation et les composants. Pour plus d'informations, consultez la rubrique d'aide [The Model Library](#).

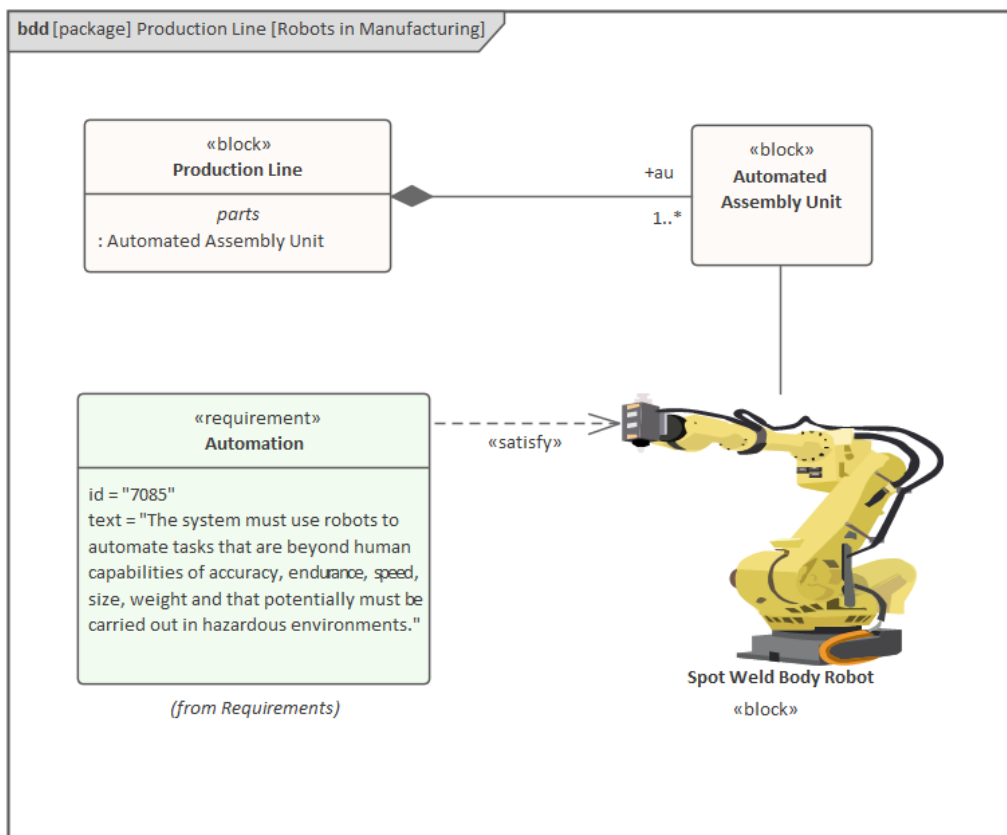
## Observations des utilisateurs

Observer les utilisateurs effectuer leur travail est un moyen utile et discret de comprendre :

- Les tâches qu'ils accomplissent et
- Comment ils utilisent les informations et autres logiciels et dispositifs matériels pour obtenir un résultat à partir de leur interaction avec un système

Même si les processus qui supportent le système prévu sont différents, les observations des processus actuels fourniront un contexte utile pour les discussions. Cela aidera également l'ingénieur à faire preuve d'empathie envers l'utilisateur, ce qui peut se traduire par une compréhension plus approfondie des problèmes auxquels il est confronté et servir de base à la découverte de solutions potentielles. Un ingénieur découvrira souvent des documents, des listes de contrôle et des cartes d'indices non mentionnés qui peuvent aider à éclairer le processus. Équipé d'un téléphone portable ou d'un appareil photo, il est également utile pour l'ingénieur de prendre des photos de l'utilisateur en train de travailler, ce qui aidera les ingénieurs et autres personnes à se souvenir de la tâche et à en discuter pendant la phase d'analyse des exigences.

Enterprise Architect supporte le modélisateur à représenter des fichiers tels que des photos et des documents numérisés directement dans le modèle, créant ainsi une représentation riche et expressive de l'utilisateur au travail. Il est possible de les représenter sous forme d'artefact (qui, avec une seule touche (F12), lancera le fichier) ou d'utiliser un lien hypertexte ou même d'inclure l'image elle-même dans un diagramme. Pour plus d'informations, consultez la rubrique [Changing Element Appearance](#).



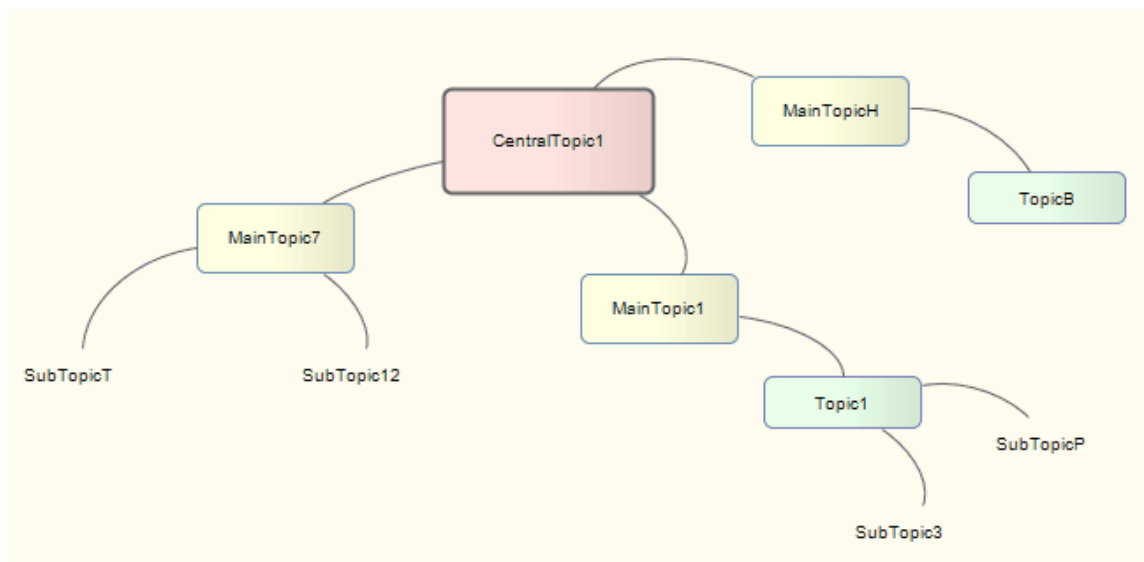
Ce diagramme représente une photographie prise par un ingénieur d'un robot de chaîne de production avancé participant à un système de fabrication. L'image peut être placée dans diagrammes et des relations peuvent être établies entre le robot et d'autres parties de la description du système, y compris les exigences.

## Séminaires des parties prenantes

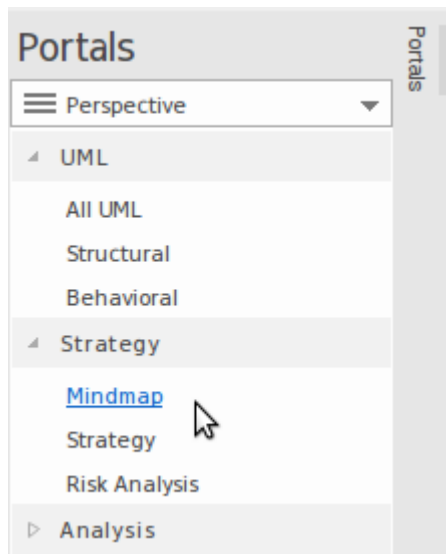
L'ingénieur Exigences est chargé de la tâche difficile de recueillir les exigences, ce qui nécessite une excellente communication avec les parties prenantes, y compris le client et l'équipe d'analyse. Une façon très efficace de faciliter la collecte des besoins des parties prenantes est d'exécuter un atelier avec toutes les parties prenantes clés présentes. Les compétences de l'ingénieur Exigences en tant que communicateur, diplomate et médiateur sont importantes pour créer un environnement collaboratif et respectueux propice à l'exploration des besoins et des préoccupations des parties prenantes. Il est impératif que l'ingénieur utilise une terminologie que les parties prenantes comprennent et qu'il fasse également preuve d'une compréhension ou d'une volonté d'apprendre les éléments qui composent le domaine de l'ingénierie.

On pense parfois à tort que ces ateliers sont l'occasion de formuler un ensemble d'exigences clairement définies pouvant être saisies dans l'outil sous forme Exigences des parties prenantes. Or, c'est loin d'être le cas. Les parties prenantes formulent généralement un large éventail d'idées, notamment des politiques, Règles Métier, des définitions de données, des contraintes Gestion de Projet, Exigences fonctionnelles, Exigences Métier, des problèmes de système existants et même des solutions suggérées. Même lorsqu'un consultant externe est utilisé pour exécuter ces réunions, l'ingénieur n'aura pas le temps de classer toutes ces déclarations au cours des réunions. Il faut donc trouver un moyen pour le rédacteur chargé de documenter les déclarations de les intégrer dans l'outil sans se soucier du type d'informations enregistrées. Les enregistrer dans l'outil plutôt que de les griffonner dans le carnet de l'ingénieur est une bonne pratique, car cela permet de les afficher pendant la réunion et aux parties prenantes de voir les commentaires des autres.

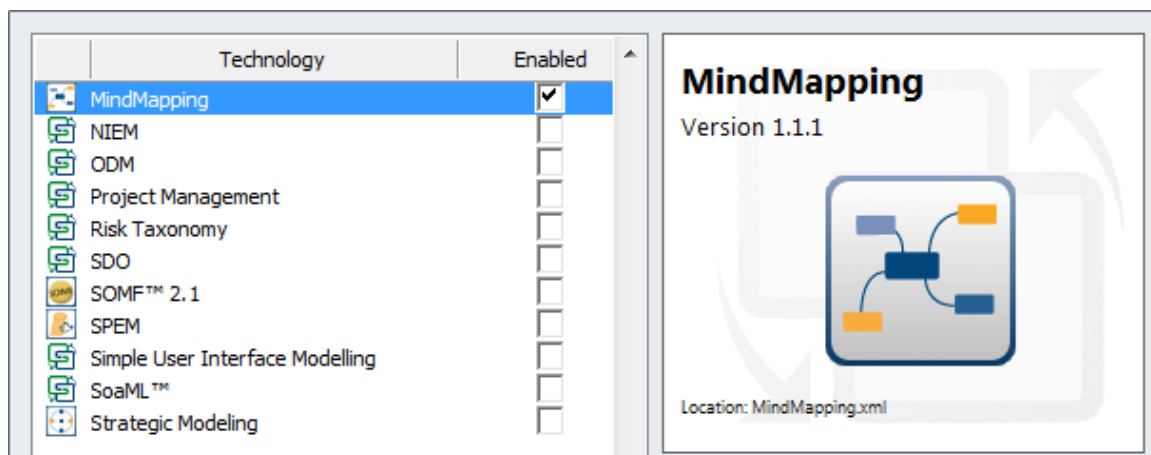
Enterprise Architect dispose de plusieurs facilités qui peuvent vous aider à organiser ces ateliers. Une méthode très pratique consiste à utiliser le diagramme Mind Mapping pour enregistrer les déclarations des parties prenantes, ce qui est très efficace car c'est une méthode bien connue et qui n'introduit aucune des formalités inhérentes aux langages modélisation tels que SysML. Ce diagramme montre une carte mentale de départ créée à partir d'un motif qui peut être modifié pour s'adapter aux besoins de l'atelier.



La facilité Mind Mapping est disponible en passant à cette perspective ou, si elle est couramment utilisée, elle peut être ajoutée à un ensemble de perspectives défini par l'utilisateur à l'aide de la facilité Mes Perspectives .



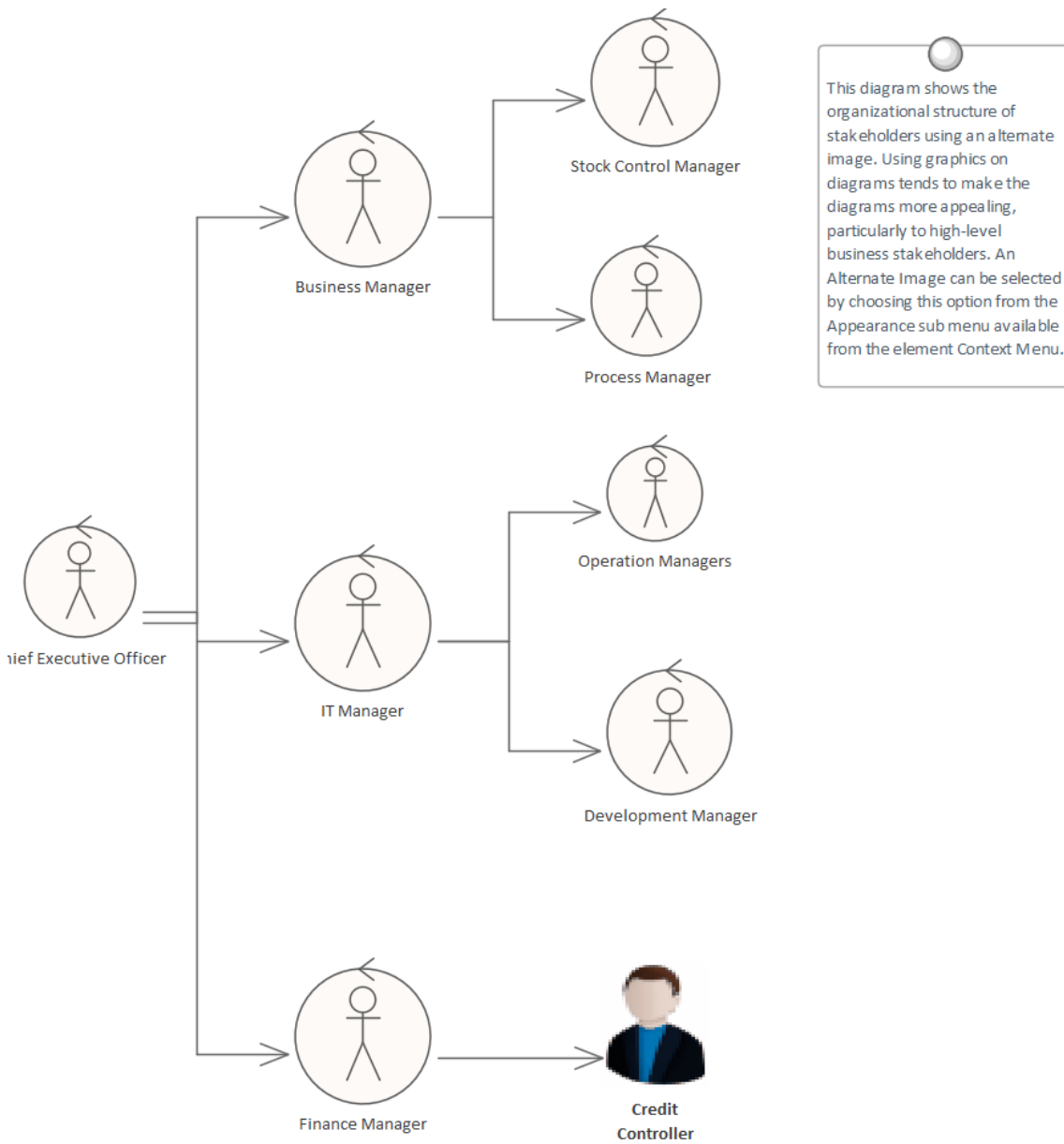
Cette perspective, comme d'autres, nécessite l'activation d'une technologie appropriée, qui dans ce cas est le Mind Mapping.



Au fur et à mesure que des termes importants sont découverts, ils peuvent être intégrés au Glossaire du Projet et, même s'il n'y a pas assez de temps pour discuter et débattre de la signification convenue, les mots serviront de liste initiale d'entités importantes dans le domaine. Alternativement, les termes peuvent être créés sous forme de blocs dans un diagramme de définition Bloc et reliés les uns aux autres par des connecteurs qui décrivent les relations importantes entre les termes.

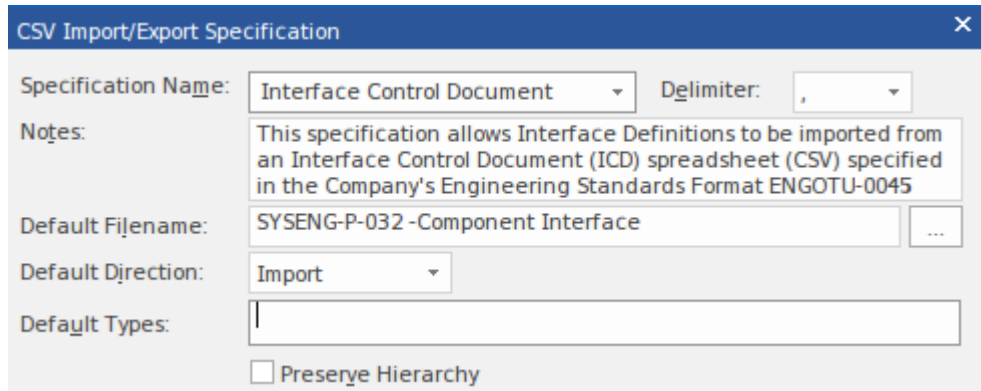
Les parties prenantes peuvent également être modélisées et leurs relations organisationnelles entre elles peuvent être décrites dans un diagramme. Il s'agit d'une technique utile qui permet aux principales parties prenantes de s'identifier dans les modèles, ce qui crée une adhésion.





## Créer Exigences

Enterprise Architect offre support complet pour le développement Exigences et fournit un certain nombre d'outils spécialisés à cet effet. Comme pour tout contenu de modèle, il est recommandé au modélisateur de vérifier si les Exigences ont été saisies dans le référentiel par quelqu'un d'autre avant de se lancer dans la tâche de création de nouvelles Exigences . Il est également possible que les Exigences aient été définies dans un autre outil tel qu'une feuille de calcul et puissent être importées dans Enterprise Architect sans qu'il soit nécessaire de créer chaque exigence manuellement.



The screenshot shows a dialog box titled "CSV Import/Export Specification" with a close button (X) in the top right corner. The dialog contains the following fields and options:

- Specification Name:** A dropdown menu set to "Interface Control Document".
- Delimiter:** A dropdown menu set to ",".
- Notes:** A text area containing the text: "This specification allows Interface Definitions to be imported from an Interface Control Document (ICD) spreadsheet (CSV) specified in the Company's Engineering Standards Format ENGOTU-0045".
- Default Filename:** A text field containing "SYSENG-P-032 -Component Interface" and a browse button (...).
- Default Direction:** A dropdown menu set to "Import".
- Default Types:** An empty text field.
- Preserve Hierarchy:** An unchecked checkbox.

Enterprise Architect dispose de deux emplacements pour Exigences ; elles peuvent être créées dans le modèle en tant qu'élément qui apparaîtra dans la fenêtre Navigateur , ou elles peuvent être créées à l'intérieur d'un autre élément en tant qu'exigence interne ou responsabilité.

## Exigences externes et internes

Enterprise Architect peut support tout type de processus d'exigences et permet de définir Exigences en tant qu'éléments du modèle. Celles-ci sont appelées Exigences externes, mais l'outil permet également de définir Exigences pour un élément spécifique, appelées Exigences internes. Un ingénieur qui souhaite définir une exigence utilisateur telle que :

*Le système doit permettre la mise à jour des horaires de bus.*

utiliserait une exigence externe. Un modélisateur souhaitant décrire le comportement d'un composant utiliserait une exigence interne pour le composant, telle que :

*L'éditeur doit support Unicode .*

Il y a souvent des conflits entre les analystes et les développeurs quant à savoir si une exigence doit être interne ou externe, et Enterprise Architect fournit un facilité de déplacer Exigences internes vers l'extérieur de l'élément.

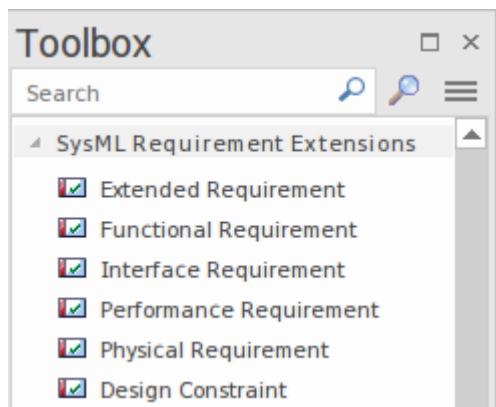
Lorsqu'elles sont déplacées, elles restent liées à l'élément d'origine.

## Catégories d'exigences

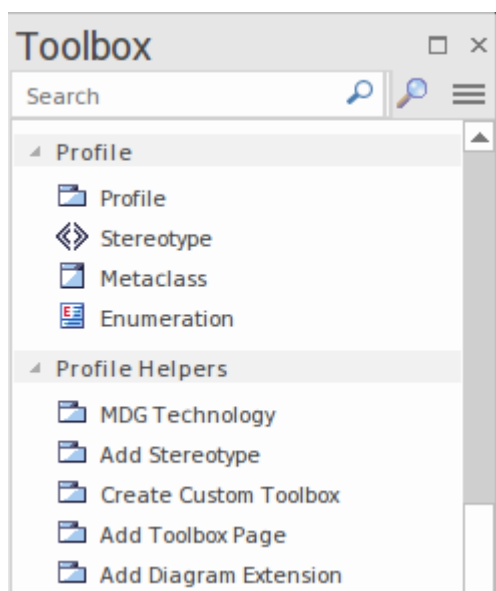
La spécification SysML fournit une liste non normative de catégories d'exigences (types). Il s'agit d'exigences stéréotypées qui affinent ou étendent l'exigence SysML de base, fournissant un mécanisme pour créer des exigences qui servent un objectif particulier ou décrivent un aspect particulier d'un système. Par exemple, les exigences physiques peuvent être utilisées pour décrire un aspect physique d'un système tel que le poids ou la taille d'un composant. Ces catégories et d'autres catégories créées par l'utilisateur peuvent avoir un nombre quelconque de propriétés supplémentaires définies telles que :

- RisqueKind
- Méthode de vérificationType

Enterprise Architect fournit de manière pratique ces catégories d'exigences sous forme d'éléments sur les pages de la boîte à outils des exigences SysML.



L'outil fournit également un système de profil sophistiqué et entièrement fonctionnel, permettant aux utilisateurs de créer des extensions de l'exigence SysML de base et de n'importe quel nombre de catégories d'exigences définies par l'utilisateur applicables au domaine de modélisation ou à l'espace de problème. Ces exigences stéréotypées peuvent être associées à des propriétés définies par l'utilisateur qui sont nécessaires pour modéliser l'élément d'exigence spécifique (ou un autre élément de modèle).

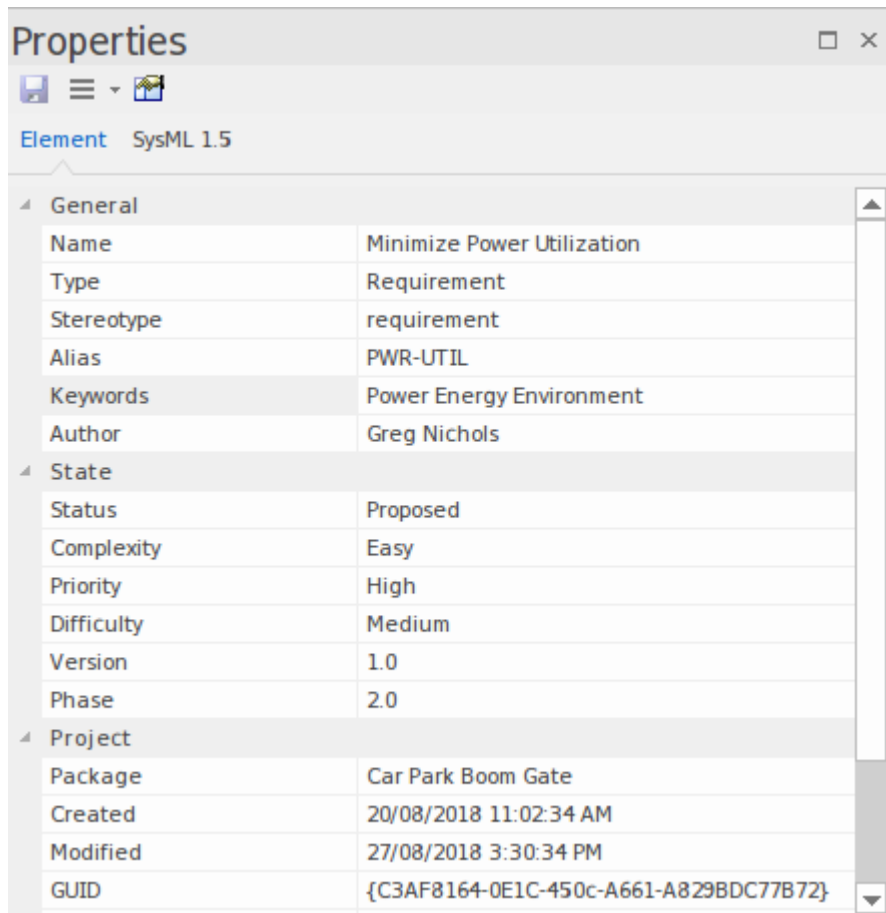


Par exemple, une équipe peut décider d'inclure une propriété de volatilité à une exigence pour garantir que le travail ne commence pas tant que l'exigence n'est pas stable (c'est-à-dire non volatile). Autre exemple : une équipe peut créer des dispositifs médicaux et avoir besoin de se conformer à diverses normes réglementaires. Chaque composant utilisé dans le

cadre de la solution peut être tenu d'être conforme. Une propriété de niveau de conformité peut être créée et le composant peut se voir attribuer un niveau indiquant sa conformité à partir d'une plage de valeurs définies dans un contrôle rotatif ou une liste déroulante. Pour plus d'informations, consultez la rubrique d'aide [Developing Profiles](#) .

## Propriétés requises

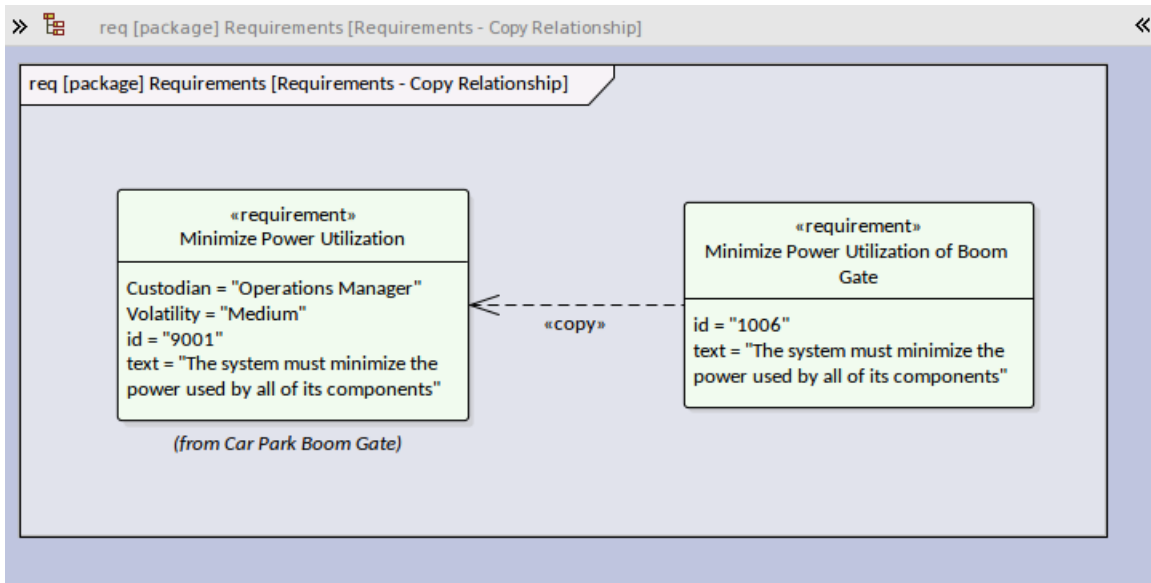
Le développement et la gestion des exigences sont essentiels à la réussite de tout projet, et les propriétés des Exigences sont importantes pour la priorisation et la manière dont elles seront élaborées et utilisées au sein d'une équipe de mise en œuvre ou de développement. Tous les éléments Enterprise Architect ont des propriétés standard telles que Statut, Auteur et Phase, mais l'élément Exigence a des propriétés supplémentaires telles que Difficulté et Priorité.



The screenshot shows a 'Properties' window for a SysML 1.5 element. The window is divided into three sections: General, State, and Project. The General section includes Name, Type, Stereotype, Alias, Keywords, and Author. The State section includes Status, Complexity, Priority, Difficulty, Version, and Phase. The Project section includes Package, Created, Modified, and GUID.

General	
Name	Minimize Power Utilization
Type	Requirement
Stereotype	requirement
Alias	PWR-UTIL
Keywords	Power Energy Environment
Author	Greg Nichols
State	
Status	Proposed
Complexity	Easy
Priority	High
Difficulty	Medium
Version	1.0
Phase	2.0
Project	
Package	Car Park Boom Gate
Created	20/08/2018 11:02:34 AM
Modified	27/08/2018 3:30:34 PM
GUID	{C3AF8164-0E1C-450c-A661-A829BDC77B72}

Certains processus d'exigences prescrivent des propriétés spécifiques telles que *le dépositaire* et *la volatilité (stabilité)* et celles-ci peuvent être configurées à l'aide Valeur Étiquetés qui peuvent être appliquées à chaque exigence. Le champ « Notes » d'une exigence a une signification particulière car il contient souvent une description formelle et contractuelle de la manière dont le système doit se comporter ou fonctionner. Pour plus d'informations, consultez la rubrique d'aide [Element Property Displays](#) .



# Spécification

La spécification Exigences est un aspect important de l'évolution d'une exigence. Elle fournit un catalogue important d'énoncés sur le comportement du système dans des conditions normales et anormales. Les Exigences intéresseront un large éventail de parties prenantes, notamment :

- Responsables de l'ingénierie
- Architectes
- Créateurs
- Les clients ou leurs substituts
- Ingénieurs système
- Ingénieurs en logiciel
- Testeurs
- Responsables de la conformité
- Ingénieurs Qualité
- Ingénieurs de sécurité

Tous ces groupes auront à la fois une contribution aux Exigences et la nécessité d'utiliser le catalogue des Exigences dans leur travail. Il existe plusieurs façons de spécifier les exigences dans Enterprise Architect , notamment :

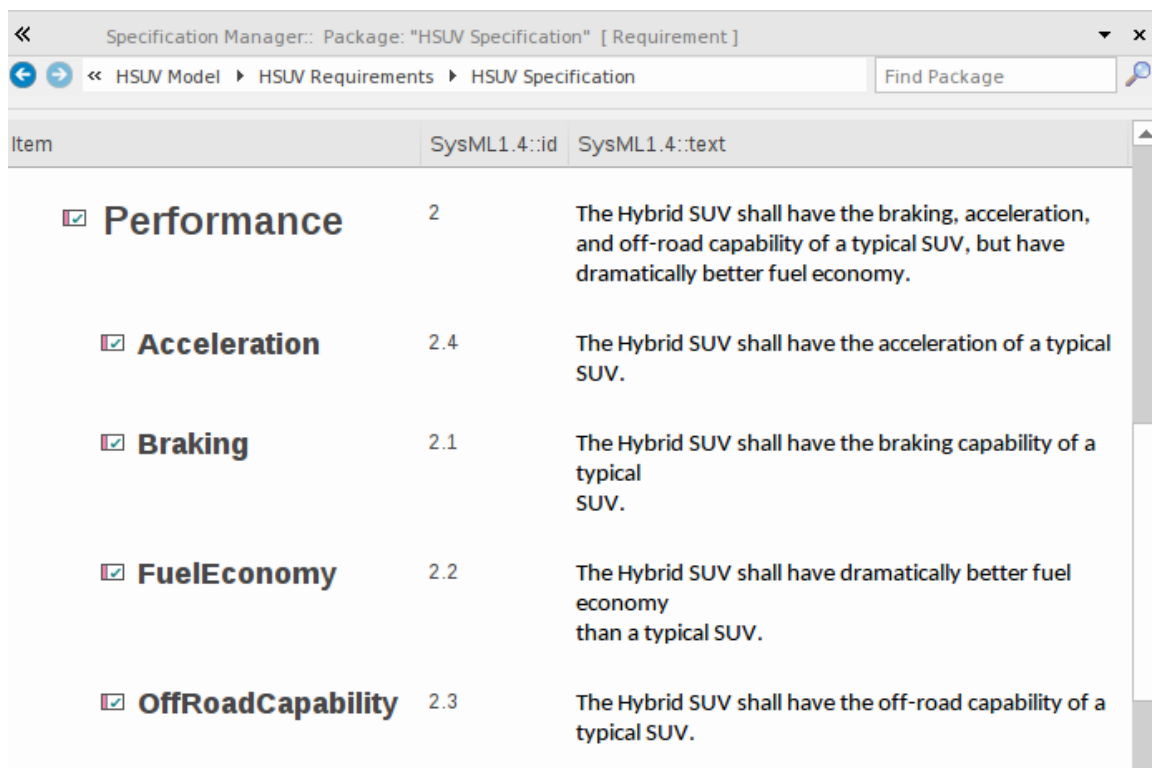
- Directement dans la fenêtre Navigateur
- Sur un diagramme
- Utilisation du Gestionnaire de Spécification

Nous examinerons le Gestionnaire de Spécification dans la section suivante, et vous verrez qu'il offre une grande flexibilité lorsque vous travaillez avec Exigences et d'autres éléments avec du contenu textuel.



## Rencontrez le Gestionnaire de Spécification

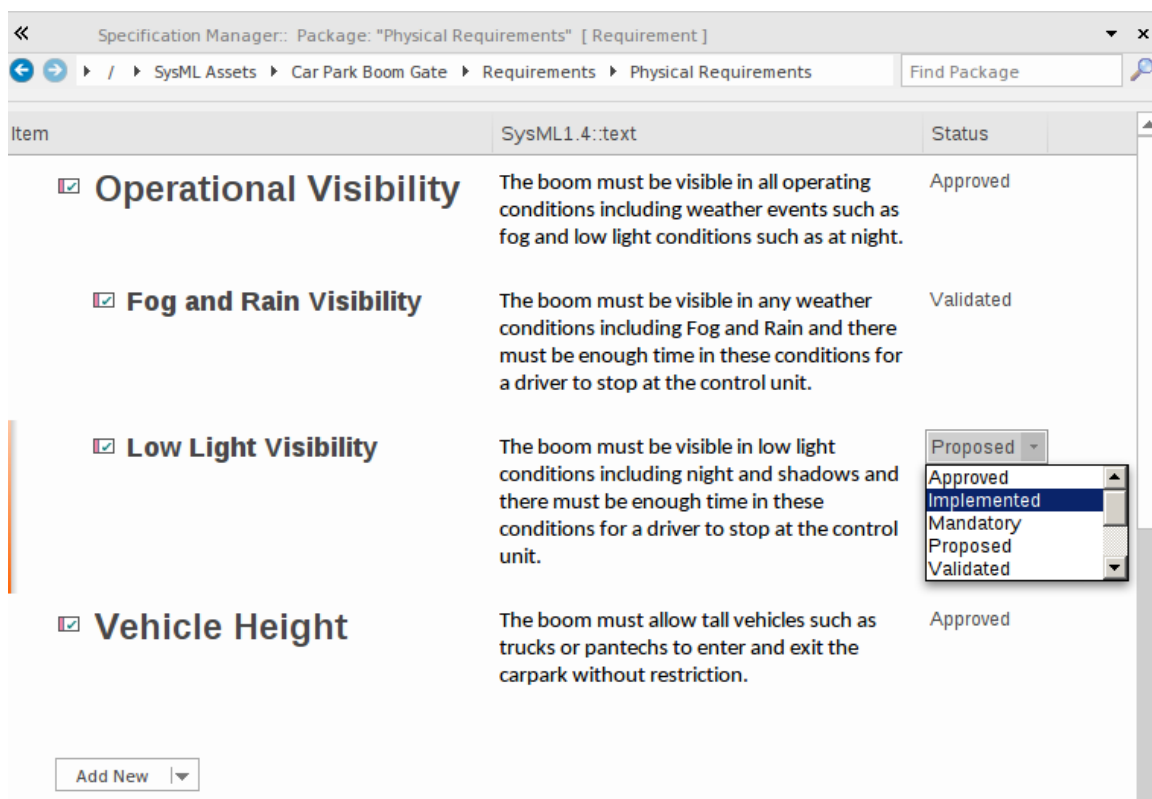
Le Gestionnaire de Spécification est un outil unique et efficace qui fournit une vue de feuille de calcul ou de traitement de texte qui peut être utilisée pour gérer n'importe quel élément, bien qu'il soit particulièrement utile lorsque vous travaillez avec Exigences qui ont toujours un texte descriptif pour décrire l'exigence en détail. De nouvelles Exigences peuvent être créées avec des noms et des descriptions détaillées, et des propriétés telles que le statut et la priorité peuvent être ajoutées ou modifiées à partir de listes déroulantes. Exigences existantes peuvent être visualisées et gérées dans une vue pratique - comme diagrammes et des fenêtres - et leur modification dans le Gestionnaire de Spécification les modifiera dans tous les autres emplacements du référentiel.



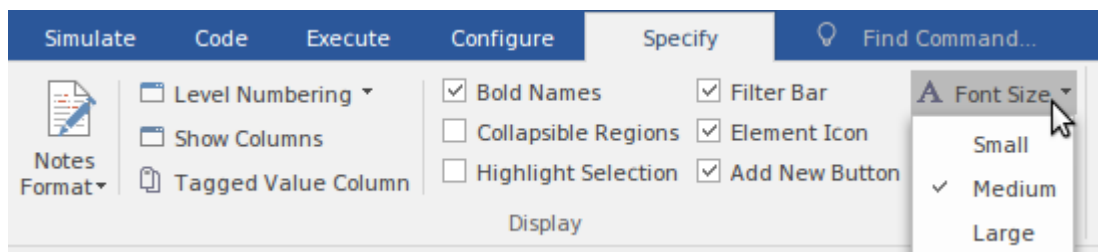
The screenshot shows the 'Specification Manager' window for a package named 'HSUV Specification'. The window title is 'Specification Manager: Package: "HSUV Specification" [ Requirement ]'. The breadcrumb navigation shows 'HSUV Model > HSUV Requirements > HSUV Specification'. There is a search bar labeled 'Find Package'. The main content is a table with the following columns: 'Item', 'SysML1.4::id', and 'SysML1.4::text'. The table contains five rows of requirements, each with a checked checkbox in the 'Item' column.

Item	SysML1.4::id	SysML1.4::text
<input checked="" type="checkbox"/> Performance	2	The Hybrid SUV shall have the braking, acceleration, and off-road capability of a typical SUV, but have dramatically better fuel economy.
<input checked="" type="checkbox"/> Acceleration	2.4	The Hybrid SUV shall have the acceleration of a typical SUV.
<input checked="" type="checkbox"/> Braking	2.1	The Hybrid SUV shall have the braking capability of a typical SUV.
<input checked="" type="checkbox"/> FuelEconomy	2.2	The Hybrid SUV shall have dramatically better fuel economy than a typical SUV.
<input checked="" type="checkbox"/> OffRoadCapability	2.3	The Hybrid SUV shall have the off-road capability of a typical SUV.

Le Gestionnaire de Spécification est l'outil idéal pour les analystes qui préfèrent travailler avec du texte plutôt qu'avec diagrammes et qui sont habitués à travailler dans un traitement de texte ou un tableur. Il présente l'avantage supplémentaire que les Exigences font partie d'un modèle et peuvent être reliées à d'autres éléments, notamment les moteurs Métier, les parties prenantes et les composants de la solution. Dans cette image, on peut voir que le statut des exigences et d'autres propriétés des éléments peuvent être modifiés à partir de listes déroulantes.



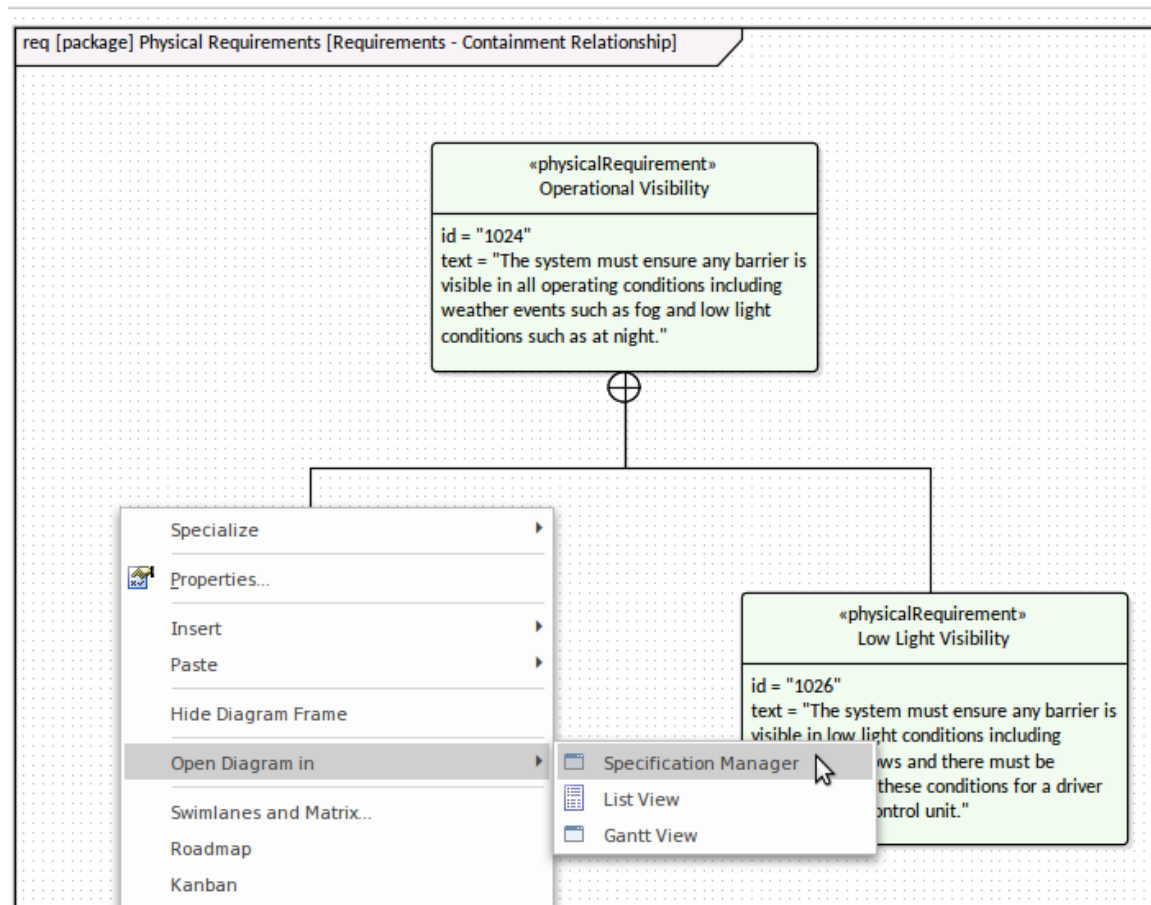
Il existe une large gamme d'options qui offrent une grande flexibilité lors de l'utilisation du Gestionnaire de Spécification, notamment l'affichage notes en colonnes comme dans une feuille de calcul ou en ligne comme dans un document, et le réglage de la taille du texte. Ces options sont disponibles à partir du ruban « Spécification - Spécifier », qui s'affiche de manière conditionnelle au lancement du Gestionnaire de Spécification.



Les filtres offrent un moyen utile de restreindre l'affichage aux éléments qui contiennent un mot ou un fragment de texte dans une colonne sélectionnée. Dans cette illustration, un modélisateur a décidé de restreindre l'affichage à toutes Exigences qui contiennent le mot « light » dans le texte de l'exigence. Il s'agit d'un excellent outil de productivité lorsque vous travaillez avec de grands ensembles d'Exigences. Il peut être utilisé pour localiser toutes Exigences ayant un statut, une priorité, une complexité particuliers ou même toutes Exigences appartenant à une partie prenante ou à une équipe spécifiée, si elles ont été définies dans le modèle.

Item	SysML1.4::text	Status
<input checked="" type="checkbox"/> <b>Operational Visibility</b>	The system must ensure any barrier is visible in all operating conditions including weather events such as fog and low light conditions such as at night.	Approved
<input checked="" type="checkbox"/> <b>Low Light Visibility</b>	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.	Implemented

Un diagramme peut également être ouvert à partir du Gestionnaire de Spécification, ce qui vous permet de modifier les éléments du diagramme en tant que groupe. Il s'agit d'une vue intéressante et appréciée par certains personnels non techniques, notamment les responsables et les clients. Pour plus d'informations, consultez la rubrique d'aide *Gestionnaire de Spécification*.



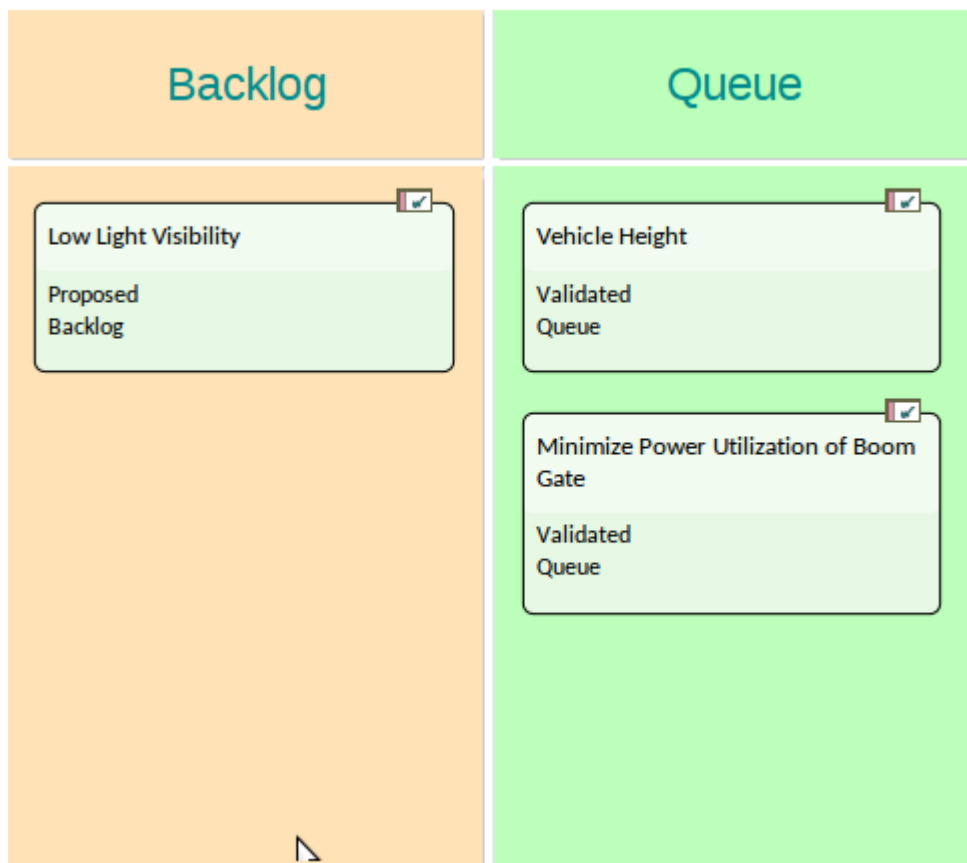
# Analyse

La phase d'analyse du développement des exigences permet de s'assurer que les exigences découvertes lors de la phase d'élicitation ont été correctement formulées et qu'elles ont le format, le niveau de détail et les propriétés appropriés, et qu'elles forment un ensemble correct et cohérent. En raison des sources et des méthodes d'élicitation disparates, les exigences enregistrées lors de la phase d'élicitation devront être remaniées et équilibrées. Il est par exemple assez courant de trouver des exigences répétées ou qui se chevauchent, ou qu'un ingénieur système oublie par inadvertance d'enregistrer les préoccupations d'une ou plusieurs parties prenantes. Des outils tels que la Matrice des relations et la fenêtre de traçabilité aideront à révéler les omissions et les problèmes liés aux exigences. Les fenêtres Discuss & Révision et Chat & Mail - y compris le Mail de Modèle facilité - fourniront également des mécanismes pour discuter, réviser et discuter des Exigences avec d'autres membres de l'équipe.

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
+ Source												
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

## Prioriser les Exigences

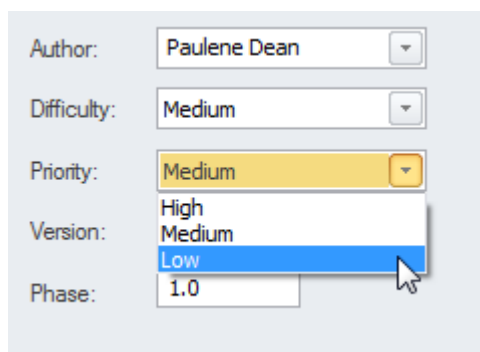
La priorisation Exigences est impérative pour la réussite d'un projet, car elle garantit que les ressources d'analyse, de développement, de test et de mise en œuvre sont concentrées sur les aspects les plus critiques du système. La priorisation est un processus de décision qui attribue une priorité à chaque exigence ; le critère le plus courant pour la catégorisation est valeur métier. valeur Métier est généralement déterminée par l'analyse coûts-avantages de la valeur que l'exigence mise en œuvre produira pour l'organisation ou ses clients. D'autres facteurs peuvent être la conformité aux politiques ou aux réglementations, l'urgence, le risque commercial ou technique et la probabilité de réussite. Exigences peuvent être visualisées dans un tableau Kanban qui peut être utilisé pour indiquer la priorité en déplaçant les éléments d'une voie de backlog vers une voie de file d'attente et en permettant également de classer les éléments dans les voies. Pour plus d'informations, consultez la rubrique d'aide [Kanban Boards](#) .



Alternativement, des recherches ou Modèle Vues pourraient être utilisés pour créer une liste d'exigences en fonction de certains critères qui permettraient de hiérarchiser les Exigences .

### Exigence Propriété prioritaire

Il existe une large gamme de critères pouvant être utilisés pour la priorisation, et chaque organisation et projet utilise généralement un type de moyenne pondérée pour déterminer la priorité. Enterprise Architect offre support flexible et complète de la priorisation des exigences, car chaque élément dispose d'une propriété « Priorité » intégrée qui peut être définie pour indiquer sa priorité, ce qui permet à l'utilisateur de sélectionner la priorité alloué dans une liste déroulante.



La liste des priorités est préchargée de manière pratique lorsque vous installez Enterprise Architect , mais celles-ci peuvent être modifiées ou entièrement révisées pour s'adapter à une organisation ou à un projet. Elles peuvent même être importées comme données de référence à partir d'un projet précédent ou, si le projet actuel a été créé sur la base d'un gabarit , les priorités de l'organisation peuvent être préchargées à partir du modèle de base. Elles peuvent être configurées à l'aide de cette option du ruban :

Paramètres > Données de référence > Types Modèle > Types généraux > Priorité

### Changer la priorité de manière collaborative

Le processus de sélection des critères et d'attribution de priorité est généralement collaboratif et se déroule souvent dans le cadre d'un atelier avec les parties prenantes ou leurs représentants qui débattent de la catégorisation. Autrefois, ce processus était laborieux et difficile, mais Enterprise Architect dispose de fonctionnalités utiles pour travailler avec les propriétés des exigences, notamment la priorité. Il existe un certain nombre de fenêtres, notamment la liste Paquetage et la liste Diagramme , qui support de travailler avec les Exigences et de modifier la priorité en ligne, en filtrant ou en triant automatiquement la liste des Exigences en fonction de la priorité nouvellement attribuée. Le Gestionnaire de Spécification est un outil utile à cet effet, fournissant une interface textuelle dans laquelle les Exigences et leurs notes peuvent être visualisées et les priorités peuvent être sélectionnées dans une liste déroulante. L'interface affiche également un certain nombre d'autres propriétés généralement utiles pour la priorisation, telles que le statut et la complexité. Pour plus d'informations, consultez la rubrique d'aide [Editing Elements](#) .

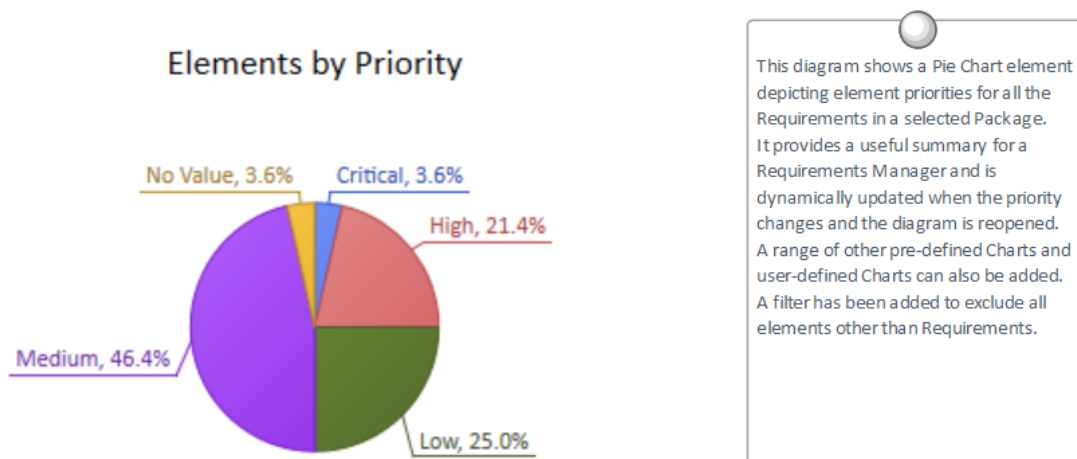
Requirement	Priority	SysML1.4::text	Stereotype	Status	Difficulty
<input checked="" type="checkbox"/> <b>Illumination</b>	Medium	The system must use strip lighting for illuminating the boom.	requirement	Proposed	Low
<input checked="" type="checkbox"/> <b>Minimize Power Utilization of Boom Gate</b>	Low	The system must minimize the power used by all of its components	requirement	Proposed	High
<input checked="" type="checkbox"/> <b>Operational Visibility</b>	High	The system must ensure any barrier is visible in all operating conditions including weather events such as fog and low light conditions such as at night.	requirement	Approved	Medium
<input checked="" type="checkbox"/> <b>Fog and Rain Visibility</b>	High	The system must ensure any barrier is visible in any weather conditions including Fog and Rain and there must be enough time in these conditions for a driver to stop at the control unit.	requirement	Validated	High
<input checked="" type="checkbox"/> <b>Low Light Visibility</b>	Critical	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.	requirement	Implemented	Low

Lorsqu'une propriété d'exigence est modifiée et enregistrée dans une fenêtre ou diagramme , la propriété sera modifiée

dans toutes les autres vues et tous les autres utilisateurs visualisant le référentiel pourront immédiatement voir la modification.

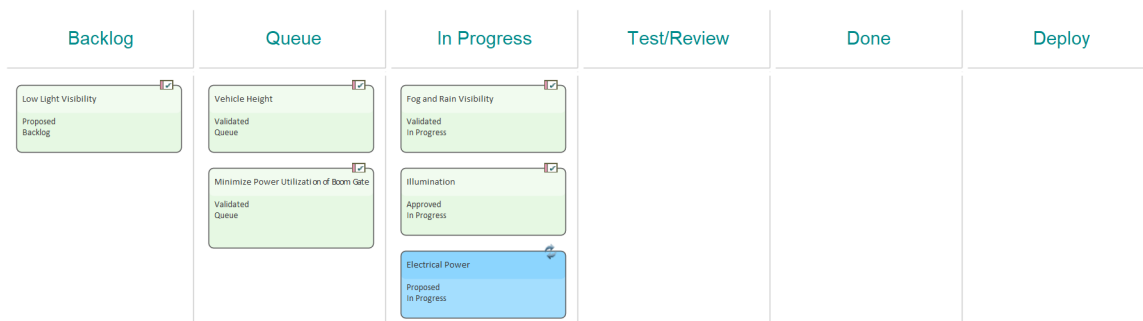
## Diagrammes du tableau de bord

Enterprise Architect propose une série de diagrammes de tableau de bord qui peuvent être utilisés pour créer une vue convaincante de la priorité des Exigences dans un Paquetage , avec la possibilité d'inclure des sous-packages. Il existe un certain nombre de graphiques préconfigurés qui peuvent être utilisés pour afficher le ratio des valeurs de priorité pour Exigences dans une partie du modèle. Les filtres ajoutent un autre niveau de configuration utilisateur, permettant à un modélisateur, par exemple, d'exclure Exigences d'un statut particulier ou de s'assurer que seules Exigences de la phase en cours sont affichées. Pour plus d'informations, consultez la rubrique d'aide [Dashboard Diagrams](#) .



## Visualisation avec les tableaux Kanban

Enterprise Architect dispose d'un diagramme Kanban Board qui peut être utilisé pour gérer Exigences et d'autres éléments de spécification ou de gestion de projet tels que les changements. Le tableau Kanban est particulièrement utile pour gérer la priorité des Exigences et d'autres éléments. Les éléments peuvent simplement être glissés sur le diagramme , puis entre les colonnes, ce qui permet aux équipes de gérer et de visualiser la progression d'une exigence entre la spécification et la mise en œuvre.



Le diagramme Kanban peut être configuré de manière à ce que lorsqu'un élément est déplacé d'une colonne à l'autre, la priorité de l'élément soit automatiquement modifiée. Pour plus d'informations, consultez la rubrique d'aide [Kanban Boards](#) .

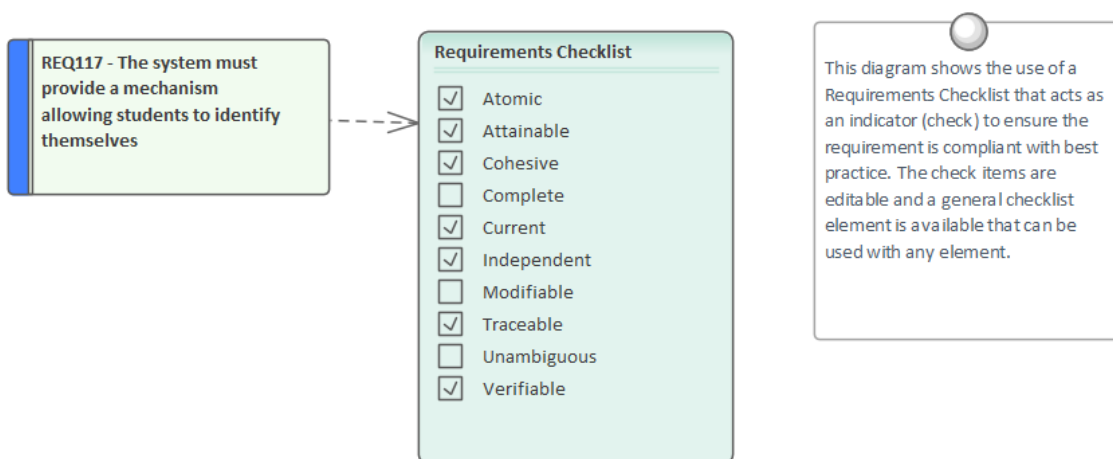




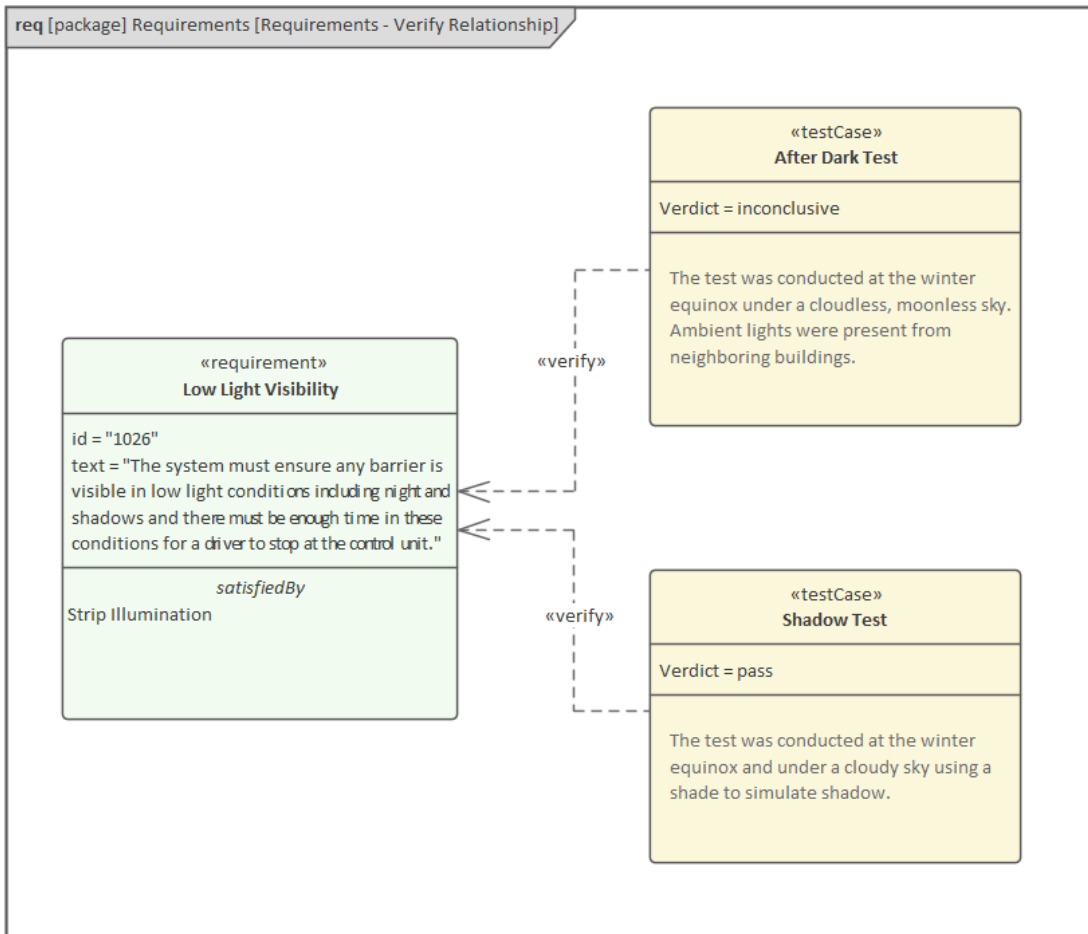
# Validation

La validation Exigences est nécessaire pour s'assurer que les Exigences sont de haut niveau, qu'elles définissent correctement le problème (ou l'opportunité) du client et qu'elles sont suffisantes pour que les équipes de mise en œuvre puissent concevoir et mettre en œuvre le produit. Il est impératif que les exigences aient le niveau de qualité souhaité et qu'elles soient complètes et nécessaires. Il existe plusieurs façons de valider Exigences , mais les deux méthodes les plus courantes sont probablement de procéder à révisions en équipe et d'assigner des cas de test aux exigences.

Les révisions d'équipe sont généralement menées par des membres de l'équipe ou d'autres analystes qui ont une certaine connaissance du domaine, mais qui ne sont pas eux-mêmes responsables du développement ou de la gestion des exigences. Enterprise Architect dispose d'un outil pratique pour faciliter ce processus, appelé Révision formelle, qui fonctionne sur l'ensemble du modèle et permet réviseurs d'enregistrer leurs conclusions dans des documents de discussion et de référencer des éléments du modèle. Il existe également un élément Liste de contrôle Exigences disponible sur la page « Exigences étendues » de la boîte à outils Exigences , qui fournit un mécanisme utile pour vérifier la qualité des Exigences .



Les cas Test peuvent être définis à plusieurs niveaux, depuis les tests d'acceptation utilisateur jusqu'aux tests unitaires. La définition des cas de test au début du processus de développement des exigences crée une double vérification des Exigences , car lorsque les cas de test sont définis, des problèmes liés aux Exigences sont souvent découverts. Enterprise Architect dispose d'un certain nombre de facilités pour définir les cas de test et un modélisateur peut sélectionner celui qui est le plus approprié à l'effort.

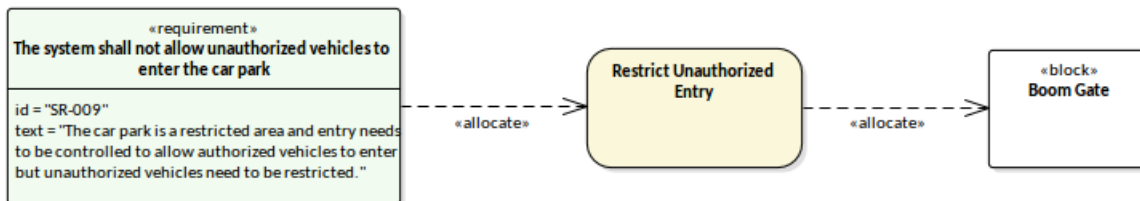


## Visualiser Exigences

Exigences peuvent être visualisées de différentes manières à l'aide de différentes fonctionnalités du produit. Certaines méthodes permettent à l'utilisateur de créer de nouvelles exigences et de modifier celles qui existent déjà, tandis que d'autres permettent simplement de visualiser les exigences. Exigences, comme tous les éléments du SysML, peuvent faire partie d'un graphe et de nombreuses sémantiques importantes sont exprimées dans ces relations ; par exemple, la relation entre une exigence et les cas de test qui la vérifient. Il est cependant assez courant que les analystes des exigences, les gestionnaires et les autres parties prenantes souhaitent visualiser les exigences indépendamment de toute relation à laquelle une exigence pourrait participer. Nous allons examiner un certain nombre de ces facilités maintenant et d'autres seront abordées dans la rubrique Visualisation Relations entre exigences.

## Diagrammes Exigences

Le diagramme Exigences peut être utilisé pour visualiser Exigences et leurs relations avec d'autres éléments, y compris d'autres Exigences. Vous pouvez bien sûr visualiser Exigences de plusieurs autres manières, mais pour de nombreuses parties prenantes, le diagramme Exigences sera plus attrayant car il fournit un moyen graphique de visualiser les connexions des Exigences avec d'autres parties importantes du modèle, y compris les parties prenantes, architecture, la conception et les tests. Ce diagramme montre comment Exigences peuvent être visualisées avec leur connexion à d'autres éléments. Une exigence stipulant que les véhicules non autorisés ne doivent pas être autorisés à accéder au parking est alloué à l'activité « Restreindre les véhicules non autorisés », qui à son tour est alloué à un Bloc représentant la barrière de sécurité.

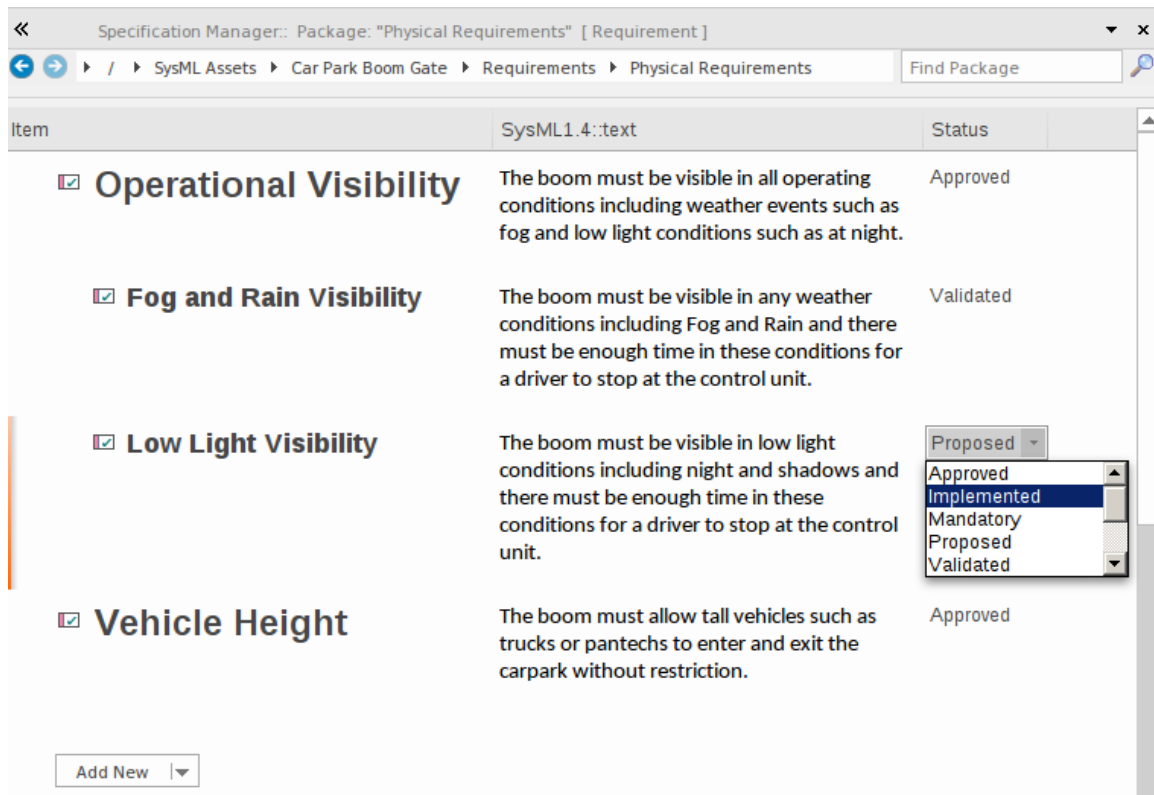


Un diagramme Exigences peut être créé à partir de différents emplacements dans l'interface utilisateur, y compris à partir de l'option de ruban « Conception > Diagramme > Ajouter Diagramme ».

La dialogue « Nouveau Diagramme » s'affiche et le diagramme Exigences peut être sélectionné dans la liste des types diagramme SysML. Pour plus d'informations, consultez la rubrique d'aide [SysML Requirements Modeling](#).

## Gestionnaire de Spécification

Le Gestionnaire de Spécification est l'outil central pour travailler avec Exigences dans Enterprise Architect . Il a été conçu dès le départ pour être un outil permettant de créer et de gérer Exigences via une interface intuitive et complète. Pour les ingénieurs ou autres intervenants habitués à travailler avec des feuilles de calcul ou des documents de traitement de texte, l'outil semblera naturel et émulerà ces deux modes de visualisation, permettant à l'utilisateur de basculer entre le mode feuille de calcul et le mode document.



Le Gestionnaire de Spécification peut être ouvert depuis le ruban : 'Design > Paquetage > Spécification Vue '.

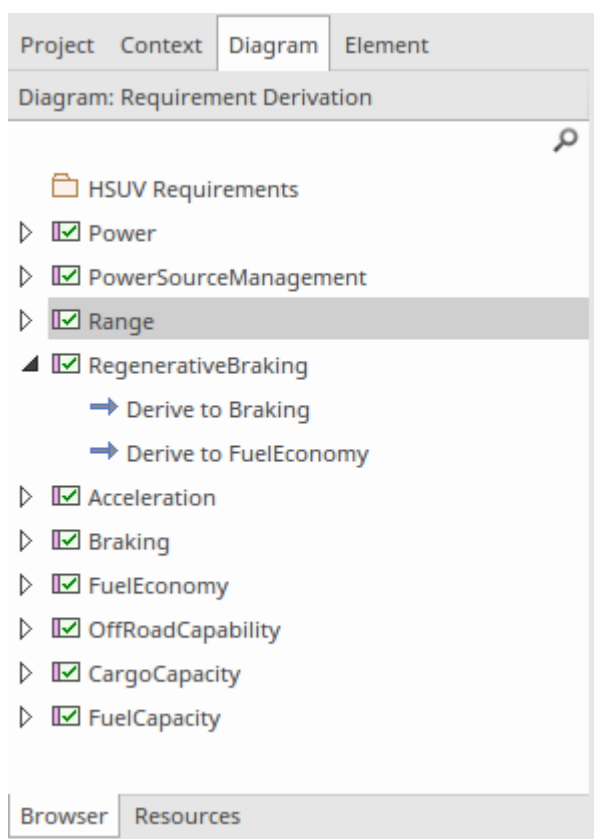
Le Gestionnaire de Spécification peut être utilisé pour visualiser tout type d'élément, mais il est particulièrement adapté aux éléments ayant une grande composante textuelle comme Exigences . Tous les éléments modifiés dans l'outil seront également automatiquement modifiés dans diagrammes Exigences , dans la fenêtre Navigateur et dans d'autres catalogues tels que tableaux . Pour plus d'informations, consultez le [Specification Manager](#) Rubrique d'aide.

## Navigateurs et Vues

Le Navigateur est l'outil de navigation central qui permet de structurer et d'explorer le contenu d'un référentiel, notamment en travaillant avec les exigences. Le Navigateur comporte un certain nombre d'onglets qui permettent de visualiser le contenu du référentiel de différentes manières. Nous avons déjà abordé le Navigateur dans une rubrique précédente, mais nous allons maintenant parler de sa pertinence pour la visualisation des exigences.

La plupart des ingénieurs système essaieront de conserver leurs exigences pour un projet ou une activité particulière dans un seul emplacement, même si dans certaines circonstances, elles doivent être séparées, par exemple pour des raisons contractuelles ou de calendrier. Une fois qu'un Paquetage Exigences a été sélectionné dans l'onglet « Projet », un utilisateur peut passer à l'onglet « Contexte » pour obtenir une vue ciblée, supprimant ainsi efficacement le bruit des autres éléments extérieurs à ce contexte. Une exigence individuelle peut alors être sélectionnée et l'onglet « Détails » sélectionné dans la fenêtre Inspecteur pour se concentrer sur les propriétés de l'exigence sélectionnée.

Les éléments et les relations contenus dans un diagramme ouvert peuvent également être visualisés via l'onglet ' Diagramme ' du Navigateur , offrant ainsi une manière alternative de visualiser le contenu d'un diagramme .

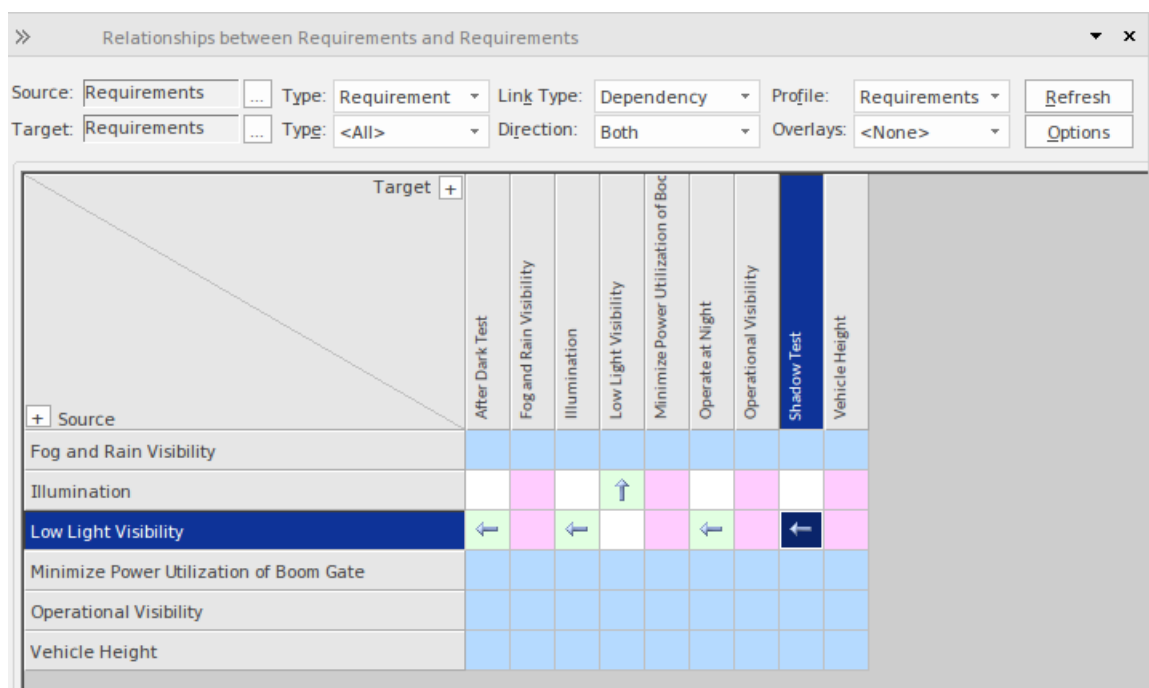


Tout Paquetage ou diagramme peut également être visualisé de différentes manières, y compris la vue de liste qui fournit une vue souvent appréciée des éléments d'une liste, similaire à une feuille de calcul où les exigences sont répertoriées en lignes et leurs propriétés sont répertoriées en colonnes.

Status				
▼ Status ▼	Name ▼	Type ▼	Modified ▼	
▲ Status: Approved				
<input checked="" type="checkbox"/> Approved	OffRoadCapability	Requirement	30/01/2020	
<input checked="" type="checkbox"/> Approved	FuelCapacity	Requirement	30/01/2020	
▲ Status: Implemented				
<input checked="" type="checkbox"/> Implemented	Braking	Requirement	30/01/2020	
▲ Status: Proposed				
<input checked="" type="checkbox"/> Proposed	Power	Requirement	27/03/2018	
<input checked="" type="checkbox"/> Proposed	Range	Requirement	27/03/2018	
<input checked="" type="checkbox"/> Proposed	CargoCapacity	Requirement	5/03/2019	
<input checked="" type="checkbox"/> Proposed	RegenerativeBraking	Requirement	27/03/2018	
<input checked="" type="checkbox"/> Proposed	PowerSourceManagement	Requirement	27/03/2018	

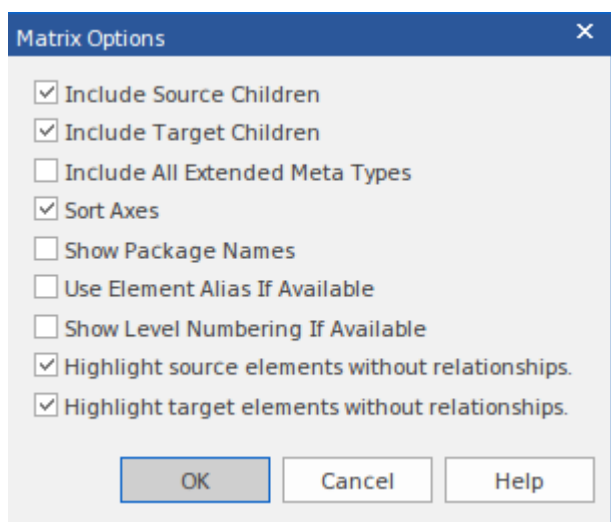
## Matrice de relations

La Matrice de relations est un outil précieux pour visualiser les connexions entre les éléments de deux Paquetages quelconques, dans une interface ressemblant à une feuille de calcul avec des lignes et des colonnes. L'outil est particulièrement utile lorsqu'il est utilisé avec Exigences et permet à un ingénieur de voir comment Exigences sont liées à d'autres éléments, y compris d'autres Exigences .



La Matrice de relations peut être ouverte à partir de l'option du ruban « Conception > Paquetage > Paquetage / Matrice ». Sélectionnez si le Paquetage actuel est le Paquetage source, Paquetage cible ou les deux.

Lorsqu'une relation existe, une icône en forme de flèche s'affiche dans la cellule à l'intersection des éléments source et cible, la pointe de la flèche indiquant la direction de la relation. La matrice peut également être configurée pour mettre en surbrillance les lignes ou les colonnes qui n'ont aucune relation, dans une couleur distincte. Cette option et d'autres peuvent être configurées dans la fenêtre Options (cliquez sur le bouton Options dans l'en-tête de la Matrice de relations).



Ces options vous permettent de personnaliser la manière dont la matrice est affichée, notamment si les éléments sont triés ou si leurs noms sont préfixés par le nom Paquetage , et si les lignes et colonnes des éléments source et cible sans connexions sont mises en surbrillance. Pour plus d'informations, consultez la rubrique d'aide [Relationship Matrix](#) .



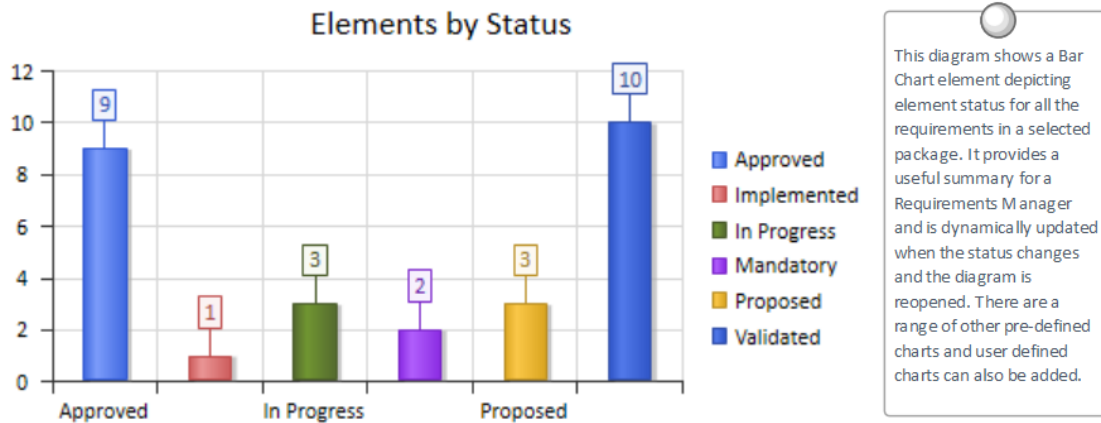
## Exigences Tableaux

Tableaux Exigences sont des vues ressemblant à des feuilles de calcul qui peuvent être créées à l'aide d'une instruction SQL pour sélectionner Exigences (ou tout autre élément) en fonction d'une instruction select, qui filtre efficacement un groupe particulier d' Exigences . Par exemple, un tableau peut être utilisé pour afficher toutes Exigences liées au sous-système d'alimentation qui sont approuvées et hautement prioritaires ou pour la décomposition des exigences de performances. N'importe quel nombre de tableaux peut être créé et ils sont actualisés de manière dynamique lorsque les propriétés sous-jacentes sont mises à jour dans le référentiel. Cela offre plus de flexibilité que la vue Liste en raison de la possibilité de sélectionner un groupe d' Exigences à partir de n'importe quel endroit du référentiel en fonction de critères spécifiés.

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		

## Gérer Exigences

Il s'agit des activités visant à maintenir un ensemble d'exigences qui représentent un accord entre l'équipe de projet et le client. Il s'agit également de s'assurer que les exigences sont acceptables pour les équipes de conception et de mise en œuvre et qu'elles sont suffisantes pour que ce qu'elles spécifient puisse être mis en œuvre dans des systèmes opérationnels, logiciels ou matériels. Enterprise Architect est une plate-forme sophistiquée de gestion des exigences et, quel que soit le domaine, la taille du projet ou la méthode suivie, il existe des outils qui simplifient la gestion de référentiels d'exigences, même volumineux, dans des projets complexes.

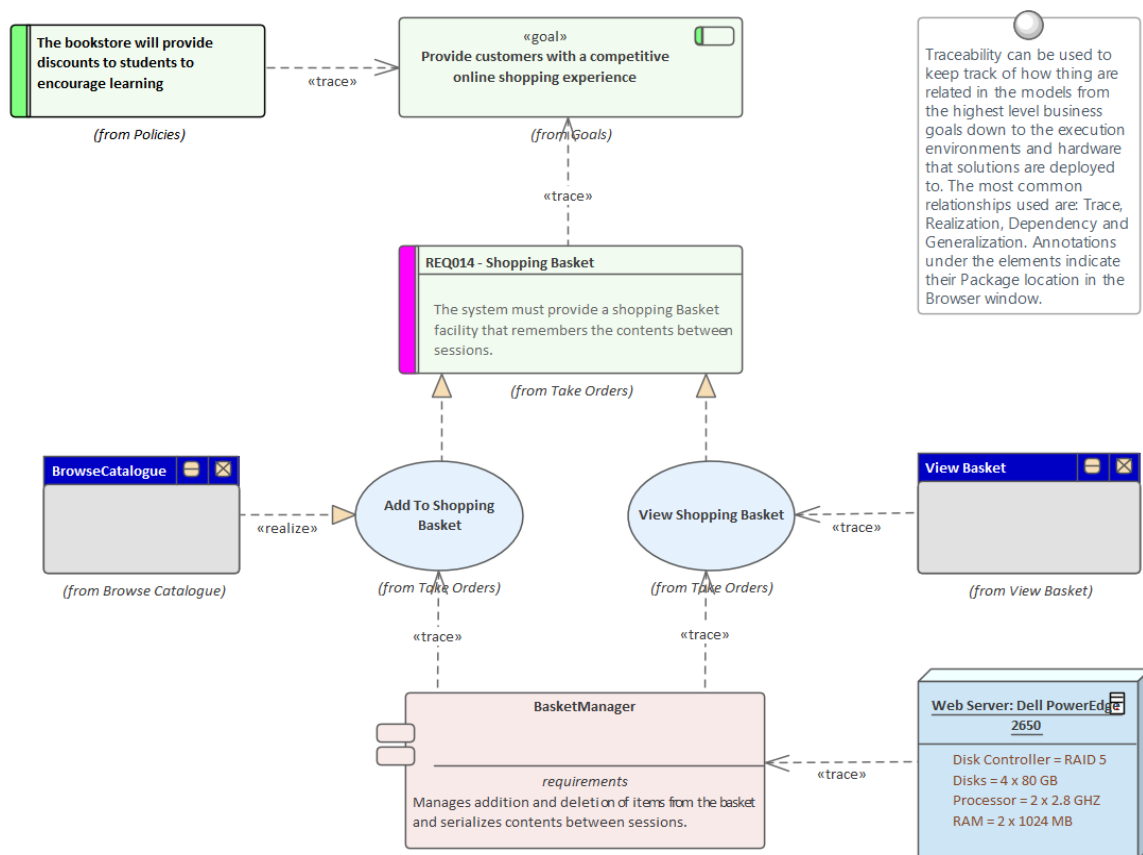


# Traçage des Exigences

La plupart des processus d'exigences exigent que Exigences soient tracées à partir de concepts de haut niveau tels que les moteurs Métier , les visions et les objectifs jusqu'aux parties des composants qui les mettent en œuvre. Pour de nombreux projets, il s'agit d'un problème insoluble car une grande partie des informations réside dans un ensemble d'outils hétérogènes tels que des documents de traitement de texte, des feuilles de calcul, des outils diagramme , des outils de présentation d'entreprise, etc. Certains Gestionnaires de Projet tentent de résoudre le problème en créant une feuille de calcul qui agit comme un registre de toutes les informations disparates, mais la gestion de ce fichier occupe des ressources de projet considérables et il est presque impossible de le maintenir à jour. Avec Enterprise Architect il est possible de modéliser toutes ces informations de projet dans un seul outil et de créer des traces faciles à maintenir et à analyser entre tous les éléments, de la déclaration de mission de l'organisation jusqu'au niveau du code de programmation, si nécessaire.

## Visualisation des traces dans diagrammes

Que vous ayez saisi Exigences du projet à l'aide d'un diagramme , d'un outil textuel tel que le Gestionnaire de Spécification ou en les important à partir d'un autre outil, la visualisation des traces d'exigences dans un diagramme permet de comprendre facilement leurs relations. Les diagrammes peuvent être créés facilement en faisant glisser et en déposant des éléments depuis la fenêtre Navigateur , ou automatiquement en utilisant l'option « Insérer des éléments liés ». Cette fonction peut être configurée et utilisée pour dessiner un graphique d'éléments à n'importe quelle profondeur, et peut être limitée à des types d'éléments et de connecteurs sélectionnés. C'est un outil de productivité pratique dans un environnement d'équipe, et même les modélisateurs ayant une connaissance approfondie du domaine et du référentiel sont surpris par les connexions qui sont affichées dans les diagrammes .



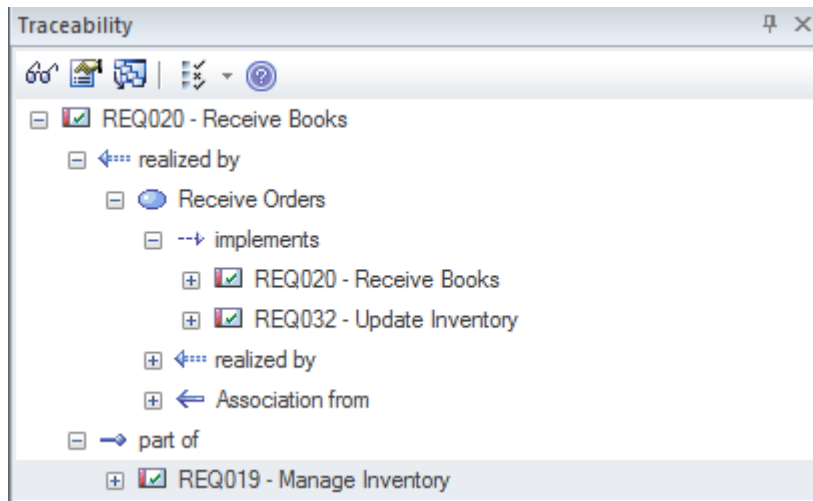
## Visualisation des traces à l'aide de la Matrice de relations

La Matrice de relations offre une autre façon de visualiser les relations entre Exigences et d'autres éléments, voire entre différents niveaux ou types d'exigences. Il est assez courant que certaines parties prenantes préfèrent une vue de type feuille de calcul des Exigences et de leurs relations, et la Matrice de relations offre un excellent moyen de présenter les relations sans recourir à un diagramme . Dans les méthodes d'exigences basées sur les cas d'utilisation, les cas d'utilisation sont censés réaliser une ou plusieurs Exigences , et ces relations peuvent être affichées visuellement dans la Matrice de relations. La liste des cas d'utilisation apparaît sur un axe de la matrice et les Exigences sont répertoriées sur l'autre axe. Un marqueur à l'intersection d'une ligne et d'une colonne s'affiche si une relation existe, indiquant qu'un cas d'utilisation particulier réalise une exigence. Relations entre les éléments peuvent être créées ou supprimées à l'aide de la Matrice de relations, et la Matrice peut être enregistrée et rouverte à tout moment ou enregistrée dans un fichier CSV afin de pouvoir être ouverte dans une feuille de calcul. Il est également possible de créer une documentation incluant la Matrice des relations, fournissant ainsi un outil de communication utile aux personnes qui n'ont pas accès au modèle.

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
+ Source												
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

### Visualisation des traces à l'aide de la fenêtre de traçabilité

Bien que diagrammes et la Matrice de relations permettent aux modélisateurs de visualiser les traces entre les éléments d'exigences, il est possible que les créateurs de ces vues du référentiel aient délibérément omis des éléments de la vue. Par exemple, un diagramme n'a pas besoin d'afficher toutes les exigences appartenant à une partie prenante particulière. La fenêtre Traçabilité présentera cependant une vue complète et intégrale des relations entre les éléments. Les relations entre les éléments seront affichées quel que soit l'emplacement des éléments dans la fenêtre Navigateur .



## Visualisation des traces à l'aide de la fenêtre Relations

Les modélisateurs choisissent souvent de masquer une ou plusieurs relations sur un diagramme afin de rendre le diagramme plus simple à comprendre ou de masquer des détails. La fenêtre Relations est une fenêtre utile à avoir ouverte car elle affiche toutes les relations qui existent entre les éléments du diagramme en indiquant si elles sont visibles ou masquées dans le diagramme .

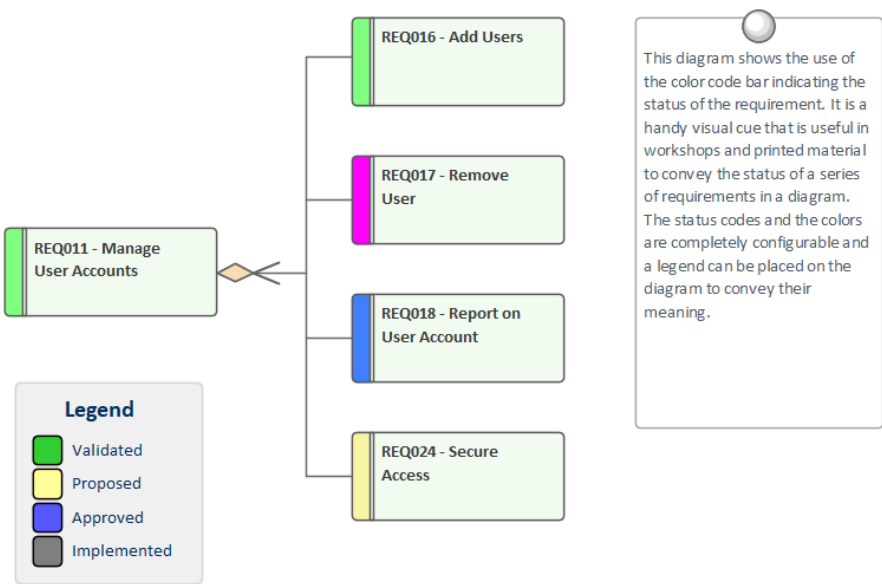
Relationships			
Relationship	Source	Target	View
Abstraction	REQ014 - Shopping Basket	Provide customers with a competitive ...	Visible
Realization	Add To Shopping Basket	REQ014 - Shopping Basket	Visible
Realization	View Shopping Basket	REQ014 - Shopping Basket	Hidden
Aggregation	REQ014 - Shopping Basket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

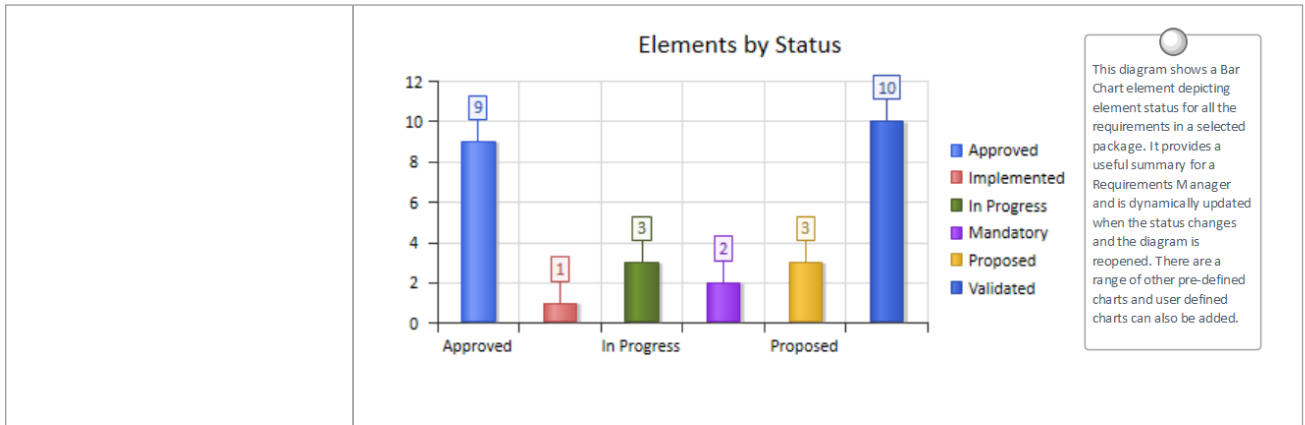
Si des relations ont été masquées dans un diagramme elles peuvent être rendues visibles en sélectionnant l'option « Afficher Relations » sur la page « Connecteurs » de la dialogue « Propriétés » du diagramme .

# Suivi Exigences

Le statut d'une exigence est un indicateur fondamental de son positionnement dans le processus de développement de l'exigence. Par exemple, les exigences dont le statut est « Proposé » indiquent qu'elles ne sont pas encore prêtes et disponibles pour le début des travaux de développement. Enterprise Architect dispose de divers outils permettant de suivre, d'analyser et de gérer le statut, à commencer par le fait que chaque exigence peut se voir attribuer un statut et que la liste des codes de statut est entièrement configurable. Le statut est affiché de manière pratique dans les vues de liste des exigences, y compris lors de l'utilisation du Gestionnaire de Spécification . Il existe également un ensemble de graphiques et de tableaux de bord prédéfinis et extensibles qui peuvent être utilisés pour obtenir une représentation visuelle convaincante du statut et d'autres propriétés des exigences.

## Outils de suivi des exigences

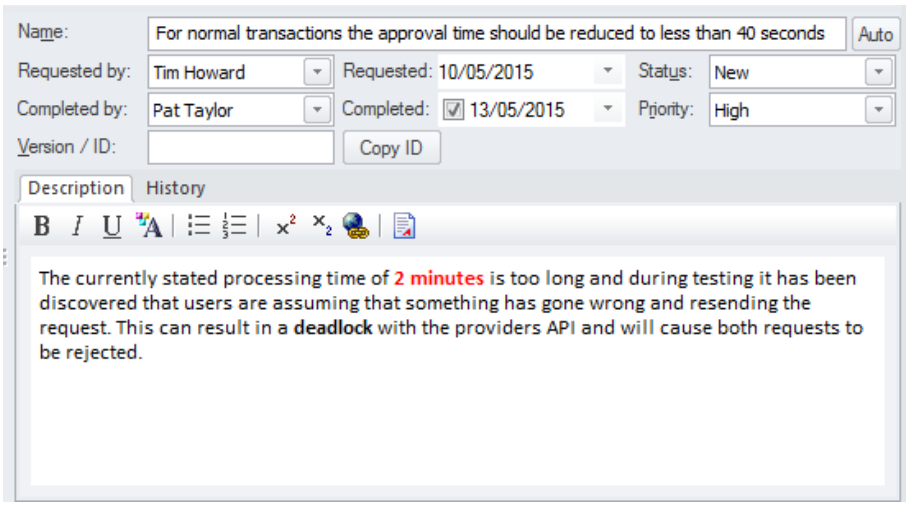
Outil	Description
<p>Codes d'état</p>	<p>Les codes d'état sont une liste contrôlée d'états qui peuvent être appliqués à n'importe quel élément, y compris Exigences . Enterprise Architect est fourni avec une liste prédéfinie de codes, mais la liste peut être configurée et les codes de la liste peuvent être modifiés et supprimés, et de nouveaux codes peuvent être ajoutés. L'état des Exigences peut être affiché dans un diagramme sous la forme d'une bande de couleur sur le côté de l'élément.</p>  <p>This diagram shows the use of the color code bar indicating the status of the requirement. It is a handy visual cue that is useful in workshops and printed material to convey the status of a series of requirements in a diagram. The status codes and the colors are completely configurable and a legend can be placed on the diagram to convey their meaning.</p>
<p>Tableaux de bord, graphiques et diagrammes</p>	<p>diagrammes de tableau de bord sont un type diagramme étendu et permettent de créer des graphiques et des diagrammes de haute qualité pour afficher les informations du référentiel de manière visuellement attrayante. Il est possible de créer un nombre illimité de diagrammes et de diagrammes et les données peuvent provenir de n'importe quel niveau de la hiérarchie Paquetage du référentiel. Enterprise Architect est fourni avec une page de boîte à outils de diagrammes et de graphiques préconfigurés, mais de nouveaux graphiques peuvent être créés à partir de n'importe quelle information du référentiel.</p>



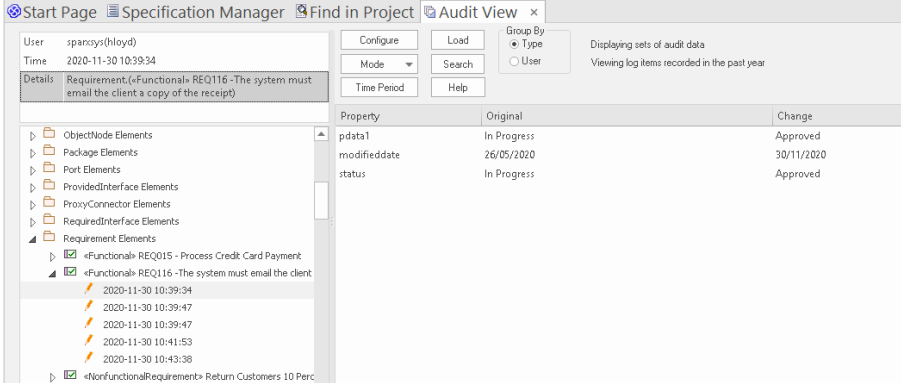
## Gérer Exigences changeantes

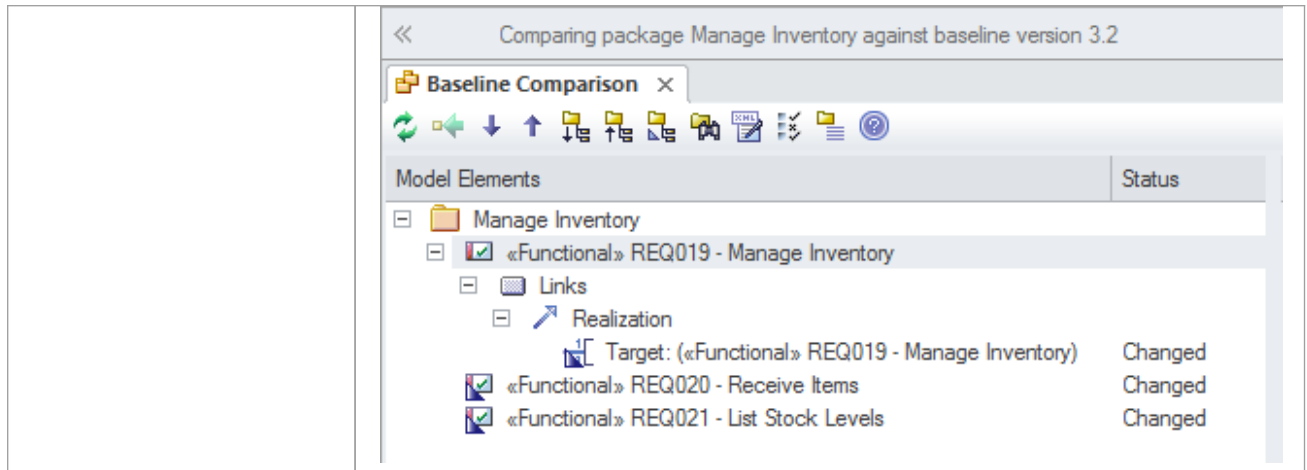
Il est inévitable que les exigences changent au cours des phases de spécification et de résolution d'un projet, et la plupart des processus de gestion des exigences disposent d'un certain type de mécanismes pour intégrer ces changements. En règle générale, un ensemble d'exigences aura été spécifié et préparé pour que les équipes de solution puissent le mettre en œuvre ; tout changement ultérieur est spécifié sous forme de demandes de changement. Quelle que soit la rigueur du processus utilisé, des modifications involontaires se produiront et devront être gérées en même temps que les demandes de changement. Enterprise Architect est une plate-forme de gestion des exigences sophistiquée, avec une gamme d'outils pour aider le gestionnaire des exigences. Les demandes de changement peuvent être gérées dans la fenêtre Maintenance, qui permet d'enregistrer et de décrire le changement demandé, ainsi que la personne qui l'a demandé, la date à laquelle il a été effectué et la personne qui a effectué le changement. Les modifications involontaires peuvent être découvertes et analysées à l'aide d'un certain nombre de fonctionnalités d'outils, notamment Auditing, Lignes de base et Contrôle de Version ; ces outils ont des fonctionnalités qui se chevauchent et peuvent être utilisés de manière isolée ou ensemble. Le système de sécurité intégré aidera également à empêcher les modifications involontaires des modèles, en permettant aux modélisateurs de verrouiller intentionnellement Paquetages et les éléments du modèle.

### Mécanismes de gestion des exigences changeantes

Mécanisme	Description
Éléments de tâche et d'effort de changement d'élément	<p>Les changements apportés aux exigences peuvent survenir par inadvertance, mais il est plus courant qu'il s'agisse d'un changement intentionnel en réponse à une grande variété de facteurs tels que la révision des besoins des parties prenantes, l'évolution de l'entreprise ou un problème mal compris. Les changements involontaires peuvent être détectés à l'aide de plusieurs outils, mais les changements délibérés peuvent être attribués à l'aide de l'élément Changement, qui peut être enregistré pour chaque élément. Une fois l'impact du changement analysé, des tâches peuvent être créées pour spécifier ce qui doit être fait pour mettre en œuvre le changement et des efforts peuvent être attribués à l'aide de l'élément Exigences Effort.</p> 
Audit	<p>L'audit est un outil intégré qui, lorsqu'il est activé, enregistre automatiquement les modifications apportées au référentiel. Il dispose de plusieurs modes et vues différents et peut être configuré pour faciliter la gestion des Exigences . Il peut suivre ce qui a été modifié dans le modèle, qui a effectué la modification et quand elle a été effectuée, en affichant les vues avant et après. Ainsi, si le texte d'une exigence a été mis à jour ou si son statut a été modifié, cela sera enregistré. La fonctionnalité d'audit chevauche celle de l'outil de référence, mais contrairement à l'outil de référence, les modifications sont enregistrées automatiquement et chaque</p>



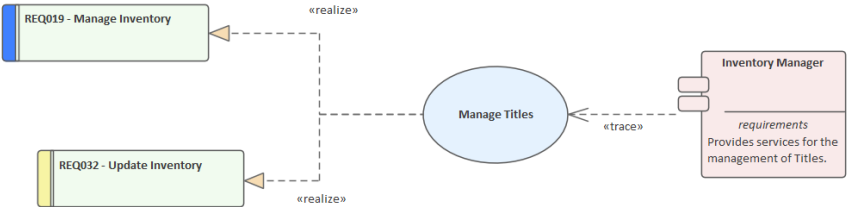
	<p>modification discrète est enregistrée. En revanche, l'outil de référence comparera uniquement le modèle actuel à une référence, quel que soit le nombre de modifications intermédiaires apportées. L'audit n'aidera pas à évaluer l'impact des modifications, mais seulement à déterminer les modifications qui se sont produites. Une fois les modifications établies, des outils tels que la Matrice de relations peuvent être utilisés pour déterminer l'impact.</p>  <table border="1" data-bbox="858 495 1423 577"> <thead> <tr> <th>Property</th> <th>Original</th> <th>Change</th> </tr> </thead> <tbody> <tr> <td>pdata1</td> <td>In Progress</td> <td>Approved</td> </tr> <tr> <td>modifieddate</td> <td>26/05/2020</td> <td>30/11/2020</td> </tr> <tr> <td>status</td> <td>In Progress</td> <td>Approved</td> </tr> </tbody> </table>	Property	Original	Change	pdata1	In Progress	Approved	modifieddate	26/05/2020	30/11/2020	status	In Progress	Approved
Property	Original	Change											
pdata1	In Progress	Approved											
modifieddate	26/05/2020	30/11/2020											
status	In Progress	Approved											
<p>Contrôle de Version</p>	<p>Contrôle de Version peut être implémenté dans Enterprise Architect pour gérer les modifications et les révisions de tout Paquetage , y compris Paquetages Exigences . Une fois implémentées, les modifications apportées aux Exigences seront enregistrées et un analyste des exigences pourra visualiser la version précédente et revenir à ces versions si nécessaire. Il existe un certain chevauchement entre cette fonctionnalité de l'outil et Auditing et Lignes de base . La différence entre cette facilité et Auditing est que Auditing enregistre simplement les modifications mais ne vous permet pas de revenir à une version précédente. La différence entre Contrôle de Version et Lignes de base est qu'un modélisateur doit intentionnellement créer une ligne de base alors qu'avec Contrôle de Version les modifications sont enregistrées automatiquement en arrière-plan. De plus, avec Lignes de base les modifications intermédiaires ne sont pas enregistrées, seule la différence entre l'exigence actuelle et celle capturée dans la ligne de base est enregistrée.</p>												
<p>Lignes de base</p>	<p>Lignes de base fournissent un mécanisme polyvalent pour gérer les modifications apportées aux Exigences . Il est possible de créer un nombre quelconque de lignes de base et, lorsque des exigences sont modifiées, ces exigences modifiées peuvent être comparées à l'une des lignes de base . Lignes de base sont généralement créées à des étapes importantes d'un projet, par exemple après une réunion des parties prenantes ou avant le début d'une itération de développement. Lorsque des différences sont constatées et que ces modifications n'étaient pas prévues ou contreviennent aux pratiques de gestion de projet, les exigences de la ligne de base peuvent être rétablies dans le modèle actuel. Lignes de base ne permettent pas d'évaluer l'impact d'un changement, mais une fois qu'un changement a été identifié, des outils tels que la Matrice des relations et les traces d'éléments peuvent être utilisés pour déterminer l'impact d'un changement.</p>												

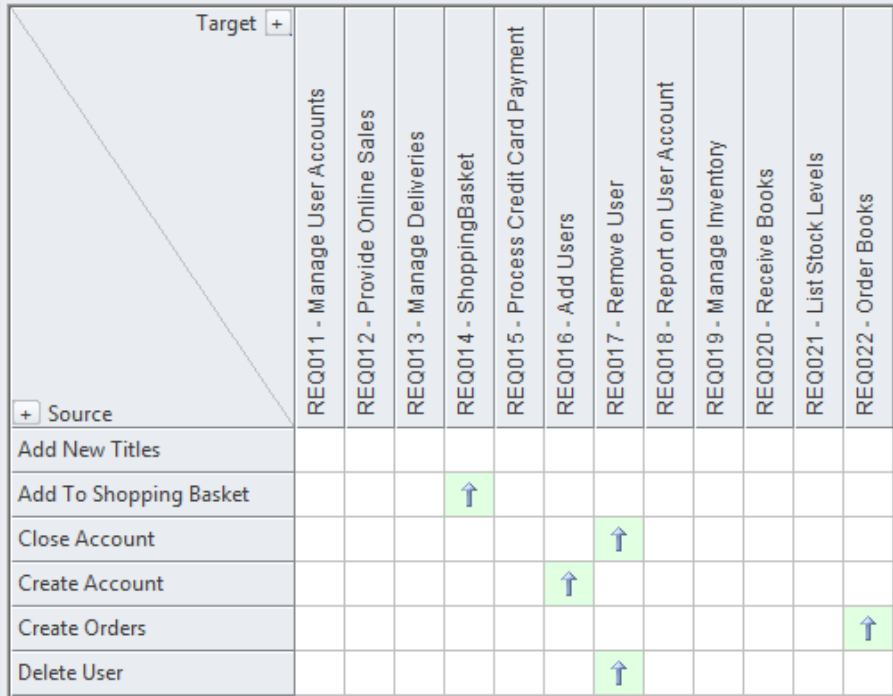
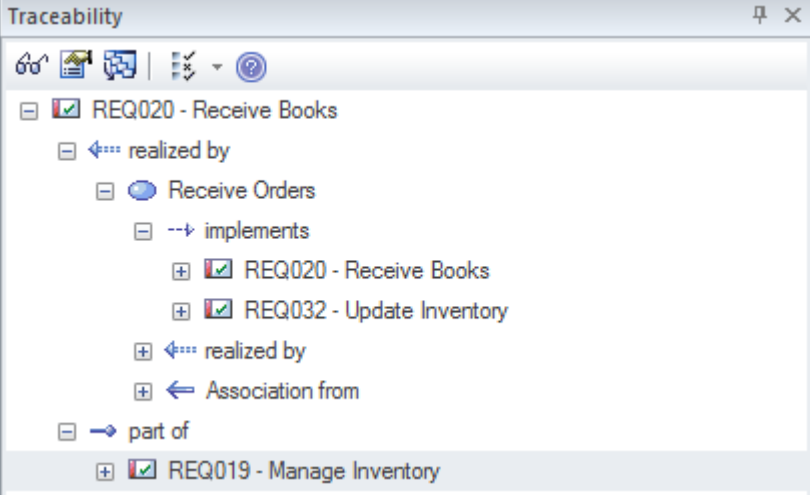


# Analyse d'impact des changements

Lorsque le développement d'un système a commencé et que les exigences changent, il y aura un impact du changement et l'effet devra être déterminé, compris et géré. Il est essentiel d'établir une traçabilité à la fois pour les éléments de processus ascendants tels que les parties prenantes et les pilotes Métier et pour les éléments de processus descendants tels que les cas d'utilisation, les composants, les cas Test et les opérations de code source afin de déterminer l'impact du changement. Enterprise Architect dispose d'un certain nombre de facilités qui peuvent vous aider à cet égard, notamment la possibilité de visualiser les traces dans diagrammes , une Matrice de relations, une fenêtre de traçabilité, des éléments de changement, de tâche et d'effort qui peuvent être utilisés pour enregistrer l'impact et ce qui est nécessaire pour le mettre en œuvre.

## Outils pour enregistrer et analyser l'impact du changement

Outil	Description
<p>Analyse à l'aide de traces d'exigences</p>	<p>La possibilité de visualiser les exigences et la manière dont elles sont connectées à d'autres éléments est un outil pratique pour analyser l'impact du changement. Exigences forment souvent une hiérarchie et lorsqu'une exigence est affectée, cela a généralement un effet d'entraînement sur les enfants de l'exigence et il est très utile de pouvoir visualiser cela dans un diagramme ou dans une hiérarchie. Exigences sont également généralement reliées aux éléments de processus en amont et en aval et un diagramme permet de visualiser et d'analyser ces connexions. La fonction Insérer des éléments associés peut découvrir ces connexions et dessiner et disposition automatiquement un diagramme permettant au modélisateur de passer son temps à analyser l'impact.</p> <h3 style="text-align: center;">Tracing Requirements</h3> <p style="text-align: center;">This diagram shows the expressive power of putting disparate elements onto a diagram.</p> <p style="text-align: center;">It shows the traceability between different layers of a system. The traceability can be from the Requirements to the Use Cases that Realize them, to the logical Components that will deliver the required functionality.</p>  <p>The diagram illustrates traceability between different layers of a system. It features three main elements: two requirements (REQ019 - Manage Inventory and REQ032 - Update Inventory) on the left, a use case (Manage Titles) in the center, and a component (Inventory Manager) on the right. Dashed arrows with open arrowheads point from the requirements to the use case, labeled with «realize». A dashed arrow with an open arrowhead points from the component to the use case, labeled with «trace».</p>
<p>Analyse à l'aide d'une matrice de relations</p>	<p>La Matrice de relations peut être utilisée pour visualiser les exigences et leurs connexions en plaçant l'exigence sur un axe de la matrice et les éléments connectés sur l'autre. Cette méthode est très utile dans les ateliers lorsque l'on travaille avec des personnes qui ne sont peut-être pas familiarisées avec les langages modélisation tels que UML ou qui travaillent mieux avec des types de vue de type tableur. Il est possible de créer un nombre illimité de matrices et de stocker leurs spécifications afin de pouvoir les rappeler facilement.</p>


	 <table border="1" data-bbox="523 197 1420 891"> <thead> <tr> <th>Target +</th> <th>REQ011 - Manage User Accounts</th> <th>REQ012 - Provide Online Sales</th> <th>REQ013 - Manage Deliveries</th> <th>REQ014 - ShoppingBasket</th> <th>REQ015 - Process Credit Card Payment</th> <th>REQ016 - Add Users</th> <th>REQ017 - Remove User</th> <th>REQ018 - Report on User Account</th> <th>REQ019 - Manage Inventory</th> <th>REQ020 - Receive Books</th> <th>REQ021 - List Stock Levels</th> <th>REQ022 - Order Books</th> </tr> </thead> <tbody> <tr> <td>+ Source</td> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>Add New Titles</td> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>Add To Shopping Basket</td> <td></td><td></td><td></td><td>↑</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>Close Account</td> <td></td><td></td><td></td><td></td><td></td><td></td><td>↑</td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>Create Account</td> <td></td><td></td><td></td><td></td><td></td><td>↑</td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>Create Orders</td> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>↑</td> </tr> <tr> <td>Delete User</td> <td></td><td></td><td></td><td></td><td></td><td></td><td>↑</td><td></td><td></td><td></td><td></td><td></td> </tr> </tbody> </table>	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	+ Source													Add New Titles													Add To Shopping Basket				↑									Close Account							↑						Create Account						↑							Create Orders												↑	Delete User							↑					
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																																																																																													
+ Source																																																																																																									
Add New Titles																																																																																																									
Add To Shopping Basket				↑																																																																																																					
Close Account							↑																																																																																																		
Create Account						↑																																																																																																			
Create Orders												↑																																																																																													
Delete User							↑																																																																																																		
<p>Analyse à l'aide de la fenêtre de traçabilité</p>	<p>La fenêtre Traçabilité est une fenêtre pratique qui montre la hiérarchie des éléments du Référentiel . Elle est particulièrement utile car elle montre de manière inconditionnelle comment les éléments sont liés les uns aux autres. D'autres vues du référentiel peuvent être configurées pour afficher uniquement des éléments particuliers dans le but de communiquer une idée alors que la fenêtre Traçabilité affichera toutes les relations auxquelles participe un élément, ce qui la rend particulièrement utile pour analyser l'impact d'un changement.</p>  <p>The screenshot shows the 'Traceability' window with a tree view. The root node is 'REQ020 - Receive Books'. It has several children: '←←← realized by' (with a sub-child 'Receive Orders'), '→→ implements' (with sub-children 'REQ020 - Receive Books' and 'REQ032 - Update Inventory'), '←←← realized by', and '← Association from'. At the bottom, there is a '→ part of' relationship with 'REQ019 - Manage Inventory'.</p>																																																																																																								

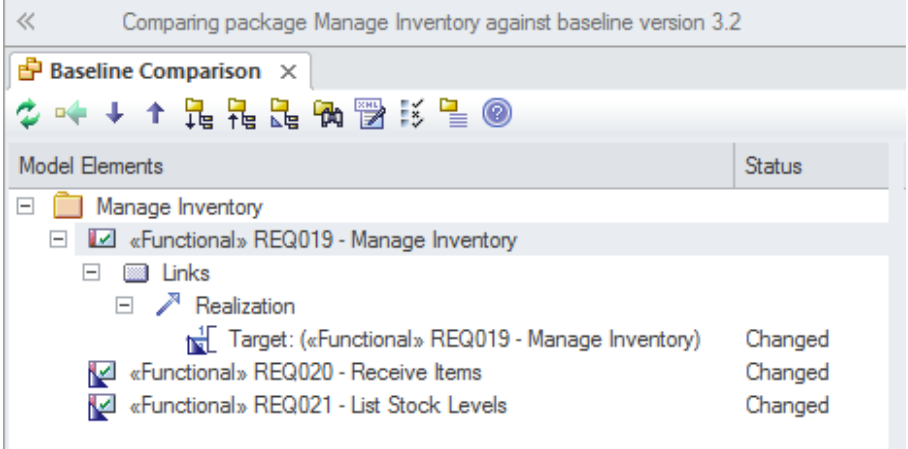
## Exigence Volatilité

Les pressions du marché pour commercialiser les produits et les systèmes le plus tôt possible sont de plus en plus fortes, ce qui met les équipes de projet sous pression pour développer, tester et déployer les produits dans des délais de plus en plus courts. Les processus d'exigences ont considérablement changé ces dernières années pour garantir que des spécifications stables, correctes et bien articulées soient fournies aux architectes, concepteurs et développeurs lorsqu'ils en ont besoin. On assiste à une évolution vers des processus itératifs et incrémentaux, ce qui nécessite de fournir un ensemble d'exigences stables pour chaque itération. La multiplication des exigences est souvent un indicateur qu'un problème n'est pas clairement compris, que les parties prenantes n'ont pas été compromises et qu'il existe des problèmes politiques non résolus, que le périmètre n'est pas défini ou que l'entreprise elle-même est en fluctuation. Enterprise Architect dispose d'un certain nombre de mécanismes qui peuvent être utilisés pour résoudre ce problème. Enterprise Architect ne dispose pas d'une propriété intégrée pour la volatilité des exigences (stabilité), mais en utilisant le mécanisme d'extension UML à usage général de Valeur Étiquetées une étiquette pourrait être créée pour enregistrer cette propriété.

Note : les exigences internes ont une propriété de stabilité, mais pas les exigences externes.

### Mécanismes de gestion de la volatilité des besoins

Mécanisme	Description
La volatilité comme Valeur Étiquetée	<p>Enterprise Architect fournit une série de propriétés pour Exigences , mais des propriétés supplémentaires peuvent être créées pour enregistrer d'autres propriétés telles que la volatilité d'une exigence ou la source de l'exigence. Cela est réalisé à l'aide du mécanisme UML Valeur Étiquetée , qui permet à tout élément comprenant Exigences d'avoir une ou plusieurs étiquettes appliquées, représentant une propriété à laquelle une valeur peut être attribuée. Enterprise Architect a étendu ce mécanisme et permet au modélisateur de créer une liste de valeurs qui peuvent être choisies dans une liste déroulante à l'aide des Valeur Étiquetées structurées prédéfinies. Cela permet à une équipe de définir sa propre liste de valeurs de volatilité, telles que extrême, élevée, moyenne, faible, minimale.</p> 
Utilisation Lignes de base	<p>L' facilité Baseline est un outil efficace qui permet à un utilisateur de prendre un instantané d'un modèle ou plus généralement d'un fragment de modèle, puis, au fur et à mesure du développement du modèle, de comparer la nouvelle version du modèle à la baseline, identifiant ainsi tout ce qui a changé depuis la prise de la baseline. Lignes de base ont une applicabilité générale, mais sont particulièrement utiles dans la gestion des exigences, où les exigences sont souvent considérées comme validées ou gelées et toute modification qui leur est apportée doit être enregistrée comme un changement. L'outil Baseline dispose d'un utilitaire de comparaison qui répertorie de manière pratique les changements entre le modèle actuel et la baseline.</p>

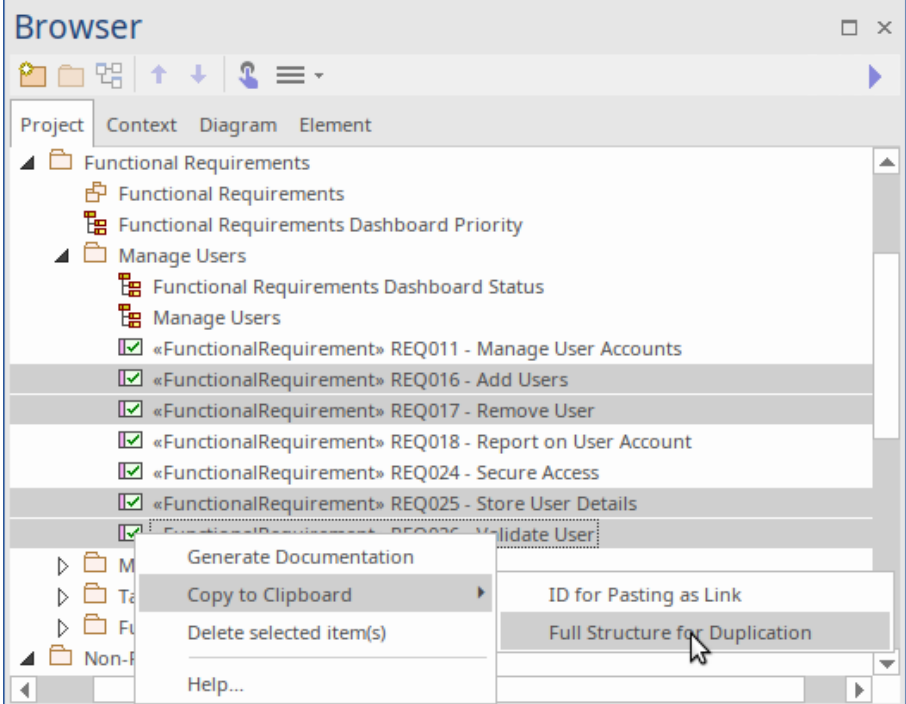
	
<p>Recherches sur les besoins de barattage</p>	<p>Enterprise Architect dispose d'une facilité de recherche sophistiquée qui permet à un utilisateur de rechercher soit dans un Paquetage sélectionné, soit dans l'ensemble du référentiel, pour localiser les éléments qui répondent à des critères précis. Cela peut être utilisé pour localiser les exigences qui n'ont pas changé en recherchant un changement dans la date de modification avant une date spécifiée, fournissant ainsi une liste d'exigences stables. Alternativement, si la volatilité a été définie à l'aide d'une Valeur Étiquetée, tous les éléments avec une volatilité spécifiée peuvent être localisés. La facilité de recherche renvoie une liste d'éléments qui peuvent être localisés dans la fenêtre Navigateur ; la recherche peut être utilisée comme base d'un Modèle Vue à utiliser pour afficher les exigences volatiles ou non volatiles.</p>

## Réutilisation des exigences

Le concept de réutilisation des artefacts d'un processus de développement système a été évoqué dans de nombreux articles et manuels, mais il a traditionnellement été limité aux composants logiciels. Ces dernières années, la notion de réutilisation des spécifications, y compris des exigences, a commencé à gagner du terrain. La réutilisation est particulièrement importante lorsque les organisations créent une famille de produits avec fonctionnalités similaires, ou lorsqu'il existe une communauté d'utilisateurs au sein d'un secteur ou d'un domaine. D'autres types d'exigences telles que les exigences de sécurité et réglementaires s'appliquent généralement à un certain nombre de projets. Règles Métier et les préoccupations des parties prenantes s'appliquent également généralement à de nombreux projets et sont mieux cataloguées en dehors des structures de projet individuelles. Enterprise Architect fournit un certain nombre de mécanismes sophistiqués pour gérer la réutilisation d'éléments dans plusieurs projets, notamment la structuration du référentiel pour la réutilisation, l'importation d'exigences à partir d'autres sources et un Service d'Actifs Réutilisables .

### Mécanisme de réutilisation des exigences

Mécanisme	Description
Structurer le référentiel pour la réutilisation des exigences	Lorsque vous configurez un référentiel, vous avez le choix de le structurer pour un seul projet ou pour plusieurs projets, qui à leur tour peuvent être organisés selon un certain nombre de programmes de travail. Enterprise Architect donne au modélisateur un contrôle complet sur la manière dont le référentiel est structuré, ce qui permet de configurer Paquetages au-dessus du niveau des projets où certaines exigences telles que Exigences Métier , réglementaires et architecturales peuvent être ajoutées.
Créer un modèle de base	Lorsque vous créez un nouveau référentiel dans Enterprise Architect , vous avez la possibilité de créer un modèle vierge à l'aide du Constructeur de Modèle pour vous aider à configurer une structure de référentiel, ou vous pouvez utiliser un modèle de base comme gabarit pour le nouveau modèle. Le modèle de base est un bon emplacement pour stocker des ressources réutilisables telles que Métier , Exigences réglementaires et architecturales et Politiques et Règles Métier .
Importer des exigences à partir d'autres modèles	Il est assez courant d'avoir plusieurs référentiels Enterprise Architect dans une organisation et il est très facile de copier et de réutiliser Exigences (ou tout autre élément) d'un modèle dans un autre. Cela peut être réalisé en copiant simplement une sélection d' Exigences ou un Paquetage entier d'un référentiel à un autre, ou même d'un projet à un autre dans le même référentiel. Enterprise Architect fonctionne de la même manière que n'importe quel autre programme Windows , en copiant simplement la sélection dans le presse-papiers, puis en lui permettant d'être collée à un autre emplacement dans le même modèle ou dans un autre référentiel ouvert.

	 <p>The screenshot shows the 'Browser' window in Enterprise Architect. The project tree is expanded to show 'Functional Requirements' and 'Manage Users'. A list of requirements is displayed, including '«FunctionalRequirement» REQ011 - Manage User Accounts' through '«FunctionalRequirement» REQ026 - Validate User'. A context menu is open over the 'Validate User' requirement, with options like 'Generate Documentation', 'Copy to Clipboard', and 'Full Structure for Duplication'. The 'Full Structure for Duplication' option is highlighted, and a sub-menu is visible with 'ID for Pasting as Link' and 'Full Structure for Duplication'.</p>
<p>Utiliser le Service d'Actifs Réutilisables</p>	<p>Le Service d'Actifs Réutilisables (RAS) est particulièrement utile pour les équipes distribuées et fournit un mécanisme simple et pratique aux modélisateurs pour distribuer ou télécharger des structures et des éléments de modèles réutilisables tels que Exigences via un référentiel partagé, accessible via une connexion Pro Cloud Server . Exigences au niveau de l'entreprise ou de l'organisation peuvent être stockées dans le RAS et différentes équipes peuvent les incorporer dans leurs modèles. La gouvernance des actifs est généralement gérée par le propriétaire de l'actif (registre) au niveau du Service d'Actifs Réutilisables .</p>



## Relations d'exigences

Enterprise Architect supporte toutes les relations entre les exigences SysML, qui peuvent être visualisées à différents endroits de l'interface utilisateur, offrant ainsi une manière flexible de travailler avec ces connecteurs importants. Les relations entre les éléments (y compris Exigences) ne sont pas visibles dans la fenêtre Navigateur, car cela encombrerait les éléments alors qu'il existe des moyens plus efficaces de visualiser les connexions.

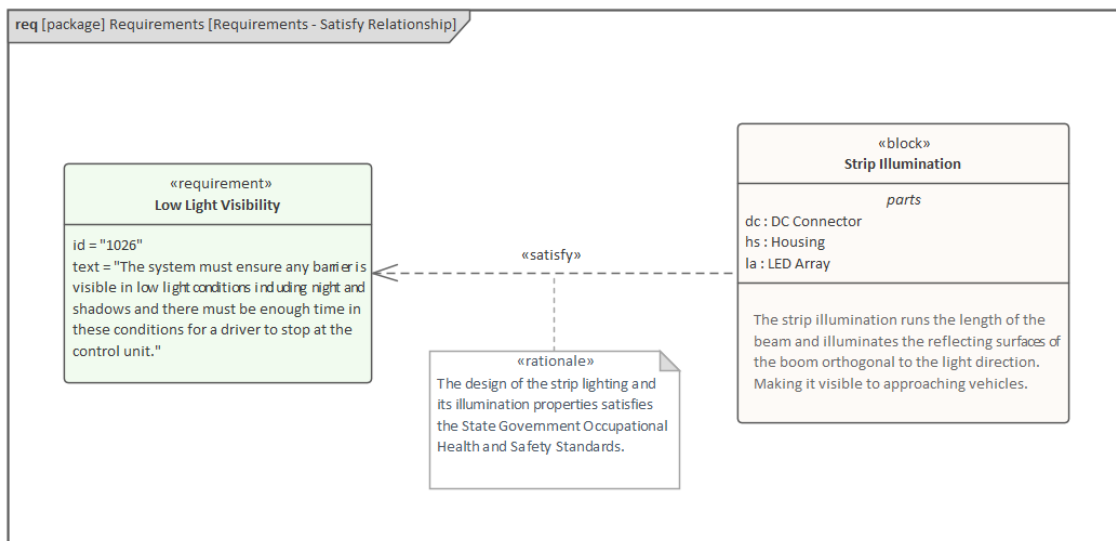
La fenêtre Relations est utile pour visualiser les éléments, que ce soit dans la fenêtre Navigateur ou dans un diagramme. Lorsqu'un élément est sélectionné dans la fenêtre Navigateur ou dans un diagramme, le contexte change et la fenêtre Relations affiche uniquement les relations qui existent entre l'élément sélectionné et les autres éléments du modèle, y compris les autres Exigences.

Les relations entre Exigences et d'autres éléments, y compris d'autres Exigences, peuvent être visualisées dans n'importe quel diagramme, y compris diagrammes Exigences, de trois manières différentes :

- Un *connecteur* entre deux éléments
- Un *compartiment* dans l'élément Requirement
- Une notation *de légende* sous la forme d'une note attachée à une exigence ou à un autre élément de modèle

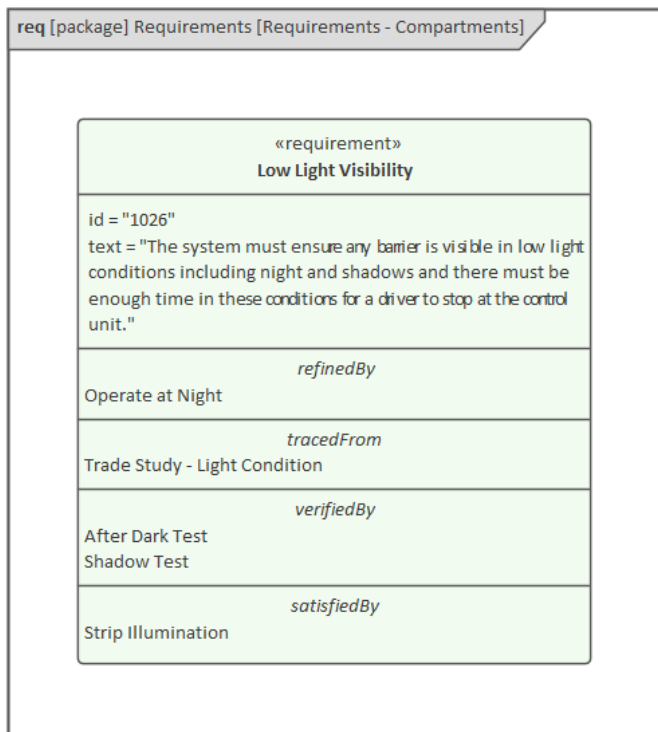
Les trois notations diagramme ont chacune leur utilité et offrent au modélisateur la flexibilité nécessaire pour choisir la représentation appropriée à un objectif et à un public particuliers.

La relation tracée entre deux éléments visibles dans un diagramme est la manière la plus courante de visualiser les relations entre exigences ; la ligne en pointillés est tracée du client (l'élément dépendant) au fournisseur (l'élément fournisseur). Ainsi, dans cet exemple, le Bloc « Éclairage par bande » est le client et il dépend de l'exigence « Visibilité en basse lumière », de sorte que la flèche pointe du Bloc (client) vers l'exigence (fournisseur).

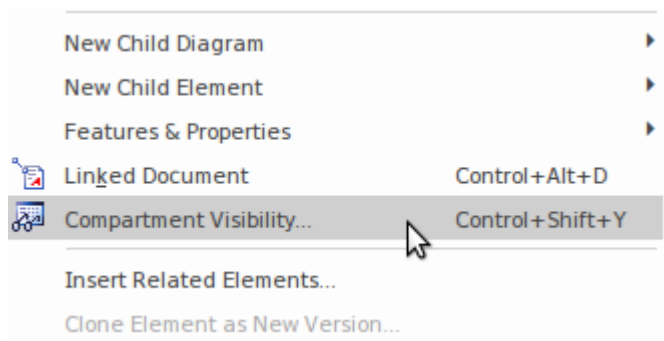


Notez également dans cet exemple que le modélisateur a choisi d'afficher le compartiment Parts, montrant les Parts qui composent le Bloc, et le compartiment Notes qui décrit le Bloc. Une justification a également été ajoutée pour qualifier la relation « Satisfy » et pour fournir une explication sur la raison pour laquelle le Bloc a été choisi dans le contexte des normes.

Les *compartiments* peuvent être utilisés pour afficher les relations auxquelles participe une exigence, ce qui constitue un moyen compact et utile de visualiser les relations entre les exigences sans avoir à inclure les éléments associés dans le diagramme.

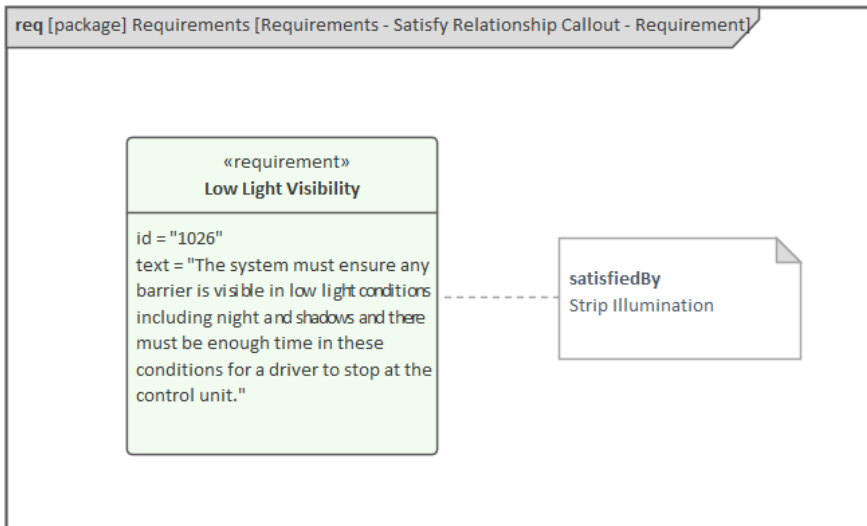


La liste des compartiments visibles peut être configurée pour chaque élément diagramme ou pour l'ensemble diagramme , offrant un contrôle granulaire précis sur la manière dont les relations sont visualisées.

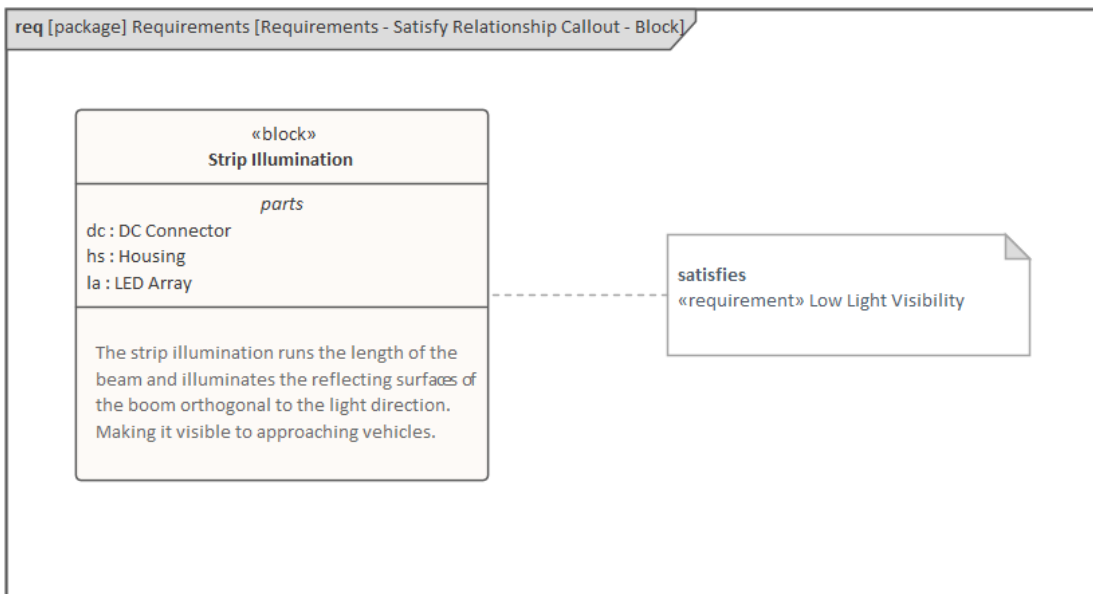


Alternativement, une notation *de légende* peut être utilisée pour afficher la relation dans une note attachée à l'élément d'exigence ou à l'élément dépendant auquel la relation se rapporte. Cette notation est particulièrement utile lorsque des éléments apparaissent dans diagrammes dans lesquels les affichages de connecteur ou de compartiment ne sont pas adaptés, tels qu'un diagramme Bloc interne, diagramme Séquence ou diagramme de cas d'utilisation, ou dans d'autres diagrammes selon ce que le modélisateur juge approprié. Les relations d'exigence sont binaires, ce qui signifie qu'elles ont deux extrémités : un fournisseur et un client. Cela signifie que la légende peut être attachée soit à une exigence, soit à l'élément de modèle associé qui, selon la relation, peut être un Bloc , un cas Test , un cas d'utilisation ou un autre élément de modèle, y compris une autre exigence.

Dans ce diagramme l'attention du modélisateur est portée sur l'exigence et l'élément Bloc est répertorié dans la note stéréotypée comme <<satisfiedBy>>.



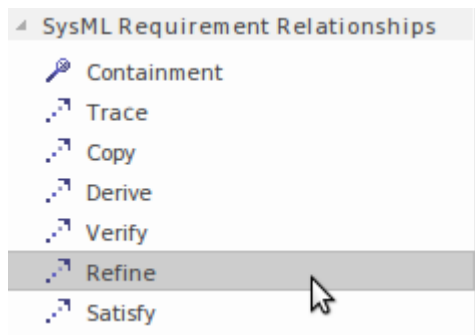
Dans ce diagramme , le focus du modélisateur est passé à un Bloc et l'élément Exigence est répertorié dans la note stéréotypée comme <<satisfait>>.



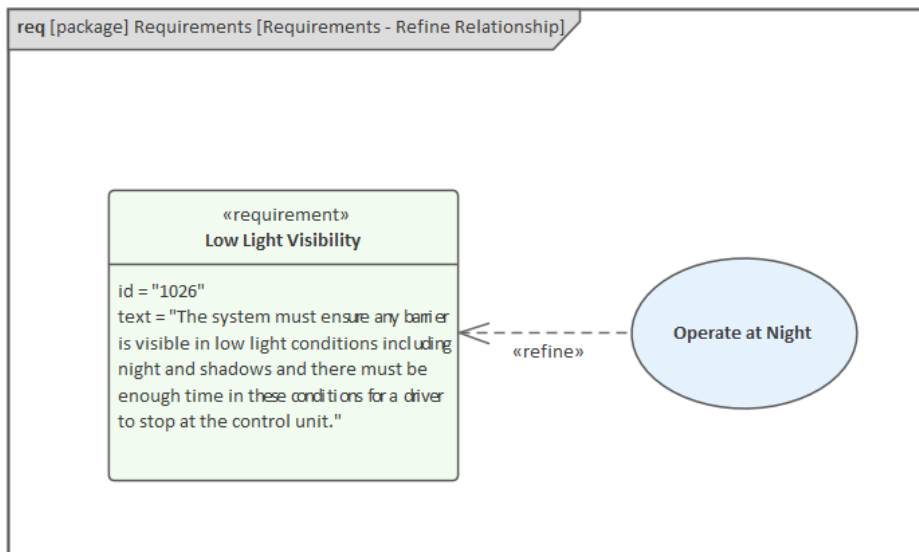
La section suivante détaille les relations d'exigences, en fournissant un exemple de chaque relation.

## Ajout de raffinement à une exigence

La relation *Affiner* est une relation entre une exigence et un autre élément de modèle qui ajoute un raffinement ou des informations supplémentaires qui aident à clarifier l'exigence afin que sa signification soit plus évidente. La relation Affiner est disponible à partir de la page « Relations » de la boîte à outils Exigences SysML.



La relation d'affinage peut être établie entre une exigence et n'importe quel élément de modèle tel qu'un cas d'utilisation, une StateMachine ou une activité. Le choix de l'élément de modèle dépendra des informations exprimées dans l'exigence et de la discrétion du modélisateur ou de l'ingénieur.

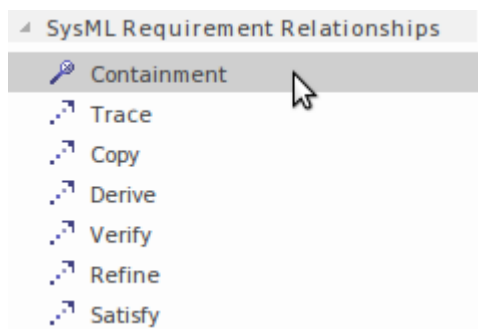


Dans cet exemple, le cas d'utilisation n'étend ni n'embellit l'exigence, mais ajoute plutôt des détails sous forme de descriptions et de scénarios qui rendront la signification de l'exigence plus facile à comprendre.

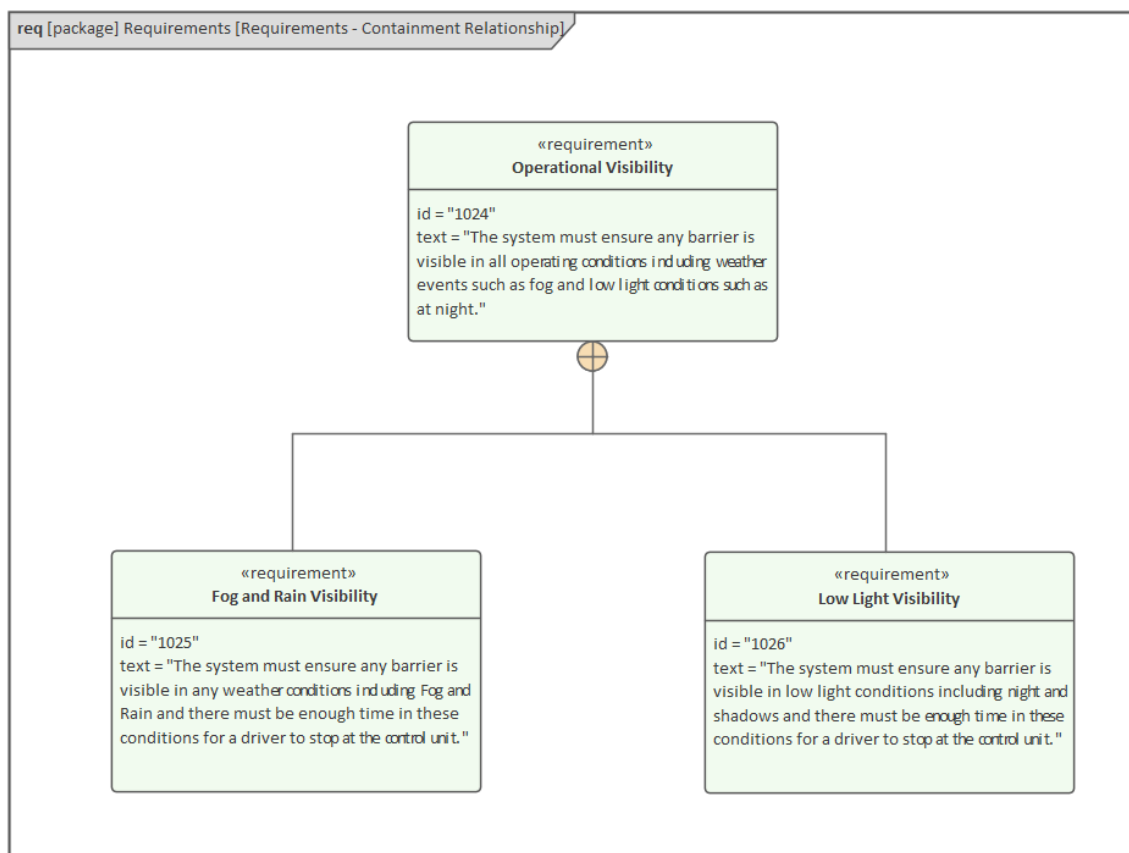
## Relation de confinement

Le nom de la relation *Containment* décrit bien son objectif : elle est utilisée pour montrer qu'une ou plusieurs Exigences sont contenues dans, ou sont regroupées par, une autre exigence de niveau supérieur. Il s'agit d'une relation fondamentale et très utilisée lors de modélisation des exigences de tout système, même de complexité modérée. Un grand système peut avoir des milliers, voire des dizaines de milliers, d'exigences, et il est préférable de les regrouper dans des hiérarchies. Une alternative à l'utilisation de la relation Containment consiste à regrouper Exigences à l'aide Paquetages . Cette méthode fonctionne lorsqu'il n'y a que deux niveaux dans la hiérarchie ou lorsque vous regroupez Exigences par type (par exemple, partie prenante ou physique), mais elle présente des limites lorsqu'elle est utilisée de manière plus extensive.

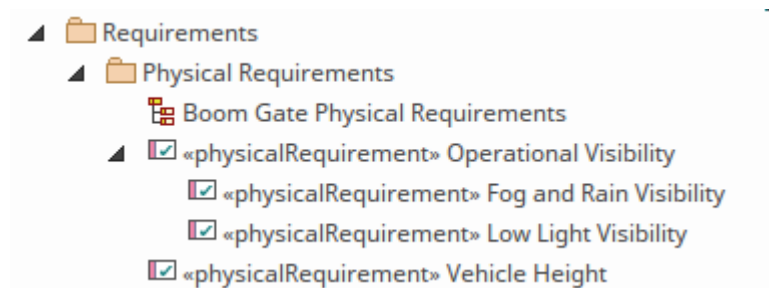
La relation de confinement est disponible à partir de la page Relations de la boîte à outils Exigences SysML.



Ce diagramme montre l'utilisation de la relation de confinement pour afficher deux Exigences de niveau inférieur qui sont « contenues » par une exigence de niveau supérieur.



Le confinement des Exigences peut également être visualisé dans la fenêtre Navigateur , où le confinement est représenté par des éléments imbriqués ou, plus formellement, les éléments contenus sont les enfants d'une autre exigence. Ce nivellement est possible pour tous les éléments du référentiel, mais a une signification particulière avec Exigences . Cette image montre les mêmes Exigences que dans le diagramme précédent, mais dans la fenêtre Navigateur .

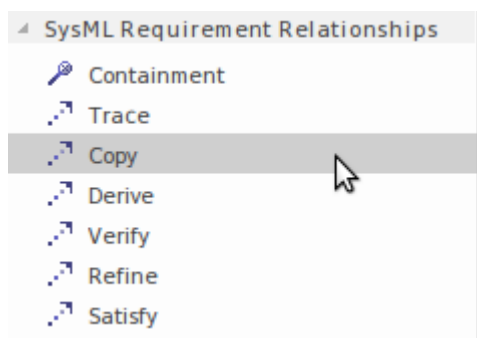


Note que l'imbrication des Exigences dans la fenêtre Navigateur ne crée pas de relations de confinement entre Exigences . En fait, il est possible que les deux méthodes différentes soient désynchronisées l'une par rapport à l'autre car ce sont des mécanismes indépendants.

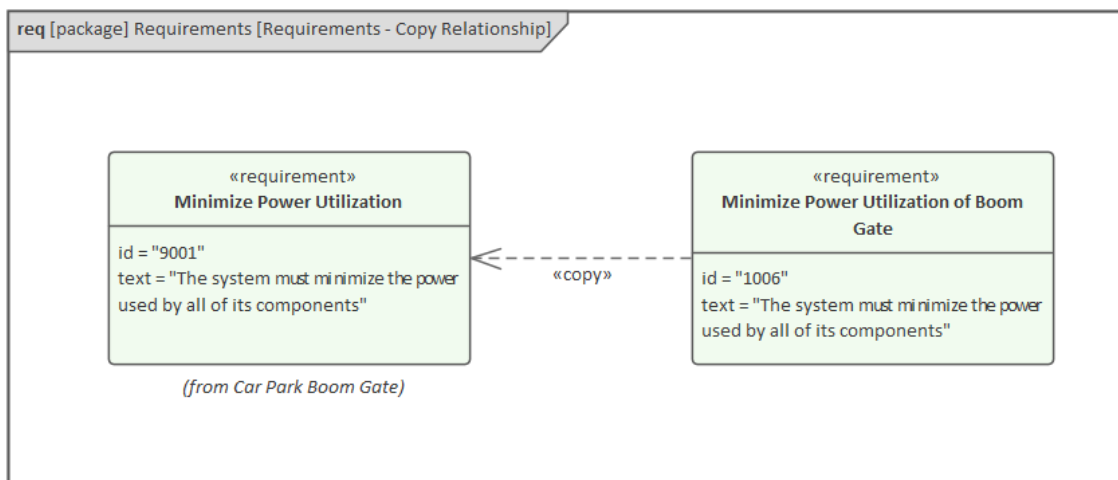
## Copier Exigences existantes

La relation *Copie* est une relation entre deux Exigences ; elle est utilisée pour montrer qu'une exigence est une copie d'une autre. La relation est un type de dépendance et est représentée par une ligne en pointillés avec le mot-clé `<<copy>>`, avec une pointe de flèche ouverte pointant de l'exigence copiée (client) vers l'exigence de base (fournisseur). Étant donné que l'élicitation et la gestion Exigences sont des activités coûteuses et chronophages, et que de nombreux projets ont souvent un chevauchement d'intérêts, il est utile de réutiliser Exigences ; la relation Copie fournit un mécanisme pour le faire. L'exigence de base est généralement stockée dans l'espace de noms d'un autre projet, mais il est considéré comme une bonne pratique de déplacer les Exigences communes (de base) vers un espace de noms situé au-dessus du niveau des projets individuels.

La relation Copie est disponible à partir de la page Relations de la boîte à outils Exigences SysML.



Ce diagramme illustre une exigence d'utilisation de l'énergie qui a été copiée pour être réutilisée dans un certain nombre de projets.

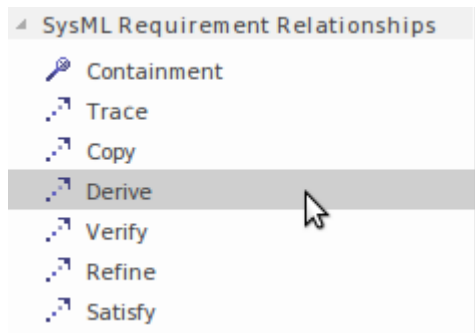


Lorsque la relation de copie est utilisée, un nouvel identifiant est attribué aux nouvelles Exigences , mais le texte de la nouvelle exigence sera une copie en lecture seule de l'exigence de base.

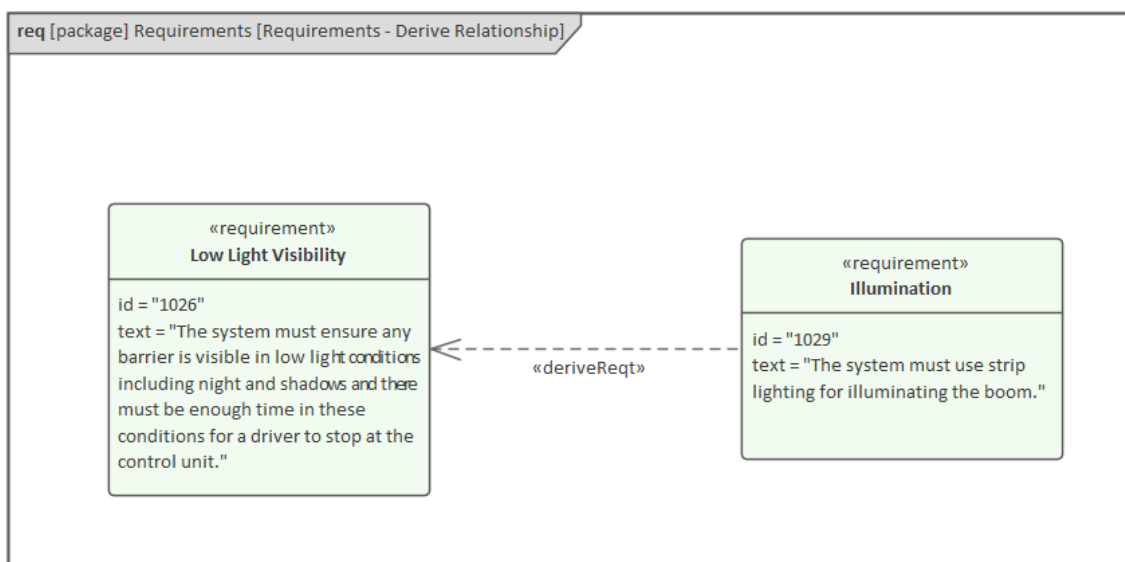
## Dérivation d'une exigence à partir d'une autre

La relation *Derive* est une relation entre deux Exigences, utilisée pour décrire le fait qu'une exigence est basée sur ou est une extension ou une dérivation d'une autre exigence.

La relation *Derive* est disponible à partir de la page Relations de la boîte à outils Exigences SysML.



Les deux Exigences se situent généralement à des niveaux d'abstraction ou de résolution différents. Une exigence d'une spécification de bas niveau peut avoir une relation de dérivation avec une exigence d'une spécification de niveau supérieur. L'exigence de niveau inférieur est généralement dérivée de l'exigence de niveau supérieur à la suite d'une enquête, d'une élaboration ou d'une analyse. L'aspect important de cette relation est que si l'exigence à l'extrémité de la flèche de la relation est modifiée, il est très probable que l'exigence dérivée devra être réanalysée.

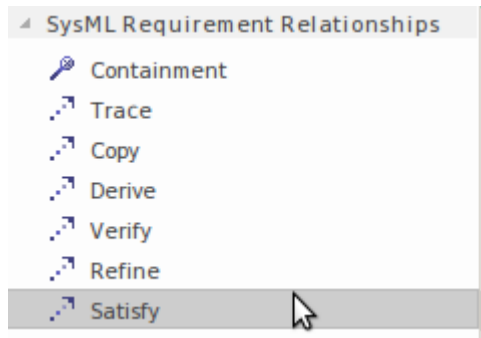




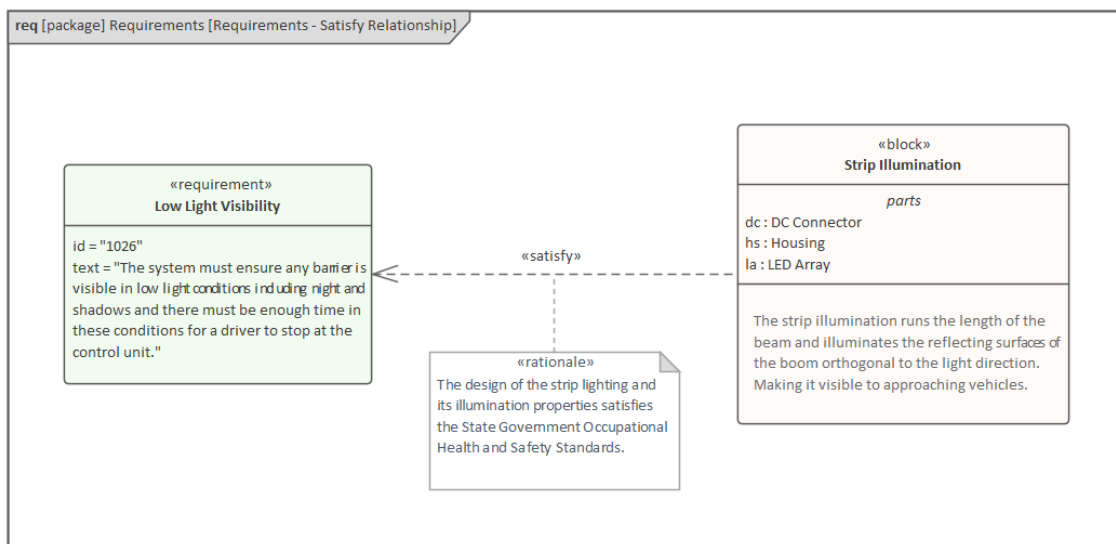
## S'assurer qu'une exigence est satisfaite

La relation *Satisfaisant* est utilisée pour montrer qu'un ou plusieurs éléments du modèle dans l'architecture ou la conception répondent à la notion exprimée dans l'exigence. Il s'agit d'un lien ou d'un pont important entre ce qui pourrait être décrit comme le problème ou l'opportunité et l'architecture, la conception et, une fois vérifiée, la mise en œuvre.

La relation Satisfy est disponible à partir de la page « Relations » de la boîte à outils Exigences SysML.

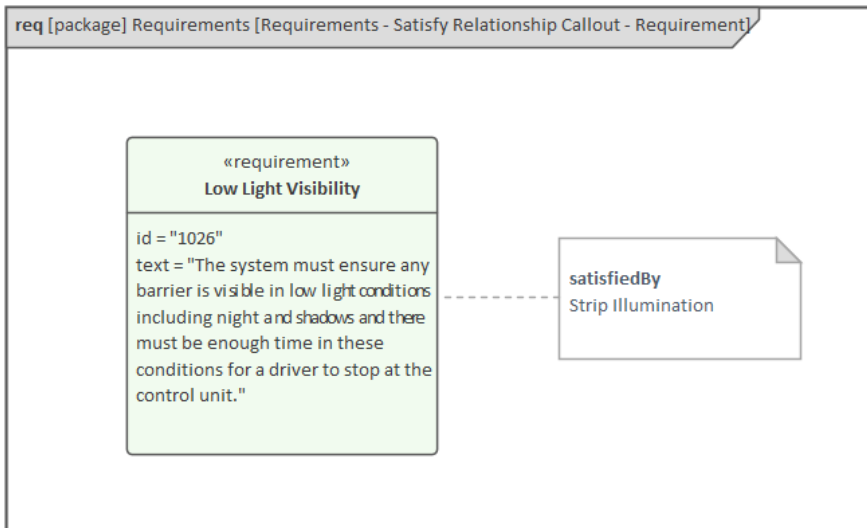


La relation est un type de dépendance et est représentée par une ligne pointillée avec le mot-clé <<satisfy>>, avec une pointe de flèche ouverte pointant de l'élément de conception (Client) vers l'exigence (Fournisseur).

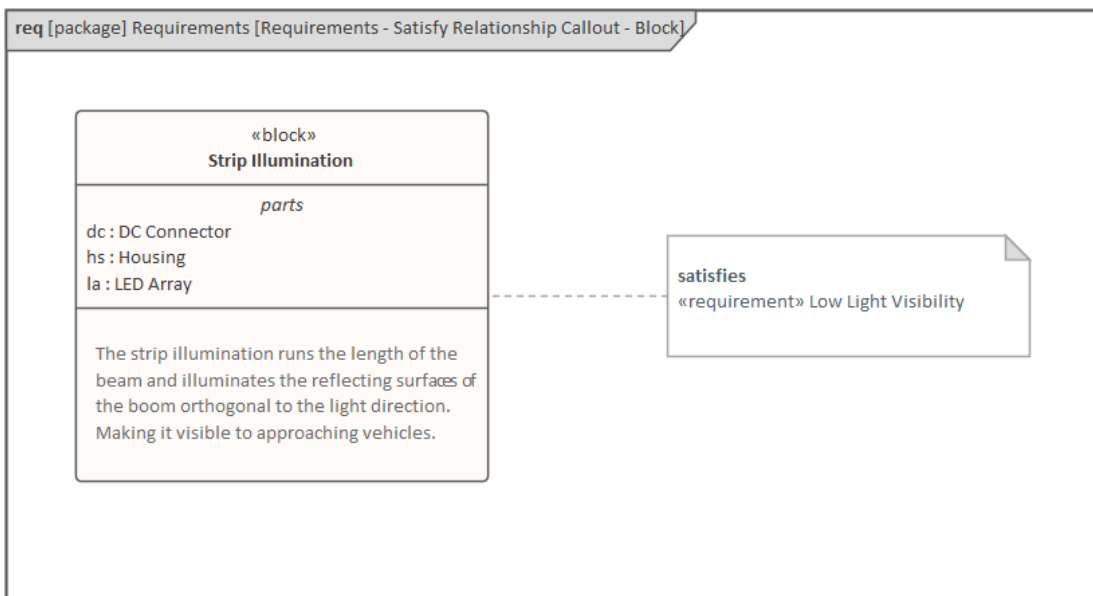


Dans ce diagramme on peut voir que le Bloc « Strip Illumination » satisfait à l'exigence relative à la visibilité de la rampe dans des conditions de faible luminosité. Une justification est également ajoutée qui décrit la conformité aux réglementations gouvernementales State .

Dans ce diagramme la relation Satisfy a été affichée à l'aide de la notation Legend.



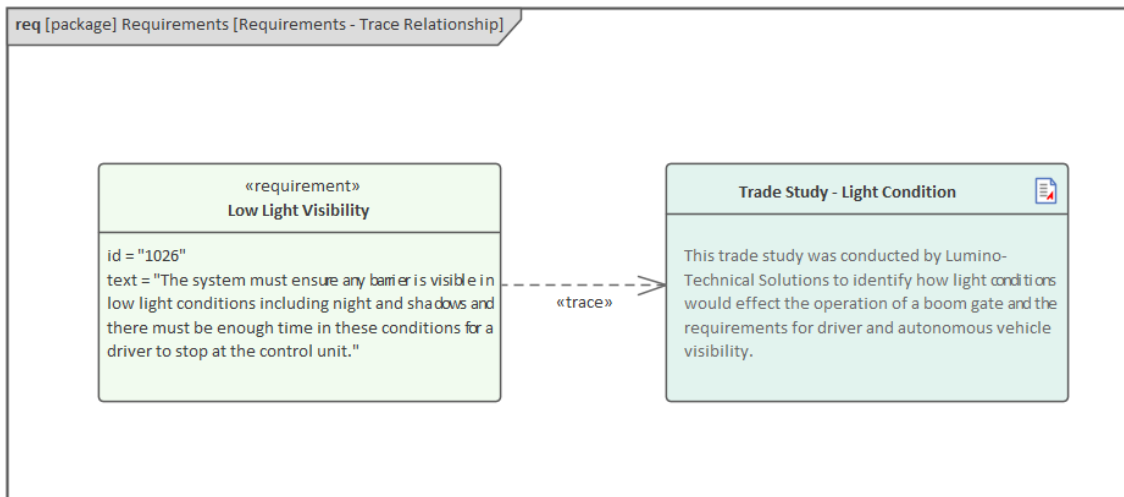
Dans le diagramme suivant, la notation de légende a été utilisée, mais cette fois, le Bloc est référencé dans la légende.



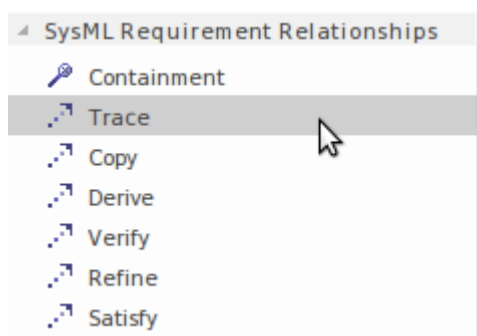
## Traçabilité des éléments Modèle

La relation *Trace* est une relation à usage général, largement utilisée, qui relie une exigence à tout autre élément de modélisation. La relation est exprimée sous forme de ligne pointillée avec le mot-clé «trace», qui indique la signification ; la pointe de la flèche pointe vers un élément de processus supérieur (un élément qui a été créé plus tôt dans le processus).

Dans ce diagramme le modélisateur souhaite montrer qu'une exigence a une relation avec une étude commerciale représentée par un artefact de document. Le document peut avoir été écrit dans Enterprise Architect ou il peut s'agir d'un document externe lié.



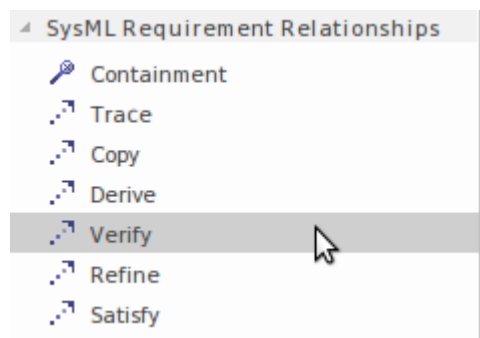
La relation Trace agit comme un fourre-tout et est utile lorsqu'un modélisateur souhaite montrer qu'une exigence a une relation sémantique avec un élément de processus en amont et qu'aucune des autres relations n'est appropriée.



## Vérifier la relation

La relation *Verify* est utilisée pour indiquer qu'une exigence a été satisfaite. La relation est un type de dépendance et est représentée par une ligne en pointillés avec le mot-clé <<verify>>, avec une pointe de flèche ouverte pointant du cas Test (client) vers l'exigence (fournisseur).

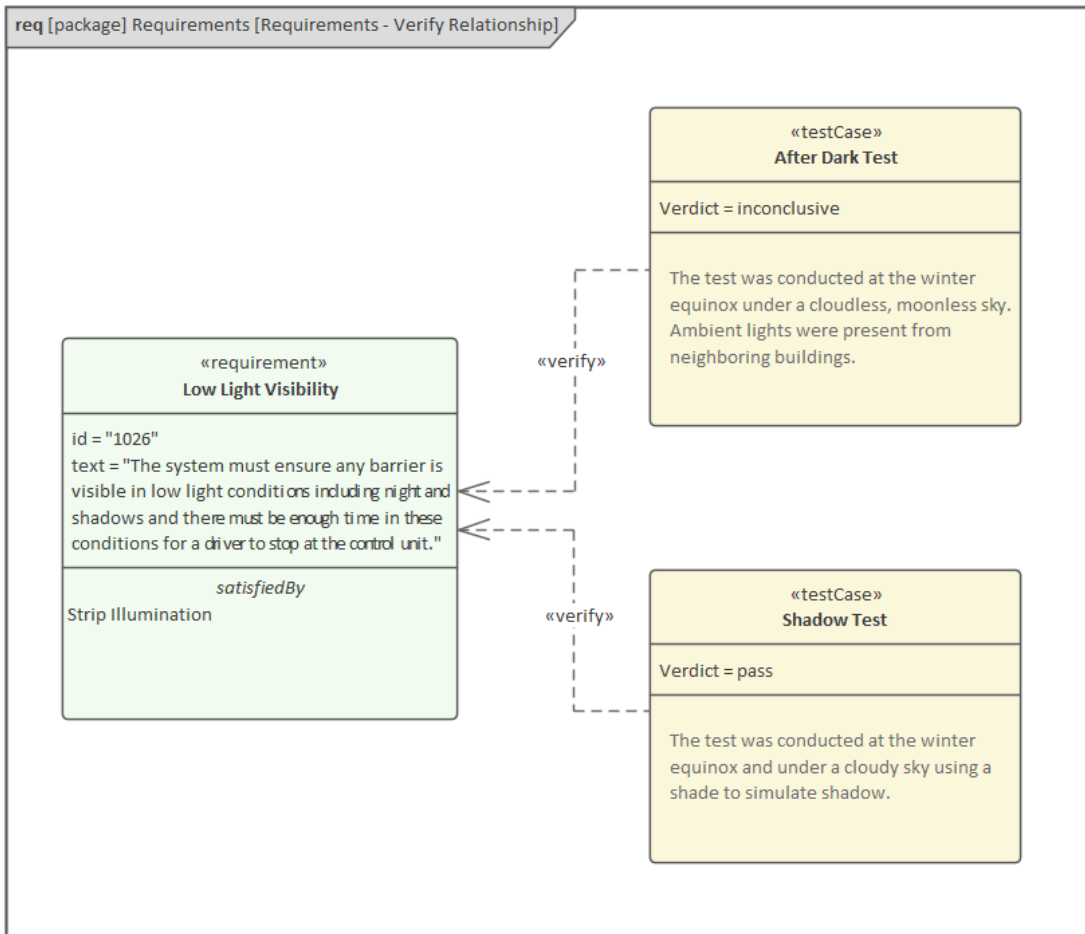
La relation Vérifier est disponible à partir de la page Relations de la boîte à outils Exigences SysML.



Le cas Test peut décrire la méthode ou le processus de test ; il contient une étiquette qui définit le verdict (résultat du test), qui peut être :

- passer
- échouer
- non concluant
- erreur
- une valeur définie par l'utilisateur

La relation *Satisfaire* a un objectif connexe dans la mesure où elle décrit quelle partie de la conception ou du système est réellement utilisée pour réaliser la notion exprimée dans l'exigence.

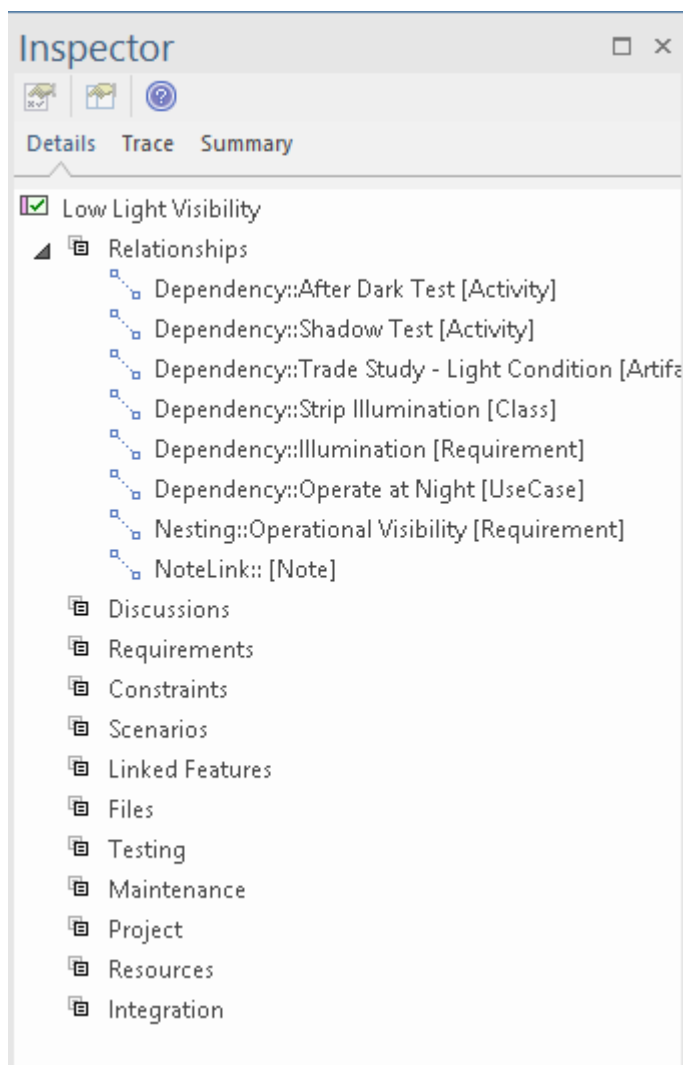


Dans ce diagramme une exigence décrit les conditions de faible luminosité et deux cas Test distincts sont utilisés pour vérifier l'exigence. Le modélisateur a choisi d'afficher le compartiment « satisfait par » pour aider à clarifier quelle partie de l'implémentation est soumise au test. Notez que le verdict est différent pour chaque cas Test : le Test After Dark n'est pas concluant alors que le Test Shadow réussit. Un modélisateur peut choisir d'afficher ou de masquer l' étiquette Verdict dans diagrammes individuels.

## Visualisation Relations entre exigences

Relations entre Exigences et les autres éléments, y compris les autres Exigences, constituent un aspect essentiel d'Ingénierie Systèmes Modèles Basée. À bien des égards, ces relations constituent des passerelles importantes entre la spécification et la conception, ou le problème et la solution. Ces relations peuvent être visualisées dans un large éventail de fenêtres spécialisées et de mécanismes d'interface utilisateur. L'une des premières choses qu'un nouveau venu dans Enterprise Architect remarquera est que ces relations ne sont pas visiblement imbriquées sous les éléments de l'onglet « Projet » de la fenêtre Navigateur. Il a été délibérément décidé de ne pas encombrer l'onglet « Projet » avec ces relations, mais plutôt de les rendre visibles dans d'autres affichages qui peuvent être ancrés et visualisés en même temps que les éléments auxquels elles se rapportent.

Cette illustration montre l'onglet « Détails » de la fenêtre Inspecteur avec le nœud « Relations » développé pour afficher toutes les relations qui sont connectées à l'exigence « Visibilité en basse lumière », qui a été sélectionnée dans l'onglet « Projet ». Il existe également une fenêtre Relations dédiée.

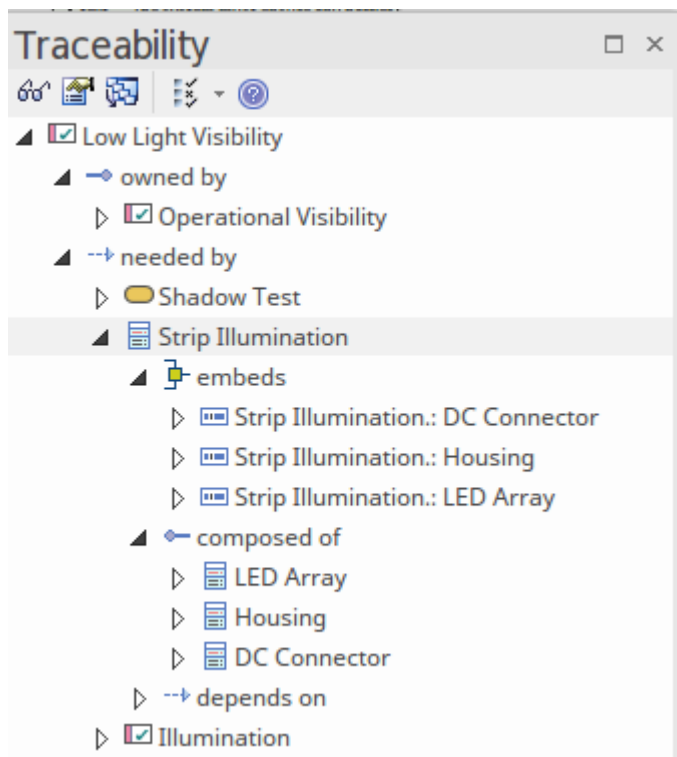


Il existe également une large gamme d'autres fenêtres dans lesquelles les relations peuvent être visualisées. Pour plus d'informations, consultez la rubrique [The Browser Window](#).

### Fenêtre de traçabilité

La fenêtre Traçabilité est une fonctionnalité utile et unique d'Enterprise Architect qui permet d'explorer les relations, afin que le modélisateur puisse parcourir efficacement le graphique des éléments et leurs connexions. Il est utile d'avoir cette fenêtre ouverte lorsque vous souhaitez visualiser la manière dont un élément est connecté et également à quoi les

éléments connectés sont connectés. Ainsi, par exemple, dans ce modèle de la barrière de sécurité, lorsque l'exigence « Visibilité en faible luminosité » est sélectionnée, la fenêtre Traçabilité indique qu'elle est connectée au Bloc « Éclairage en bande » qui satisfait l'exigence.



La fenêtre Traçabilité peut être ouverte à partir de l'option du ruban « Conception > Élément > Trace ».

Le modélisateur dans cette situation pourrait également être intéressé par les aspects structurels du Bloc « Strip Lighting » et peut ainsi suivre les relations de cet élément pour découvrir ses relations structurelles, en parcourant le graphique pour trouver des réponses aux questions et en explorant le modèle.

Si vous ne souhaitez pas visualiser les relations, il est préférable de ne pas ouvrir cette fenêtre, car son contenu doit être rendu à chaque fois que vous changez le focus sur un autre élément et, pour les éléments bien connectés, cela peut prendre un certain temps, augmentant le temps nécessaire pour se déplacer dans le modèle. Pour plus d'informations, consultez la [Traceability Window](#) Rubrique d'aide.

## Fenêtre Relations

La fenêtre Relations est une fenêtre utile à avoir ouverte lorsque l'on travaille avec des préoccupations transversales, comme c'est généralement le cas avec Exigences . Lorsqu'un élément est sélectionné dans une fenêtre, la fenêtre Navigateur ou un diagramme , la fenêtre Relations affiche une liste des connecteurs qui ciblent l'élément sélectionné (cible) ou émanent de l'élément (source). Un autre aspect utile de cette fenêtre est que la colonne ' Vue ' indique si la relation est visible dans le diagramme actuellement ouvert.

La fenêtre Relations peut être ouverte à partir de l'option du ruban ' Démarrer > Toutes Windows > Propriétés > Responsabilités > Relations'.

Un modélisateur peut également localiser tous diagrammes qui contiennent la relation sélectionnée, en choisissant l'option « Rechercher dans tous Diagrammes » dans le menu contextuel. Dans cette illustration, on peut voir que la relation « deriveReq » qui relie l'exigence « Visibilité en cas de faible luminosité » et l'exigence « Illumination » existe dans deux diagrammes . Pour plus d'informations, consultez la rubrique d'aide [The Relationships Window](#) .

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

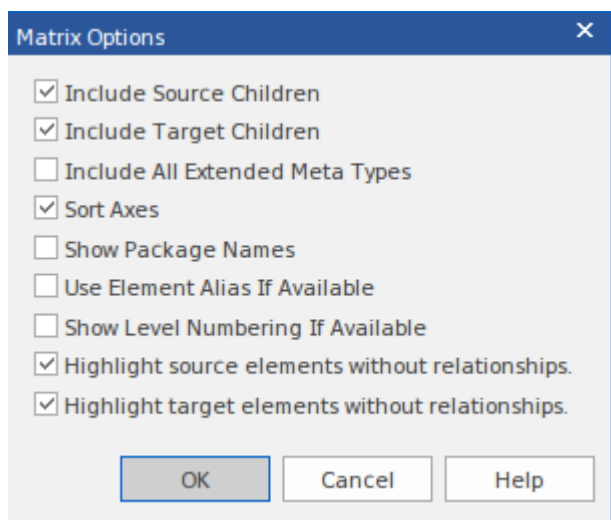
## Matrice de relations

La Matrice de relations est un excellent outil pour visualiser les connexions entre les éléments de deux Paquetages , dans une interface ressemblant à une feuille de calcul avec des lignes et des colonnes. L'outil est particulièrement utile lorsqu'il est utilisé avec Exigences , et permet à un ingénieur de voir comment Exigences sont liées à d'autres éléments, y compris d'autres Exigences .

La Matrice de relations peut être ouverte à partir de l'option du ruban « Conception > Paquetage > Paquetage / Matrice ». Sélectionnez si le Paquetage actuel est le Paquetage source, Paquetage cible ou les deux.

Lorsqu'une relation existe, une icône en forme de flèche s'affiche dans la cellule à l'intersection des éléments source et cible, la pointe de la flèche indiquant la direction de la relation. La matrice peut également être configurée pour mettre en surbrillance les lignes ou les colonnes qui n'ont aucune relation dans une couleur distincte. Cette option et d'autres peuvent être configurées dans la fenêtre Options, disponible à partir du bouton Options dans l'en-tête de la Matrice de relations.

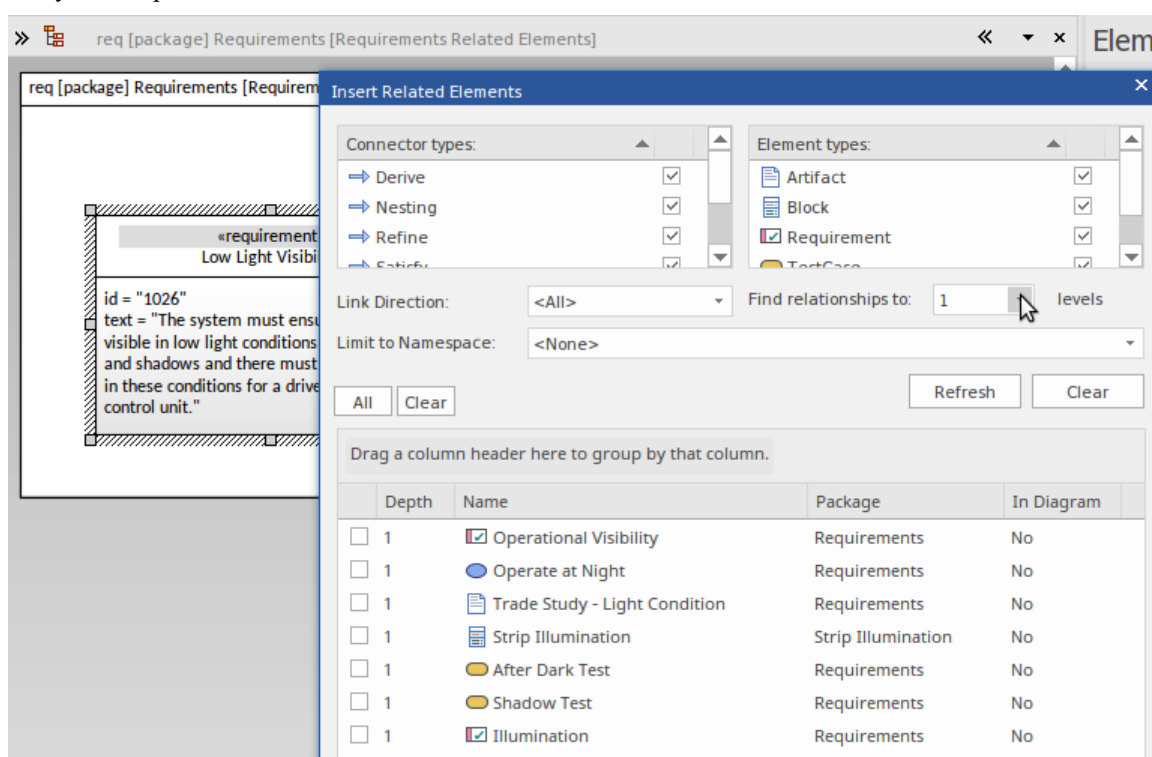




Ces options vous permettent de personnaliser la manière dont la matrice est affichée, notamment si les éléments sont triés et leurs noms sont préfixés par le nom Paquetage , et si les lignes et colonnes des éléments source et cible sans connexions sont mises en surbrillance. Pour plus d'informations, consultez la rubrique d'aide [Relationship Matrix](#) .

## Insérer des éléments connexes

La fonctionnalité Insérer des éléments associés est un outil de productivité qui permet à un ingénieur ou à un autre intervenant de construire rapidement un diagramme en insérant un élément central, puis en demandant à l'outil de rechercher tous les éléments liés à cet élément, jusqu'à une profondeur de connectivité arbitraire. Cela aide l'ingénieur à explorer efficacement le graphique des éléments et à créer un diagramme qui montre comment d'autres éléments du référentiel sont directement liés à cet élément central et comment d'autres éléments sont liés à ces éléments. Les types d'éléments et de connecteurs peuvent être spécifiés pour l'inclusion ou l'exclusion, et la profondeur peut être modifiée pour intégrer plus ou moins d'éléments et de connecteurs dans le diagramme . Le diagramme peut être automatiquement mis en page ou l'outil disposition diagramme peut être utilisé pour réorganiser la disposition afin de la rendre plus attrayante ou pertinente.

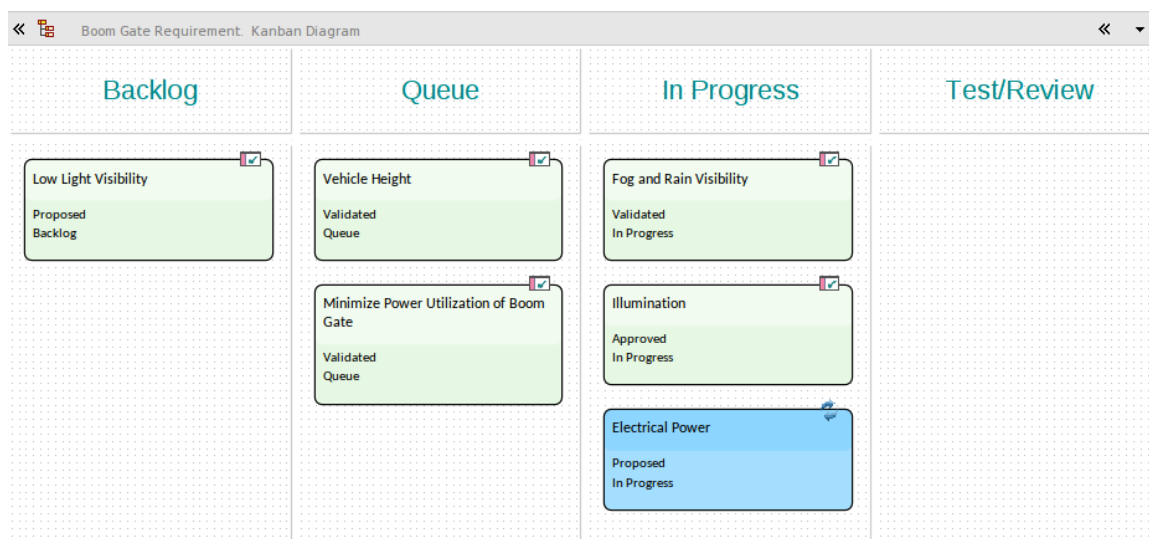


La fonctionnalité Insérer des éléments associés peut être utilisée en sélectionnant un élément dans un diagramme, puis en utilisant l'option de ruban « Conception > Élément > Ajouter un élément > Élément associé ».

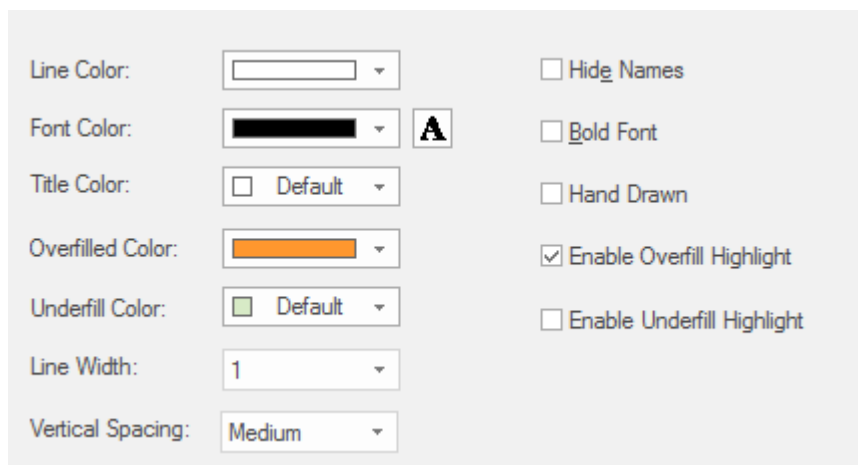
La fonctionnalité peut être utilisée comme un outil exploratoire où les diagrammes rendus sont construits dans le cadre d'un processus d'enquête et sont rejetés une fois qu'ils ont rempli leur fonction. Alternativement, la fonctionnalité peut être utilisée pour créer des diagrammes plus permanents qui peuvent être enregistrés et réutilisés pour la visualisation. Dans tous les cas, l'outil fera gagner du temps à l'ingénieur et permettra la création de diagrammes précis et expressifs qui impressionneront forcément les parties prenantes qui n'auraient autrement pas pu visualiser les connexions entre les éléments. Pour plus d'informations, consultez la [Insert Related Elements](#) Aide sujet.

## Diagrammes Kanban

Les diagrammes Kanban peuvent être utilisés pour visualiser les exigences (et d'autres éléments) au fil des étapes d'un processus ou d'une méthodologie standard ou propriétaire. Le concept de Kanban trouve son origine dans l'analyse de l'efficacité des processus dans l'industrie automobile au Japon à la fin du <sup>XXe</sup> siècle. Il a depuis été adopté par un certain nombre de communautés de logiciels et de systèmes comme un moyen efficace de gérer les exigences depuis le backlog ou la file d'attente jusqu'à leur mise en œuvre. Enterprise Architect dispose d'une implémentation complète et pragmatique de Kanban qui peut supporter un certain nombre de processus d'exigences différents.

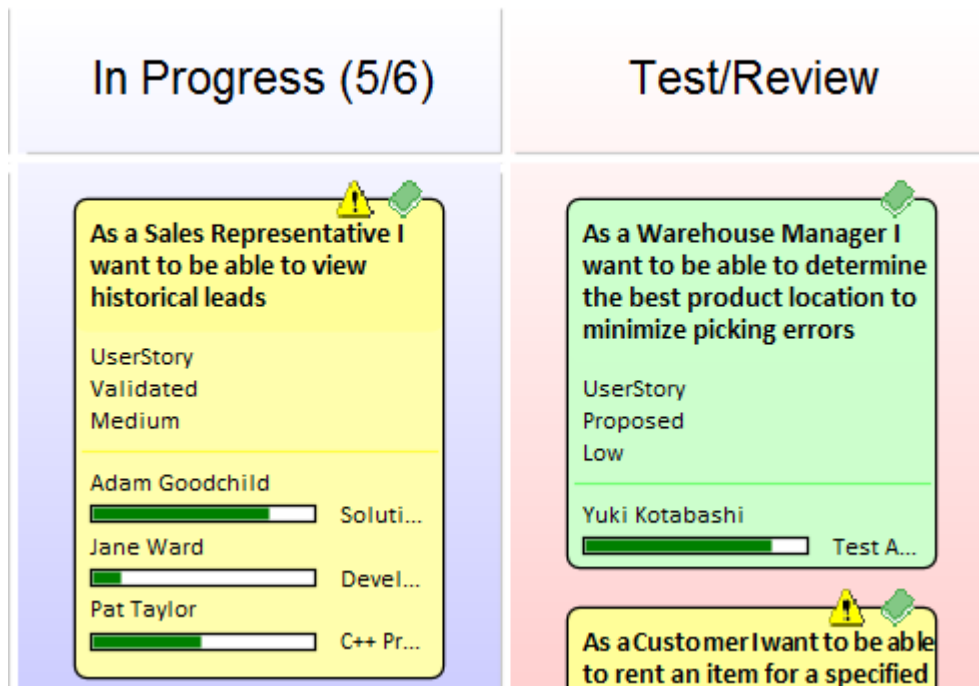


Dans ce diagramme nous voyons un certain nombre de colonnes qui représentent les étapes du processus d'exigences, ce qui permet de faire glisser les éléments entre les colonnes, généralement de gauche à droite, mais parfois, les éléments peuvent être renvoyés vers un Backlog, par exemple. Le diagramme est entièrement configurable par l'ingénieur, ce qui permet de configurer le nombre de colonnes et leurs noms ainsi qu'un large éventail d'autres aspects du diagramme, notamment Propriétés liées, les limites des travaux en cours et les couleurs, pour n'en citer que quelques-uns.



Il est également possible de montrer l'avancement d'une exigence particulière en appliquant des allocations de ressources

et en affichant chaque ressource sous forme de barre de progression indiquant le pourcentage d'achèvement de la tâche. Dans ce diagramme nous voyons un certain nombre d'éléments, dont l'un montre trois ressources travaillant sur le même élément. Pour plus d'informations, consultez le [Kanban Boards](#) Rubrique d'aide.



## Documenter Exigences

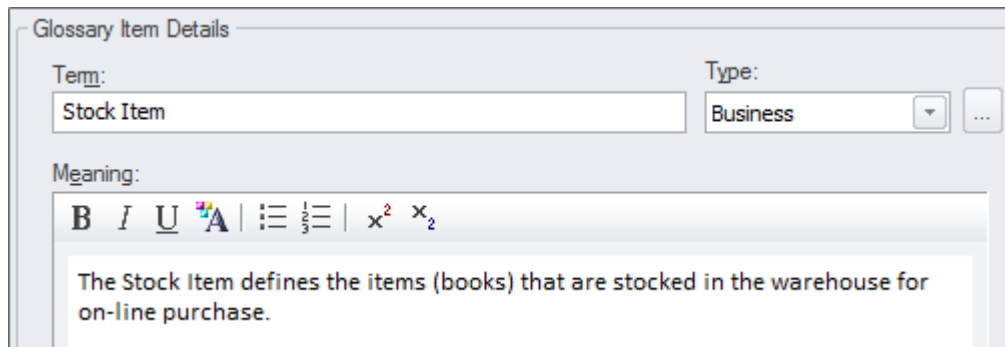
Un certain nombre de documents sont couramment produits dans le cadre de la discipline Exigences Engineering, tels que la Spécification Exigences logicielles (système) et Rapport de cas d'utilisation. Ceux-ci peuvent être générés automatiquement à partir d'un modèle d'exigences à l'aide gabarits intégrés. En outre, une large gamme d'autres documents peuvent être produits à l'aide gabarits intégrés ou personnalisés. La facilité de documentation d' Enterprise Architect est hautement configurable et de nombreux rapports peuvent être produits à l'aide du système gabarit, mais pour les rapports plus complexes, il existe une facilité appelée Documents virtuels qui permet à un éditeur de modéliser la structure du document et de sélectionner le contenu à partir de n'importe quel endroit du référentiel, en appliquant gabarits différents à chaque section du document. Il existe également une grande variété d'options qui peuvent être appliquées au niveau de la génération gabarit ou de documents, et le moteur Scriptant peut être utilisé pour injecter du contenu dans un document ou pour produire l'intégralité du rapport.



Il existe également une facilité Document Personnalisé qui permet à un modèleur de définir un document créé dynamiquement en faisant simplement glisser du contenu depuis le Navigateur sous forme d'éléments, diagrammes ou Paquetages et en appliquant un gabarit intégré ou défini par l'utilisateur à chaque élément qui spécifie comment le contenu sera rendu. Cela permet de visualiser un document et peut inclure tout contenu ou image fabriqué à la main en plus du contenu injecté depuis le Navigateur.

## Glossaire du Projet

Un Glossaire du Projet répertorie et définit les termes importants pour un projet ou un programme de travail. Le Glossaire du Projet peut être généré sous forme de document isolé ou inclus sous forme de section dans un ou plusieurs autres documents. Il fournit un point de référence unique pour les termes importants du projet et leur signification ; lorsque de nouveaux documents sont générés, les termes sont automatiquement mis à jour. Le Glossaire peut être généré au format DOCX ou PDF, ou au format HTML qui peut être inclus dans un site Web de projet ou d'organisation. Le Glossaire permet au modélisateur de classer les termes en types définis par l'utilisateur, et ceux-ci peuvent avoir des styles appliqués lorsqu'ils sont générés dans la documentation.



Glossary Item Details

Term:  Type:  ...

Meaning:

**B** *I* U A | 

- ☰
- ☷

 |  $x^2$   $x_2$

The Stock Item defines the items (books) that are stocked in the warehouse for on-line purchase.

Le Glossaire du Projet peut être consulté et géré à partir de cet emplacement du ruban :

Conception > Dictionnaire > Glossaire > Glossaire Vue

## Spécification logicielle requise

Ce document décrit les Exigences du système, son comportement dans des conditions définies et les contraintes auxquelles il doit faire face ; il sera généralement lu par diverses parties prenantes. Il existe un gabarit Exigences intégré qui peut être utilisé pour générer le document, bien que le modélisateur soit libre de créer un nouveau gabarit qui pourrait être basé sur celui-ci ou créé à partir d'un gabarit vierge. Lorsque le document a du contenu provenant de divers emplacements dans la fenêtre Navigateur , il serait plus opportun d'utiliser la facilité Documents virtuels, qui permet à l'utilisateur de créer un modèle du document (similaire à un document maître dans un traitement de texte) qui comporte un certain nombre de sections appelées Documents Modèle . Le contenu de ces derniers peut être choisi n'importe où dans la fenêtre Navigateur .

# Software Requirements Specification

Online Bookstore

Version 1.0 • Proposed

## Décrire les objectifs de l'utilisateur avec des cas d'utilisation

Les cas d'utilisation ont été conçus à l'origine par Ivar Jacobson, un ingénieur électricien suédois qui a également été un important promoteur du développement du Unified Modeling Language (UML) . Les cas d'utilisation sont utilisés comme méthode de représentation des exigences fonctionnelles du point de vue des utilisateurs. On dit qu'ils sont axés sur les objectifs, car le cas d'utilisation définit l'objectif que l'utilisateur essaie d'atteindre lors de son interaction avec le système. Enterprise Architect supporte le développement de diagrammes de cas d'utilisation, mais supporte également entièrement en charge la modélisation et la gestion du texte des cas d'utilisation ; il dispose d'un outil unique et très productif pour travailler avec les cas d'utilisation, appelé Scenario Builder.

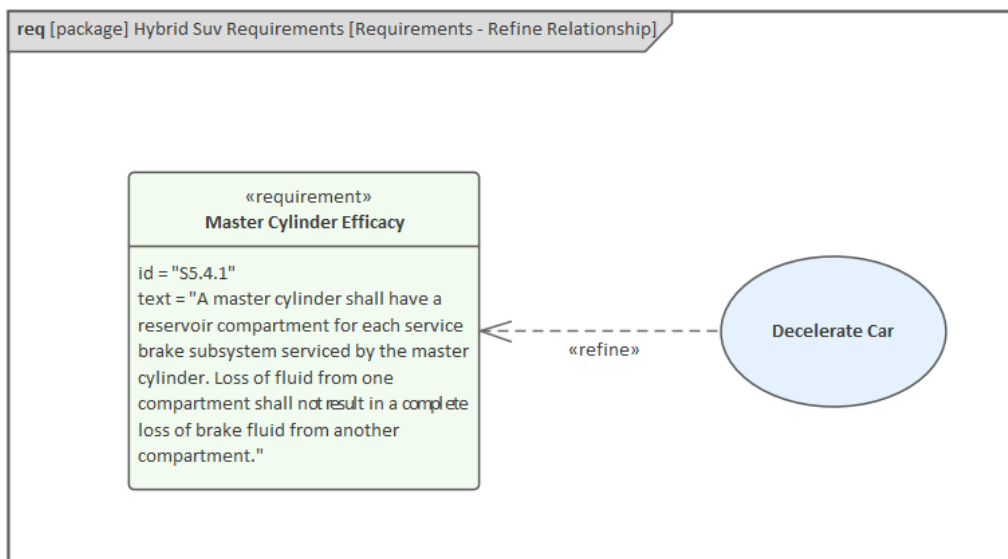
Cet outil innovant permet non seulement de modéliser des cas d'utilisation à n'importe quel niveau de détail, mais crée également automatiquement des modèles comportementaux qui permettent de visualiser les étapes détaillées d'un cas d'utilisation et l'interaction entre l'acteur et le système et de les relier à d'autres parties du modèle.

Le cas d'utilisation est un proche cousin du récit utilisateur, qui est utilisé dans un certain nombre de techniques de développement logiciel Agile. Le terme, inventé à l'origine en suédois, se traduit plus naturellement par scénario d'utilisation, ce qui fournit une explication plus convaincante de la méthode.

## Exigences et cas d'utilisation

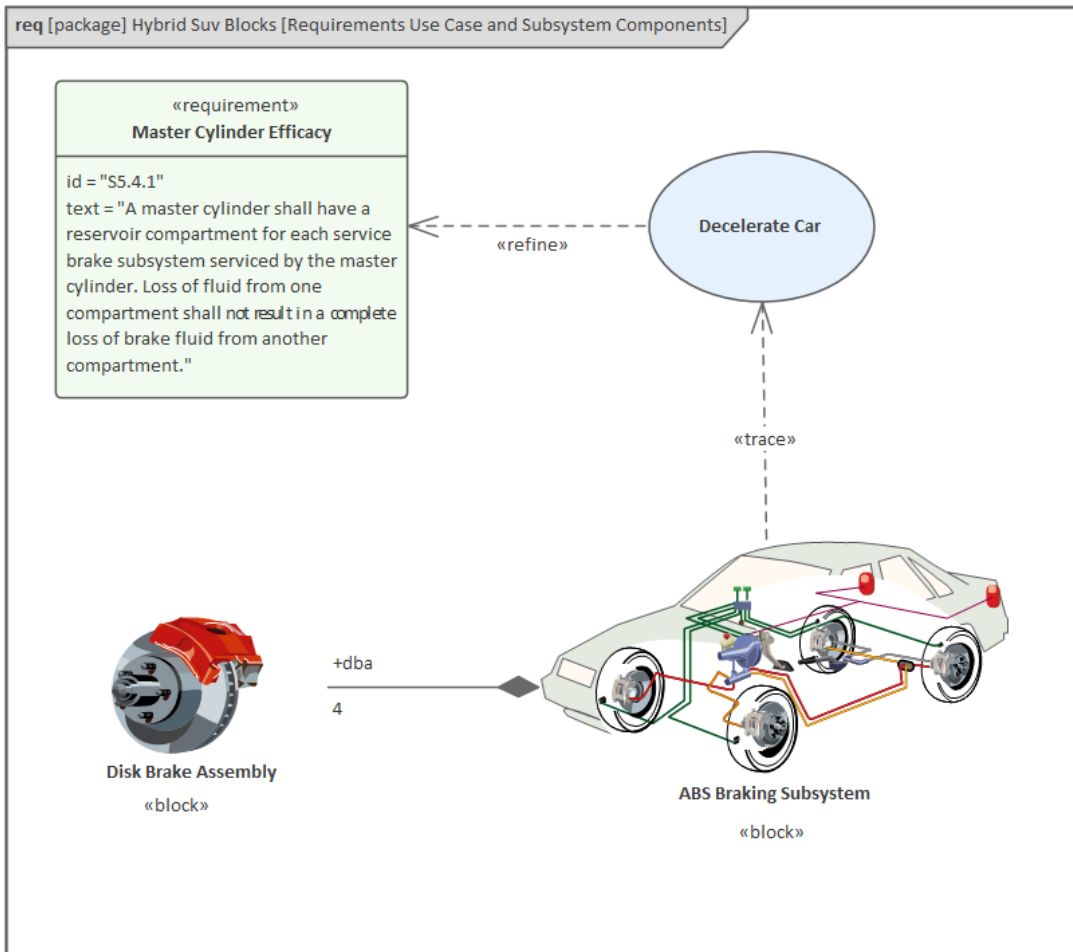
La technique des cas d'utilisation est fondamentalement très simple et a été conçue à l'origine pour garantir que les exigences fonctionnelles soient rédigées du point de vue de l'utilisateur. Ce point de vue a permis de garantir que les systèmes déployés seraient adaptés à l'objectif et acceptés par la communauté diversifiée d'utilisateurs. Il existe cependant une grande quantité de littérature contradictoire et un nombre tout aussi important de styles pour définir les cas d'utilisation. Cela a conduit à la confusion et à l'incertitude et a eu tendance à atténuer la valeur que l'on peut tirer de cette technique efficace et simple.

En ingénierie logicielle, de nombreuses méthodes préconisent l'utilisation de cas d'utilisation comme alternative au développement Exigences, car le Unified Modeling Language (UML) n'inclut pas d'élément d'exigence formel. En revanche, la plupart des méthodes Ingénierie Systèmes Modèles Basée utilisant SysML combinent l'application de cas d'utilisation et Exigences. Cela résulte du fait que SysML définit à la fois un cas d'utilisation et un élément d'exigence, de sorte que ces deux éléments peuvent être liés l'un à l'autre et compléter la spécification du système pour apporter clarté et précision à l'importante discipline de l'ingénierie et de la gestion des exigences.

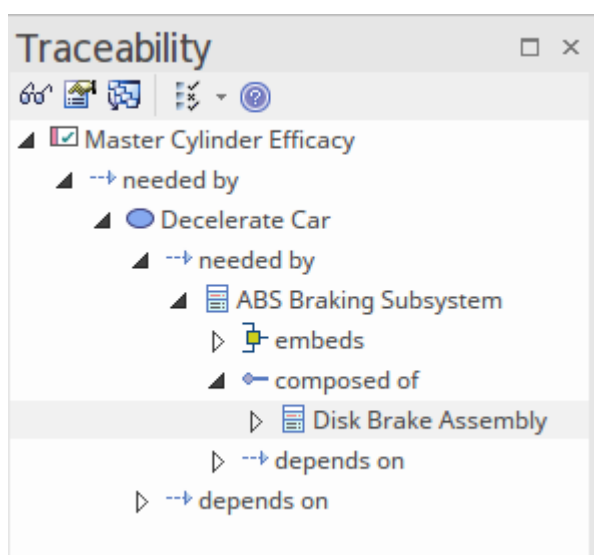


Dans ces deux diagrammes le modélisateur a utilisé la relation <<affiner>> pour indiquer que le cas d'utilisation de *décélération de la voiture* affine ou ajoute une explication supplémentaire pour clarifier l'exigence *d'efficacité du maître-cylindre*. Cela fournit un mécanisme permettant de retracer les composants de niveau d'implémentation qui sont connectés au cas d'utilisation jusqu'à l'exigence et finalement jusqu'à la partie prenante.





La fenêtre Traçabilité peut également être utilisée pour visualiser les connexions entre les éléments du modèle à différents niveaux d'abstraction et pour voir la connexion d'un Bloc qui fait partie d'un assemblage de sous-système à l'exigence qui a spécifié la fonctionnalité.



Les cas d'utilisation sont généralement utilisés pour affiner les exigences de haut niveau et pour exprimer la communication et l'interaction entre l'utilisateur et le système.

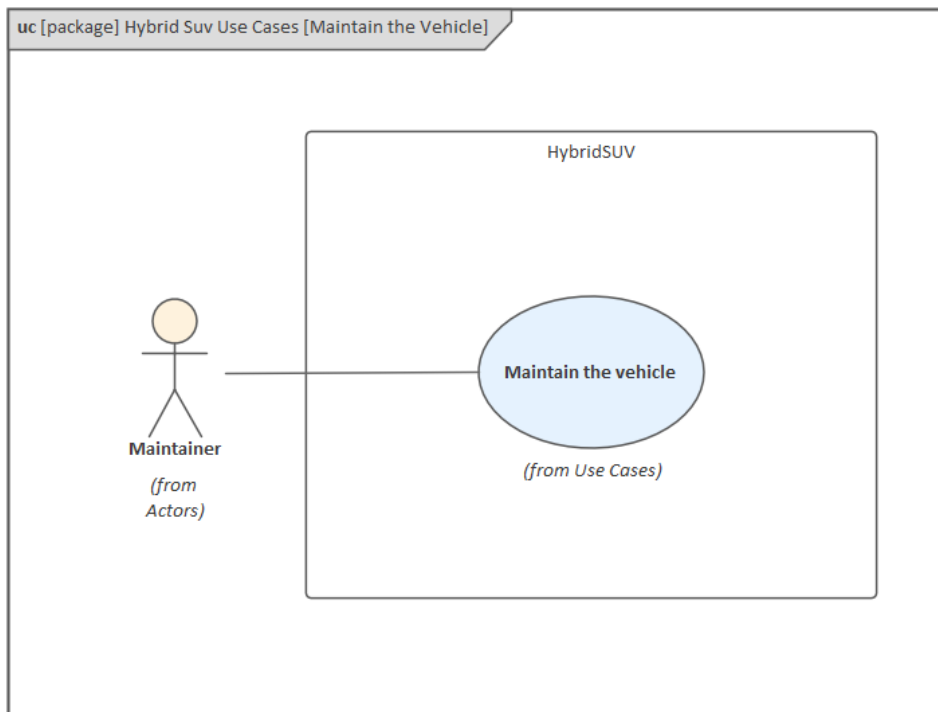


## Présentation Diagrammes de cas d'utilisation

Le diagramme de cas d'utilisation est un diagramme simple qui décrit visuellement les objectifs des utilisateurs par rapport au système ou à une partie du système. Cela pourrait être paraphrasé par « la valeur que le système fournit aux acteurs ». Les diagrammes de cas d'utilisation semblent assez simplistes, avec un petit nombre d'éléments :

- Sujet
- Acteurs
- Cas d'utilisation

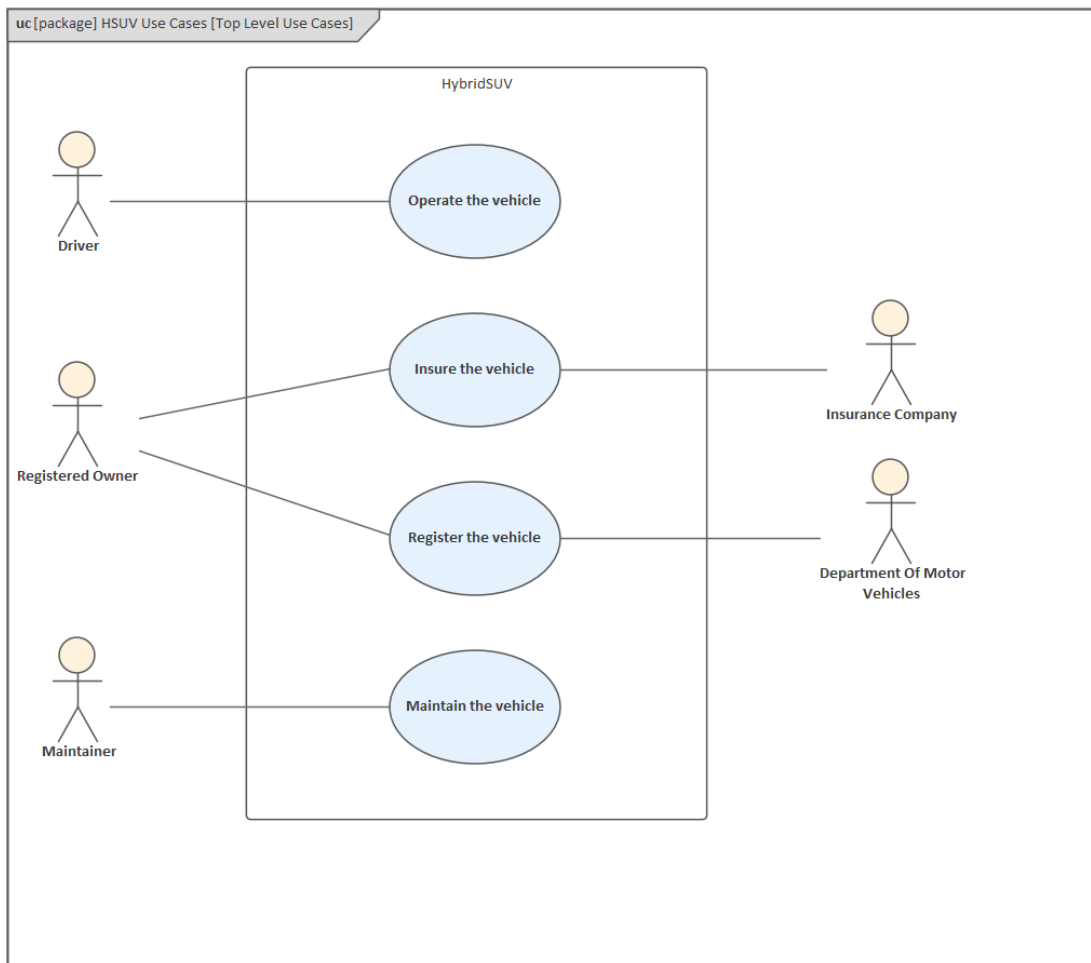
Ceux-ci sont reliés par une série de relations.



Le sujet ( bordure ) fournit un contexte pour la définition et représente un système ou une partie d'un système ; les acteurs se situent par définition à l'extérieur du sujet et les cas d'utilisation à l'intérieur. La relation de chemin de communication traverse par définition le périmètre du sujet car elle relie un acteur à un cas d'utilisation. Là encore, le nombre de relations est assez limité, mais chacune a une signification spécifique dans le diagramme .

- Chemin de Communication
- Prolonger (également avec condition)
- Inclure
- Généralisation

Comme pour tous les éléments SysML, les éléments présentent à la fois un aspect graphique et textuel, et dans la description des cas d'utilisation, l'accent est généralement davantage mis sur l'aspect textuel ou narratif.



Il est possible de créer un nombre illimité de diagrammes de cas d'utilisation pour représenter l'interaction des utilisateurs avec le système ou une partie d'un système. Il est important de comprendre que les cas d'utilisation sont destinés à décrire la valeur le système fournit à ses utilisateurs et qu'ils ne sont pas destinés à être décomposés par décomposition fonctionnelle. Il s'agit sans aucun doute de l'erreur la plus courante commise par les modélisateurs novices, ce qui a pour conséquence d'atténuer les avantages considérables que l'on peut tirer de cette technique.

Le modèle de cas d'utilisation peut être enrichi par un mécanisme appelé « structurer le Modèle de cas d'utilisation », qui élimine le texte répétitif, classe les acteurs et les cas d'utilisation et spécifie les points d'extension. Ce mécanisme sera abordé plus loin dans ce chapitre. Pour plus d'informations, consultez la rubrique d'aide [SysML Use Case Models](#).

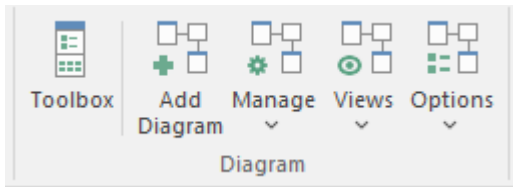
## Création d'un Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation peut être créé à partir de plusieurs emplacements dans l'Interface Utilisateur en sélectionnant :

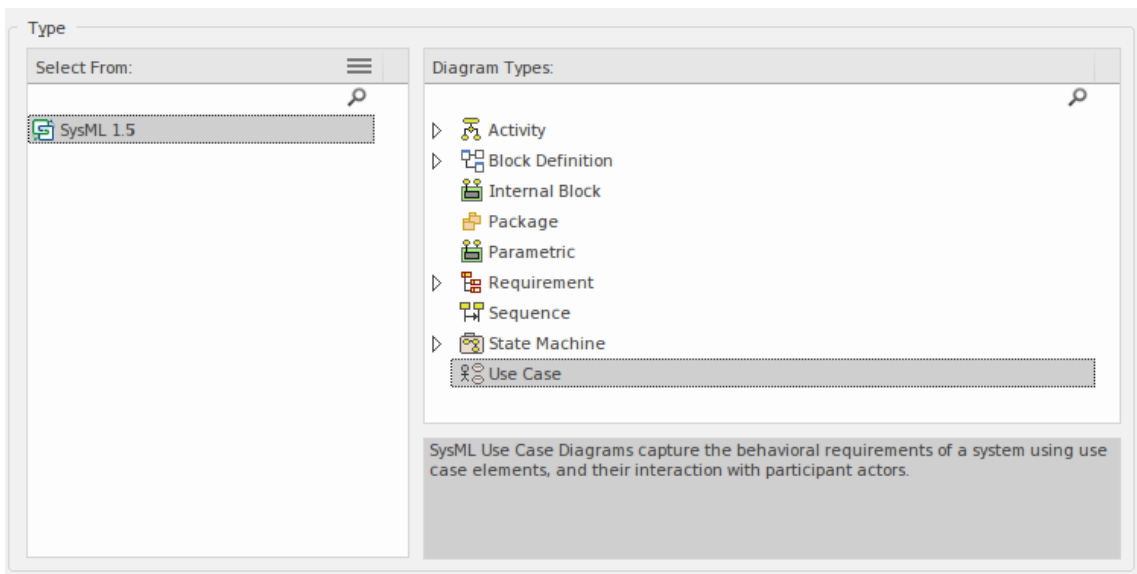
- Ruban de conception - *Ajouter une icône Diagramme* sur le panneau *Diagramme*
- Barre d'outils de la fenêtre Navigateur - *Icône Nouveau Diagramme*
- Menu contextuel de la fenêtre Navigateur - *Ajouter Diagramme*

Nous allons utiliser le ruban Conception pour créer un diagramme de cas d'utilisation. Tout d'abord, sélectionnez l'emplacement dans la fenêtre du Navigateur où vous souhaitez placer le diagramme de cas d'utilisation. Comme pour tous diagrammes, il peut s'agir soit d'un Paquetage, soit d'un élément, mais il est courant d'insérer diagrammes de cas d'utilisation dans un Paquetage. Une fois l'emplacement Paquetage sélectionné dans la fenêtre Navigateur, sélectionnez :

Conception > Diagramme > Ajouter Diagramme



La sélection de cette option ouvrira la dialogue *Nouveau Diagramme* , qui vous permettra de nommer le diagramme ; le nom par défaut sera celui du Paquetage ou de l'élément qui contient le diagramme . Avec la perspective SysML choisie et la version de SysML sélectionnée, une liste de diagrammes s'affichera, vous permettant de choisir le diagramme de cas d'utilisation. Cliquez sur le bouton OK pour créer un nouveau diagramme de cas d'utilisation à l'emplacement sélectionné dans la fenêtre Navigateur . La Vue Diagramme s'ouvrira, vous permettant de commencer à ajouter des éléments et des connecteurs qui décrivent la valeur que le système fournira à ses utilisateurs. Enterprise Architect affichera également les pages « Cas d'utilisation » de la boîte à outils Diagramme qui contiennent les éléments et les relations définis par la spécification SysML comme applicables à la construction diagrammes de cas d'utilisation. N'importe quel nombre d'autres pages de la boîte à outils peut être ouvert, si nécessaire, en plus des pages de la boîte à outils *Éléments communs* et *Relations communes* qui seront toujours disponibles.



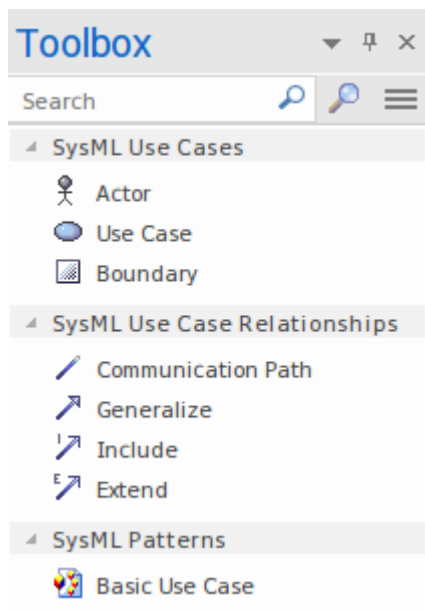
Les éléments et connecteurs les plus importants utilisés avec le diagramme de cas d'utilisation sont :

### Éléments

- Acteur - utilisé pour décrire le rôle joué par un utilisateur par rapport au système
- Cas d'utilisation - utilisé pour décrire la valeur qu'un système fournit à ses utilisateurs
- Bordure - utilisée pour montrer la portée des systèmes (généralement une par diagramme )

### Connecteurs

- Chemin Communication - utilisé pour connecter les acteurs aux cas d'utilisation
- Généraliser - utilisé entre deux acteurs ou entre deux cas d'utilisation
- Inclure - utilisé entre les cas d'utilisation pour réutiliser les étapes du scénario
- Étendre - utilisé pour embellir un cas d'utilisation avec des détails supplémentaires

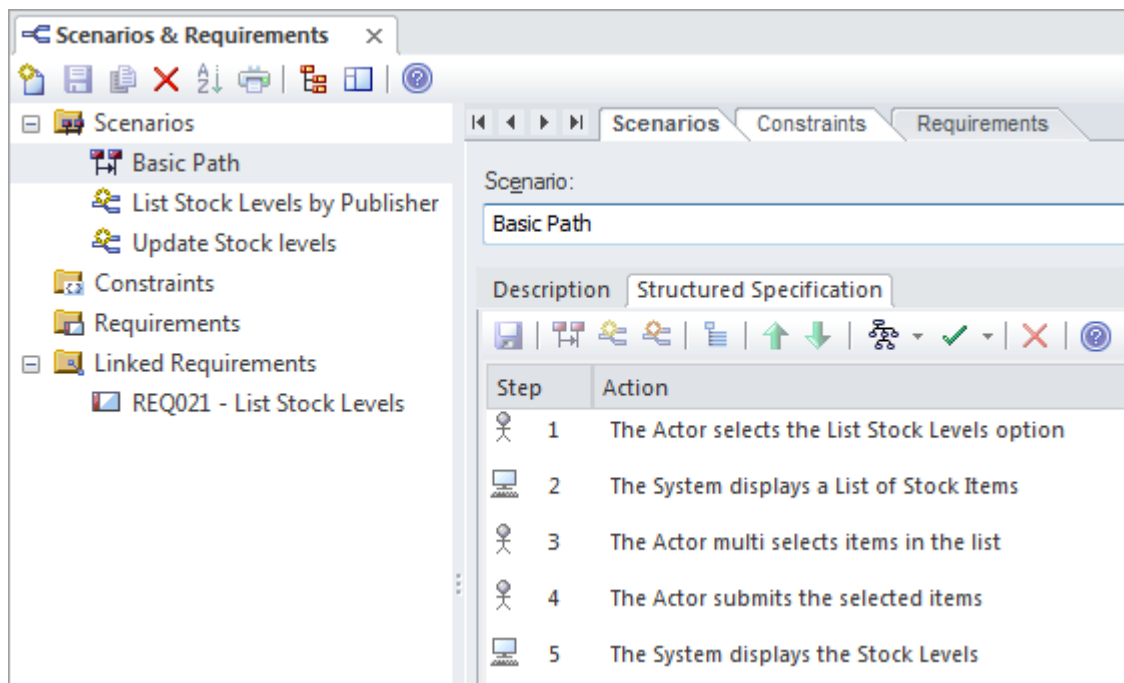


Les éléments peuvent être ajoutés au diagramme en les faisant glisser de la boîte à outils vers la Vue Diagramme . Il est considéré comme une bonne pratique de commencer par un élément Bordure , qui doit être nommé de manière appropriée pour décrire le système, le sous-système ou l'entité modélisée par le diagramme de cas d'utilisation. Laisser le nom vide ou lui donner un nom qui n'indique pas clairement au lecteur quel système ou partie de système est modélisé peut conduire à une mauvaise interprétation du diagramme . Une fois la Bordure ajoutée et dimensionnée de manière appropriée dans le diagramme , des acteurs et des cas d'utilisation peuvent être ajoutés - les acteurs étant positionnés à l'extérieur de la Bordure et les cas d'utilisation à l'intérieur. L'étape suivante consiste à ajouter des relations de chemin Communication entre les acteurs et les cas d'utilisation, définissant ainsi la valeur que les acteurs dérivent du système.

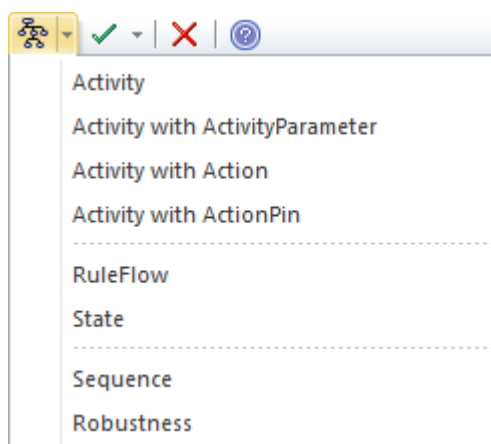
Une fois qu'un diagramme de base a été créé, et que la connaissance du domaine et des comportements du système est révélée, il est possible de structurer ou d'embellir le diagramme en utilisant les relations supplémentaires Inclure, Étendre et Généraliser. Le nouveau venu est mis en garde contre l'utilisation trop libérale de ces relations, et toute tentative d'utilisation de la décomposition fonctionnelle atténuera la valeur du modèle de cas d'utilisation, qui est intentionnellement large dans sa description pour permettre aux parties prenantes d'obtenir une vue à 10 000 mètres des services fournis par le système, le sous-système ou l'entité modélisée.

## Découvrez le créateur de scénarios

Il existe une large gamme d'outils pour travailler avec les cas d'utilisation, mais aucun n'est plus important et utile que le générateur de scénarios. Cet outil unique comble le fossé entre ce qui a été traditionnellement fait dans les traitements de texte ou les outils isolés qui séparent le diagramme de cas d'utilisation avec ses acteurs et ses cas d'utilisation, et les étapes des scénarios.



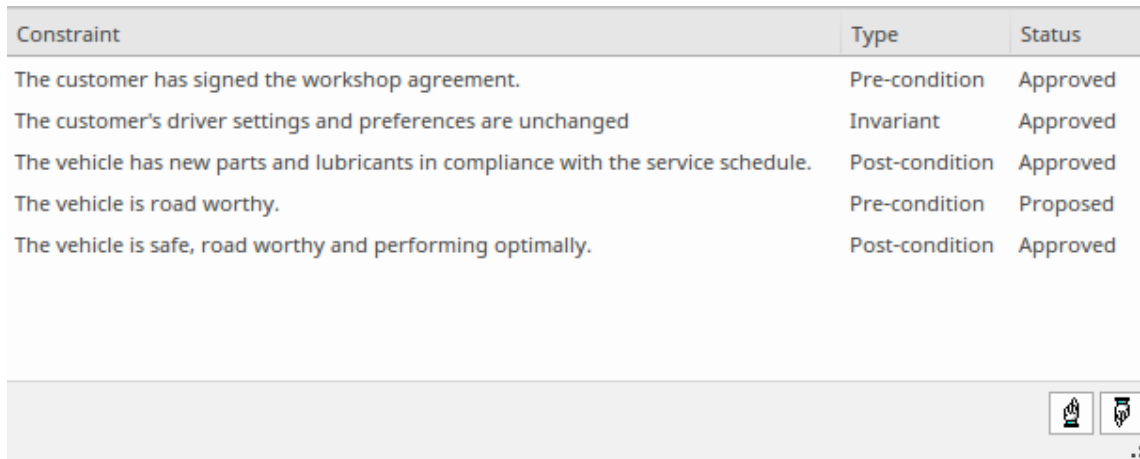
Le générateur de scénarios fournit également un mécanisme permettant de générer automatiquement des modèles comportementaux directement à partir des étapes du scénario, permettant ainsi de relier les éléments de l'architecture et de la conception à des étapes individuelles.



L'outil propose toute une gamme d'options qui conviendront à tout cas d'utilisation ou processus Exigences, allant de ce que l'on appelle communément un processus de base où l'acteur et les cas d'utilisation sont nommés et le cas d'utilisation est décrit, à un processus partiellement habillé où le flux de base est complété. Un processus entièrement habillé détaillera généralement toutes les étapes du flux de base et définira et détaillera les étapes des scénarios alternatifs et d'exception. Pour plus d'informations, consultez la rubrique [Scenario Builder](#).

De plus, n'importe quel nombre de contraintes peuvent être ajoutées, telles que des pré-conditions, des post-conditions et des invariants.

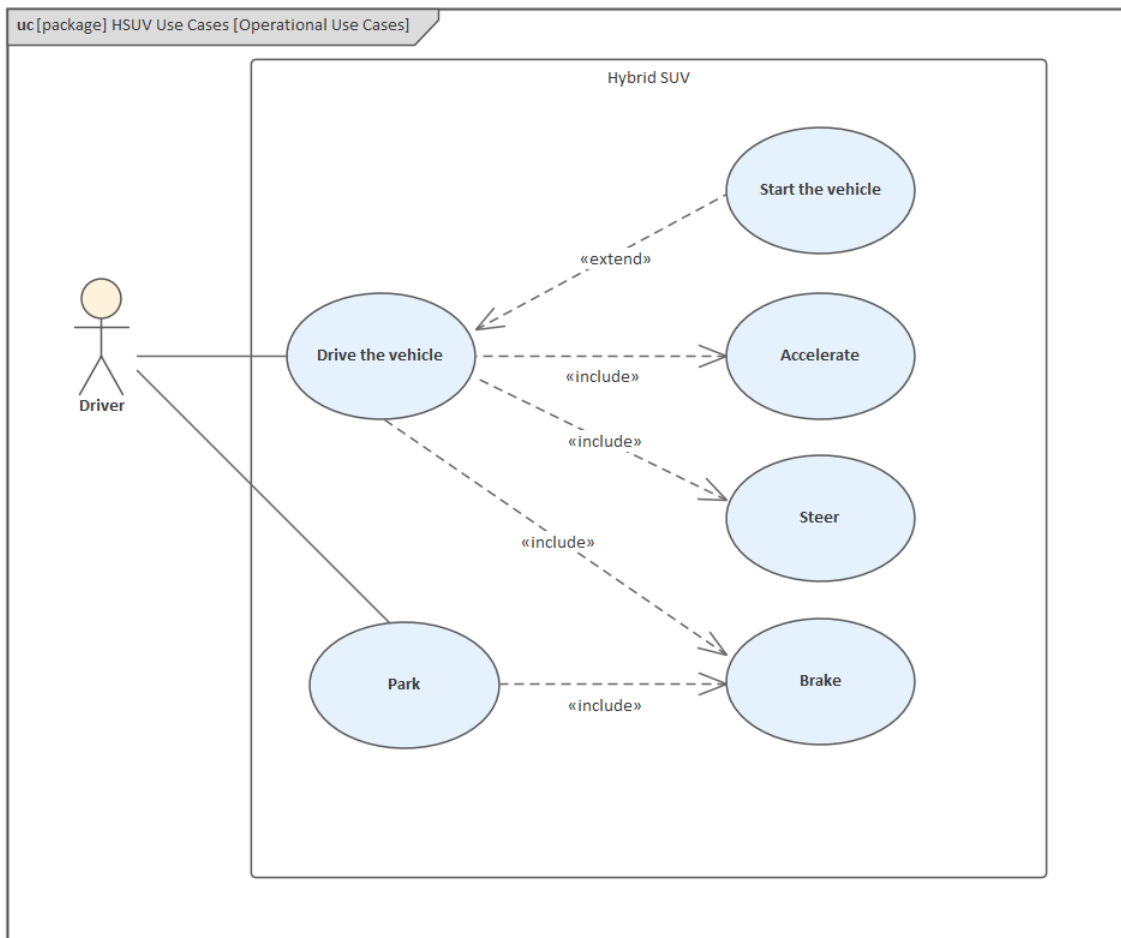
Constraint	Type	Status
The customer has signed the workshop agreement.	Pre-condition	Approved
The customer's driver settings and preferences are unchanged	Invariant	Approved
The vehicle has new parts and lubricants in compliance with the service schedule.	Post-condition	Approved
The vehicle is road worthy.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





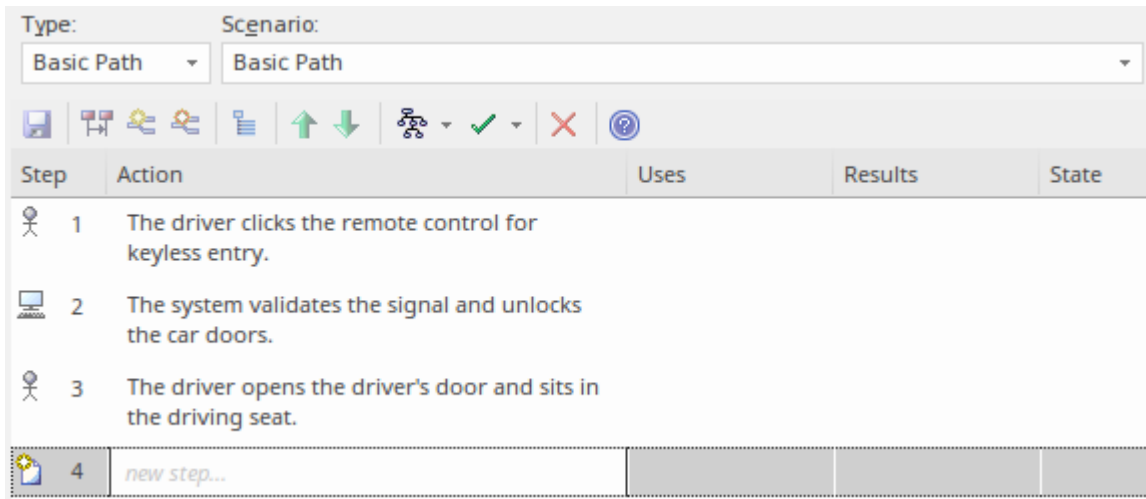
## Structurer un Modèle de cas d'utilisation

Bien que le modèle de cas d'utilisation offre un niveau élevé de visualisation et qu'il soit déconseillé aux ingénieurs système d'appliquer une décomposition fonctionnelle, SysML fournit un certain nombre de mécanismes qui peuvent aider à structurer un modèle de cas d'utilisation afin de garantir que des éléments de fonctionnalité distincts peuvent être réutilisés. Ces mécanismes se composent de la relation de dépendance <<include>>, de dépendance <<extend>> et de généralisation.



## Générer Diagrammes de comportement

Enterprise Architect dispose d'un outil de productivité très utile qui permet de générer automatiquement diagrammes de comportement à partir des spécifications de cas d'utilisation définies dans le générateur de scénarios. Cela permet de visualiser ces descriptions textuelles. Il permet également d'établir des relations entre les étapes d'une description de cas d'utilisation et d'autres éléments modélisation .



The screenshot shows the Scenario Editor interface in Enterprise Architect. At the top, there are two dropdown menus: 'Type:' set to 'Basic Path' and 'Scenario:' set to 'Basic Path'. Below these are several icons for editing the scenario, including a save icon, a refresh icon, a list icon, up and down arrows, a diagram icon, a checkmark, a red X, and a help icon. The main area is a table with the following columns: Step, Action, Uses, Results, and State.

Step	Action	Uses	Results	State
1	The driver clicks the remote control for keyless entry.			
2	The system validates the signal and unlocks the car doors.			
3	The driver opens the driver's door and sits in the driving seat.			
4	<i>new step...</i>			

## Rapport de cas d'utilisation

La création de la documentation des cas d'utilisation est traditionnellement un processus manuel et, dans de nombreux projets, les documents s'étalant sur des centaines de pages, leur production consomme de précieuses ressources. Ces documents élaborés à la main deviennent difficiles à maintenir et restent isolés des autres parties du projet telles que Exigences, Règles Métier et les composants de la solution. Enterprise Architect dispose d'un outil multifonction appelé Scenario Builder qui permet au modélisateur de spécifier des cas d'utilisation et des scénarios à l'intérieur du modèle et ceux-ci peuvent être générés automatiquement dans une documentation de haute qualité à l'aide de gabarits intégrés. Il existe deux gabarits intégrés qui peuvent être utilisés pour générer un rapport de cas d'utilisation : l'un documente le cas d'utilisation à un niveau récapitulatif et l'autre à un niveau détaillé.



### Exemple de contenu d'un Rapport de cas d'utilisation

Le rapport détaillé sur les cas d'utilisation répertorie tous les détails du cas d'utilisation et les étapes détaillées, y compris les chemins de base, les scénarios alternatifs et d'exception. D'autres informations, notamment Exigences internes, les conditions préalables et postérieures et d'autres contraintes, seront également incluses dans le rapport. Si un diagramme Comportementale tel qu'un diagramme d'activité a été créé automatiquement, ce diagramme sera également affiché dans le rapport.

**Altemate. 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

Page 3 of 4

Use Case Details

19 May, 2015

**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:

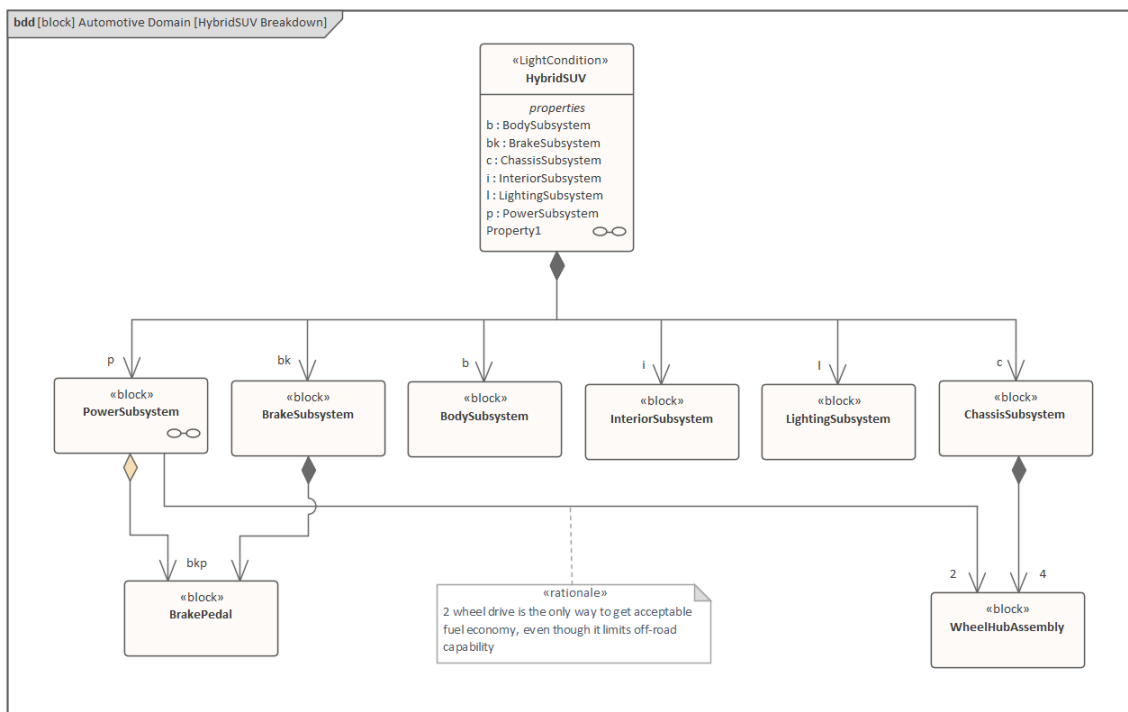
**4. System returns a listing of titles and quantity in stock for the publisher**

Uses:

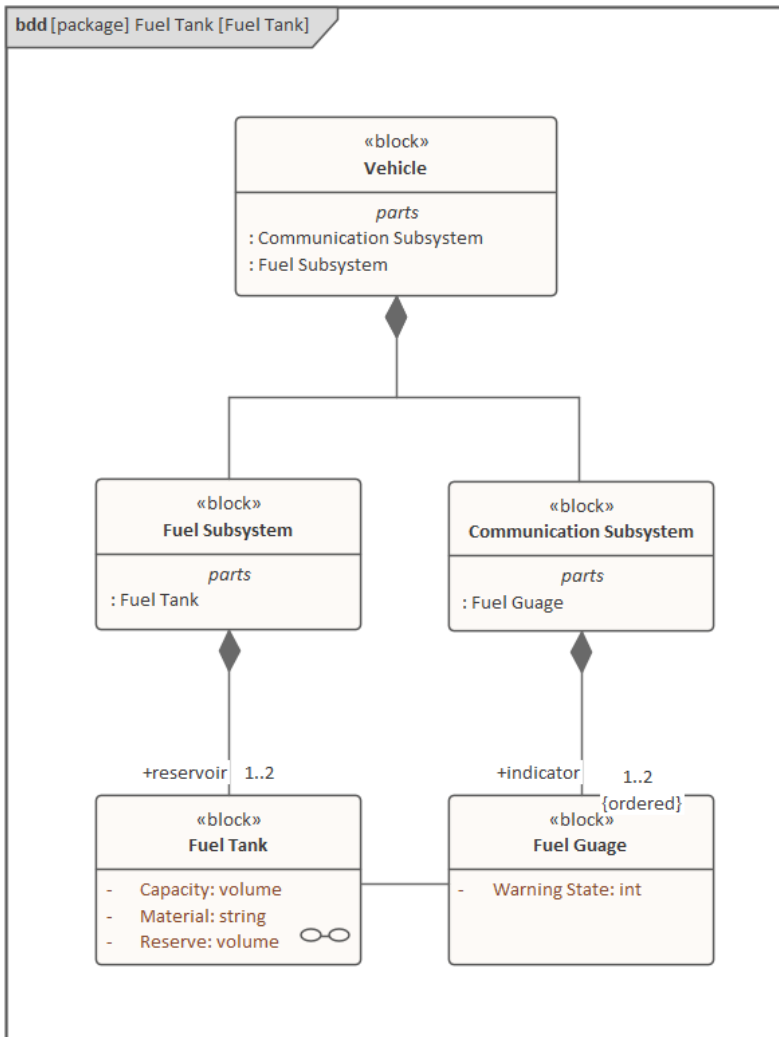
## Utilisation de blocs pour Modèle la structure et les contraintes

Les constructions et expressions du langage dans SysML, comme dans nos langues naturelles, peuvent être divisées en types structurels et comportementaux. Dans des langues comme l'anglais, l'allemand ou le japonais, les noms décrivent la structure et les verbes décrivent le comportement. Les phrases contiennent généralement une combinaison de noms et de verbes qui mettent en lumière un aspect du monde du locuteur. Le SysML a une division similaire, avec des éléments qui décrivent la structure et d'autres éléments qui décrivent le comportement. Dans le SysML, les éléments structurels (noms) sont décrits à l'aide d'un Bloc. Lorsque les ingénieurs créent des diagrammes, ils auront souvent un mélange d'éléments de comportement ou de structure et ils décriront un aspect particulier d'un système - mettant en lumière un aspect du système modélisé.

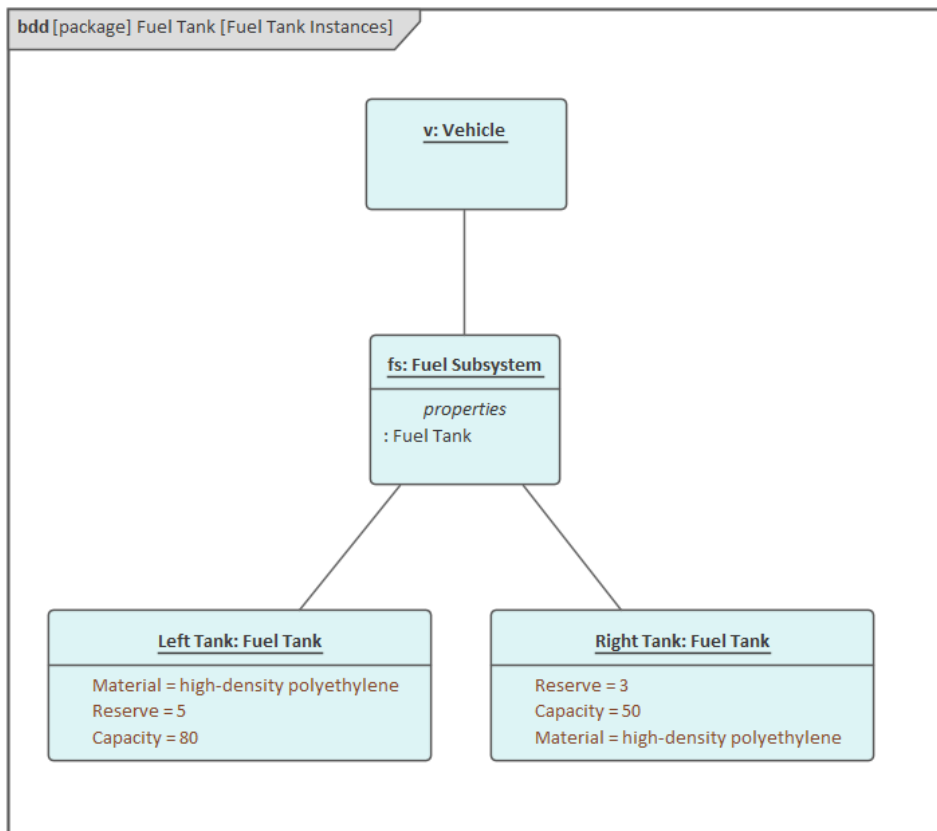
Le Bloc est l'unité fondamentale de la structure d'un système. Il peut être utilisé pour décrire un système entier, un sous-système, un composant, un élément qui circule dans un système, une contrainte ou des entités qui résident à l'extérieur d'un système. De la même manière que nos langues naturelles, un Bloc peut représenter quelque chose d'abstrait, de logique ou de physique. Il s'agit d'un concept important, et les auteurs et lecteurs du SysML doivent être clairs quant à l'intention de la représentation. Par exemple, dans une architecture logique, il y a généralement des Blocs représentant des idées ou des conceptions conceptuelles qui, au moment de la conception détaillée et de la construction, peuvent être réalisées par des composants physiques et tangibles. Un architecte système peut définir un Bloc appelé *Sous-système de détection de collision* qui est une expression d'un composant système logique qui pourrait, lors de la phase de conception détaillée, être en partie réalisé par un ensemble d'émetteurs radar et laser, de détecteurs et de caméras.



Un certain nombre de nos langues naturelles possèdent un terme grammatical appelé *classificateurs*, qui regroupe les éléments (noms) d'un lexique en classes d'éléments qui partagent des caractéristiques et un comportement communs. Ce même principe s'applique aux blocs, qui sont essentiellement un type de classificateur qui regroupe une collection d'instances qui partagent les mêmes fonctionnalités structurelles et comportementales. Les instances d'un Bloc peuvent être modélisées de manière générique ou on peut leur attribuer des valeurs précises, comme le volume d'essence contenu dans un réservoir à un moment particulier d'un trajet ou au moment d'un accident.



Dans le diagramme des réservoirs de carburant, la voiture est modélisée comme un niveau de classificateur ( Bloc ), où le modèle décrit un véhicule générique et représente le fait qu'un véhicule peut avoir un ou deux réservoirs de carburant au maximum. Ce diagramme des instances de réservoirs de carburant décrit cependant un véhicule particulier qui possède deux réservoirs de carburant ayant des capacités et des volumes de réserve différents.

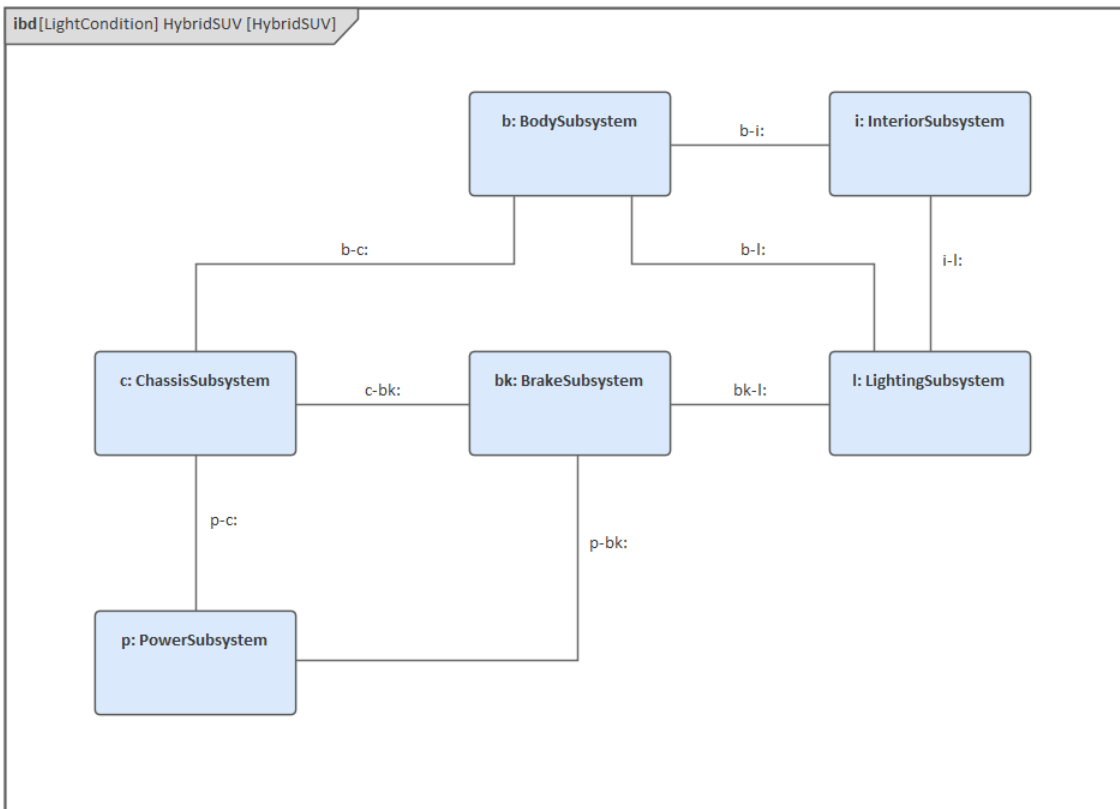


Un Bloc définit une collection (ou un ensemble) de fonctionnalités qui sont utilisées pour décrire un système, un sous-système, un composant ou tout autre élément d'intérêt. Ces fonctionnalités peuvent inclure des fonctionnalités structurelles et comportementales, telles que des propriétés, des opérations et des réceptions, pour représenter l'état du système et le comportement que le système est capable de présenter.

Enterprise Architect dispose d'un ensemble d'outils qui aident l'ingénieur système à travailler avec des blocs et à visualiser la structure et le comportement de ces éléments essentiels dans la définition d'un système. Ces facilités comprennent :

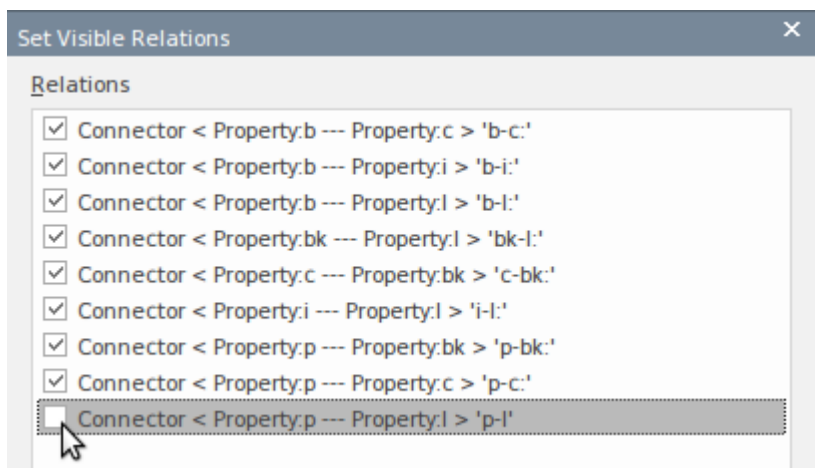
- Interrompre lorsqu'une Variable Change de Valeur , qui décrit les Blocs, leurs fonctionnalités , les points d'interaction et les relations structurelles
- Diagramme Interne de Bloc , qui capture la structure interne d'un Bloc en termes de propriétés et de connecteurs entre les propriétés

Ce Diagramme Interne de Bloc montre comment un certain nombre de sous-systèmes coopèrent pour créer la structure du véhicule. Par exemple, le sous-système d'éclairage est connecté au sous-système intérieur, qui à son tour est connecté au sous-système de carrosserie.



Certaines relations ont été supprimées dans le diagramme . Par exemple, le sous-système d'alimentation électrique aurait généralement une connexion avec le sous-système d'éclairage. Ce point est important, car les nouveaux venus dans SysML et Enterprise Architect pensent souvent que chaque relation définie doit être affichée dans un diagramme . Bien que cette affirmation semble vraie, il est important de se rappeler qu'un modélisateur, comme un dessinateur créant une caricature, laissera souvent des détails de côté dans les illustrations pour concentrer l'attention du spectateur sur d'autres éléments et connecteurs subjectivement plus importants.

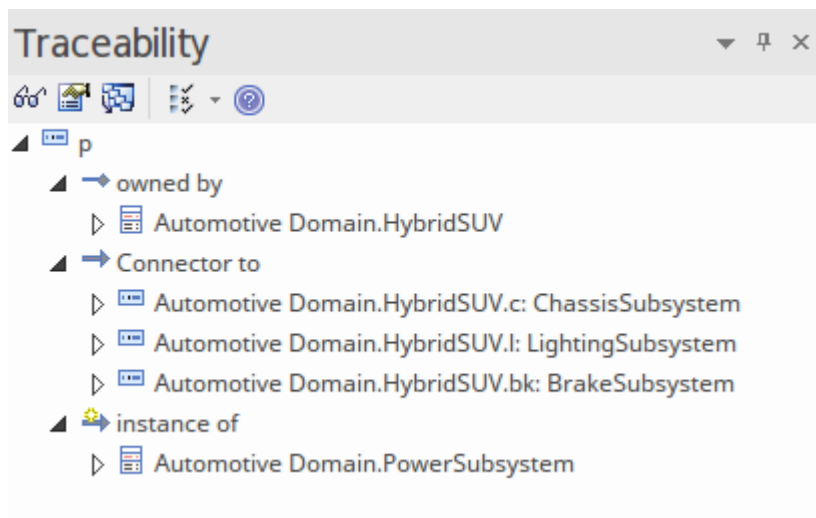
Cette capture d'écran montre comment un ingénieur peut définir les relations visibles pour un diagramme .



Si un connecteur n'est pas coché dans cette dialogue il ne sera pas affiché dans le diagramme actuel. Il peut cependant être visible dans d'autres diagrammes où les éléments connectés sont affichés. Cela peut être défini à partir de « Disposition > Diagramme > Apparence > Visibilité » > Option de ruban « Définir Relations visibles ».

Quels que soient les connecteurs affichés dans un diagramme , un modélisateur peut toujours visualiser tous les connecteurs d'un élément en sélectionnant l'élément dans le diagramme et en affichant la fenêtre de traçabilité. Dans cette capture d'écran, le sous-système d'alimentation a été sélectionné et même si le connecteur entre le sous-système d'alimentation et le sous-système d'éclairage a été défini sur « non visible » dans le diagramme , la relation est visible dans la fenêtre de traçabilité.





Les fonctionnalités d'un Bloc sont soit structurelles, soit comportementales.

Les fonctionnalités structurelles sont de trois types :

- **Pièces** - qui décrivent la composition d'un Bloc ; par exemple, que le châssis d'un véhicule est composé de deux essieux et de quatre ensembles de roues
- **Références** - qui décrivent la relation du Bloc avec d'autres Blocs (y compris lui-même) ; par exemple, qu'un train métropolitain a une relation avec une gare et un système de câblage aérien
- **Valeurs** - qui décrivent les aspects quantifiables d'un Bloc ; par exemple, des éléments tels que les dimensions, la température et la luminosité

Les fonctionnalités Comportementale incluent :

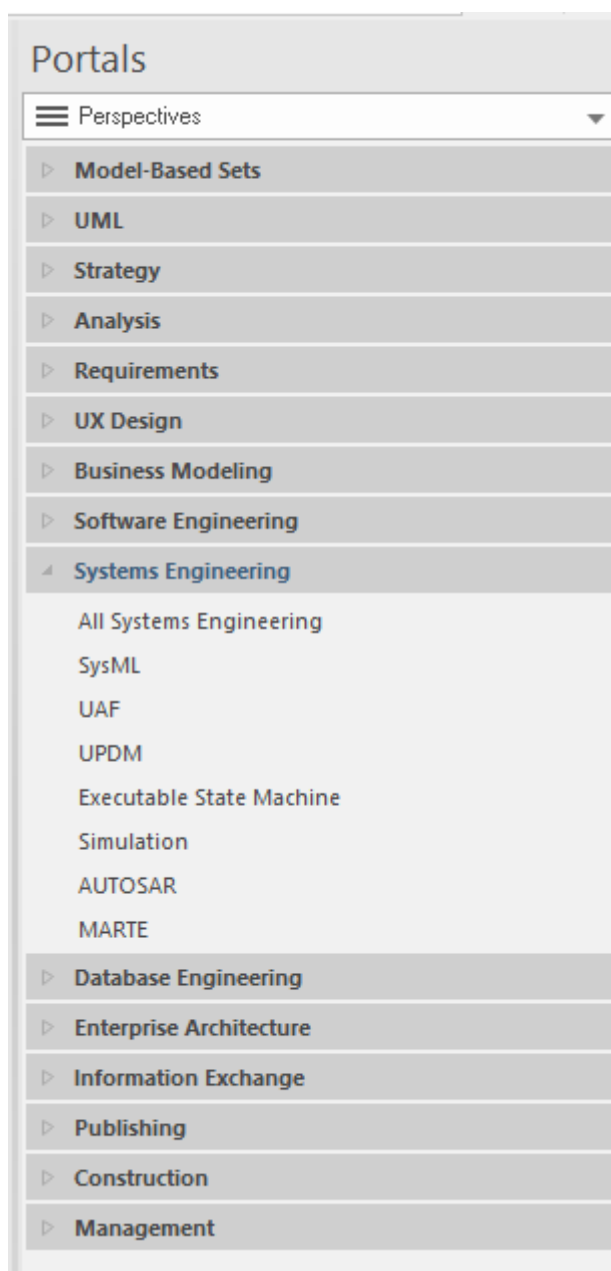
- **Opérations** - représentant généralement des requêtes synchrones
- **Réceptions** - représentant des requêtes asynchrones à partir d'un signal

## Démarrage avec des blocs

Les concepteurs d' Enterprise Architect ont créé une grande flexibilité pour les ingénieurs systèmes et d'autres disciplines, reconnaissant que les membres de l'équipe exercent souvent une variété de rôles et doivent changer de casquette plusieurs fois au cours d'un projet, voire au cours d'une même journée. Perspectives et espaces de travail offrent un moyen pratique et efficace pour un ingénieur système de changer de rôle efficacement sans avoir à lancer un autre outil.

### Définition de la perspective et de l'espace de travail

Les ingénieurs systèmes qui travaillent avec l'outil depuis un certain temps auront sélectionné une perspective dans l'ensemble de perspectives d'Ingénierie des Systèmes ; il s'agit généralement de la perspective SysML, leur donnant accès à tous les motifs et pages de la boîte à outils pour créer l'un des diagrammes SysML, y compris les diagrammes de définition Bloc et diagrammes Bloc internes.



Comme expliqué dans la rubrique d'introduction, les espaces de travail peuvent être définis pour faciliter modélisation d'un type particulier ; dans le cas des diagrammes de définition Bloc , l'un des espaces de travail contenus dans l'ensemble d'espaces de travail de base sera utile pour modélisation , notamment :

- Diagramme de base
- Modélisation du noyau
- Défaut

Un ingénieur qui travaille au niveau d'un projet peut également trouver utiles certains espaces de travail de construction, notamment :

- Feuilles de Route
- Kanban
- Publication de documents
- Révisions et discussions

## Création de définitions Bloc ou Diagrammes Bloc internes

Il existe deux diagrammes que vous créerez généralement lorsque vous travaillerez avec des blocs :

- diagramme de définition Bloc (BDD) - utilisé pour afficher les relations structurelles entre les blocs, y compris les hiérarchies des parties et du type, et les connexions de référence à d'autres blocs
- diagramme Bloc internes (IBD) - utilisé pour montrer comment les propriétés de la pièce sont connectées directement ou via des points d'interaction tels que les ports et les interfaces

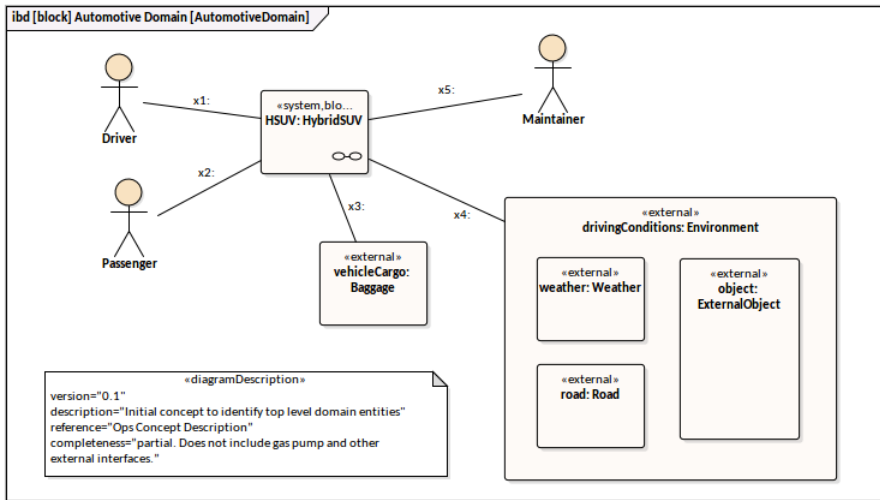
Nous discuterons du diagramme de définition Bloc dans ce chapitre du Guide ; dans le chapitre suivant, nous discuterons du diagramme Bloc interne, qui démontrera comment les blocs peuvent être utilisés dans un contexte donné.

## Diagramme Interne de Bloc - Setting Context

L'un des diagrammes les plus importants à créer au début d'une initiative est un diagramme de contexte, qui décrit le produit ou le service modélisé dans le contexte de son environnement ou de son domaine. Cela permet à l'observateur du modèle d'avoir une image claire et de comprendre comment le produit se situe dans un ou plusieurs des environnements dans lesquels il devra fonctionner. Il donne également une indication précoce de ce qui est dans le champ d'application et de ce qui n'est pas dans le champ d'application du projet. Les éléments du diagramme ont été marqués « Externes » (à l'aide d'un stéréotype) indiquant qu'ils font partie de l'environnement ou du contexte du produit ou du service.

Des images ont été utilisées pour adoucir le diagramme , le rendant plus attrayant pour un large éventail de parties prenantes, y compris les publics professionnels et non techniques.

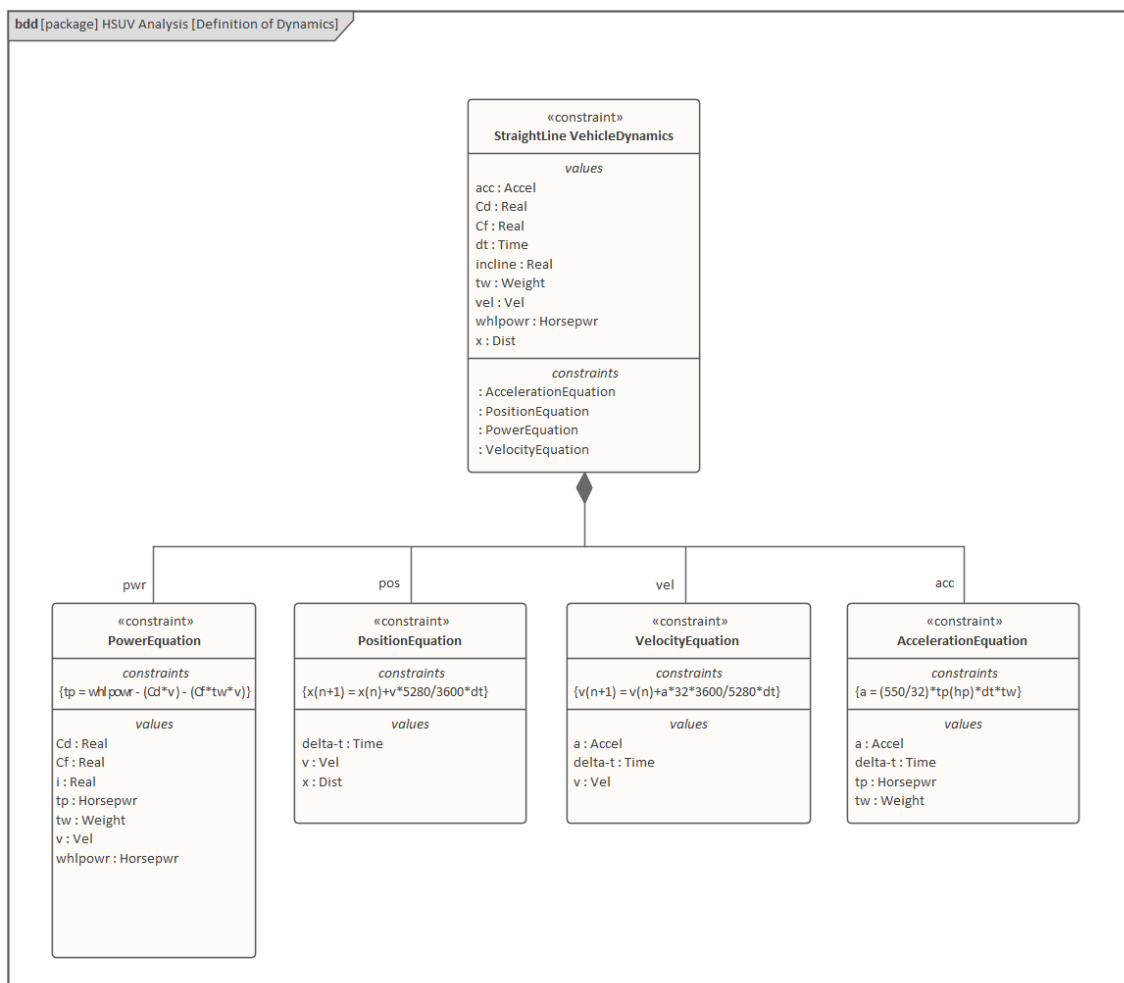
Ce diagramme Bloc internes montre le SUV hybride dans le contexte d'un environnement urbain typique. Il est envisagé de créer au moins un autre diagramme , montrant un environnement tout-terrain où les routes seraient remplacées par des chemins de terre et des rivières, et les bâtiments par des montagnes et des arbres.



# Modélisation des contraintes sous forme de blocs

L'analyse technique exige que l'ingénieur exécute diverses fonctions, notamment la construction de modèles de performance et de fiabilité, l'analyse des compromis, l'analyse des alternatives et les études commerciales. Ces analyses nécessitent souvent l'utilisation d'expressions et d'équations mathématiques qui sont utilisées pour contraindre les éléments d'une analyse. SysML fournit un mécanisme de langage sous la forme d'un ConstraintBlock qui est utilisé pour modéliser les équations graphiquement. Il s'agit d'un mécanisme utile qui permet d'articuler l'expression avec ses paramètres et leurs types. Les équations modélisées peuvent ensuite être réutilisées dans un certain nombre de contextes différents, permettant à un ingénieur de définir la formule de la deuxième loi du mouvement de Newton  $\{F=m*a\}$  ou la définition de Carnot, ce qui donne un théorème fondamental de la thermodynamique  $\{p= W/t =(mg)h/t\}$ . L'un des résultats utiles de modélisation graphique de ces équations est qu'elles peuvent être liées à d'autres éléments du modèle tels que Exigences d'une partie prenante, les objectifs de la mission et les éléments de niveau inférieur tels que les blocs et les artefacts d'implémentation.

Enterprise Architect permet de modéliser ces blocs de contraintes puis de les réutiliser comme Propriétés de contrainte sur diagrammes Paramétriques . Les définitions de contraintes peuvent être regroupées dans des bibliothèques et non seulement utilisées dans l'initiative en cours, mais également réutilisées dans plusieurs projets et initiatives. Dans une rubrique ultérieure, nous verrons comment les contraintes peuvent être intégrées dans un réseau d'équations et utilisées sur diagrammes Paramétriques pour évaluer des alternatives et pour effectuer des analyses de compromis et d'alternatives. La précision et l'excellence technique de l'outil garantissent que les équations définies de cette manière peuvent être créées, maintenues et utilisées avec rigueur.



# Présentation Diagrammes de définition Bloc

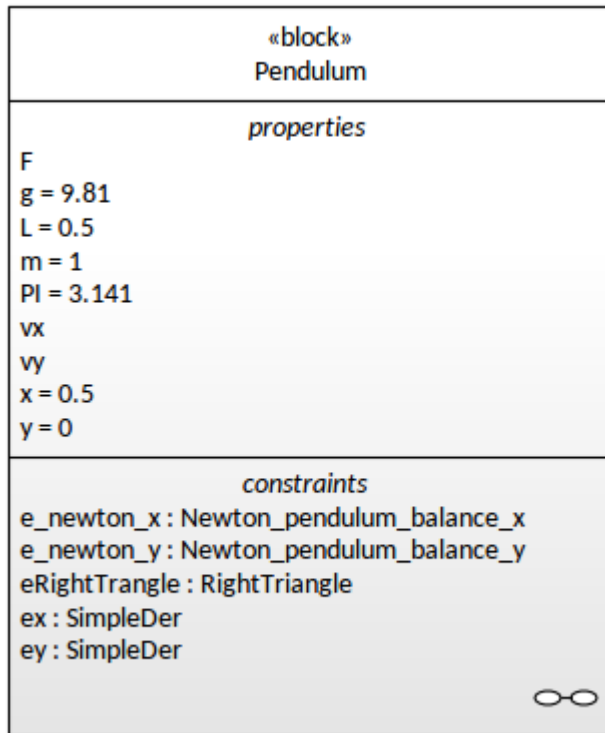
Les blocs fournissent un concept unificateur pour décrire la structure d'un élément ou d'un système, notamment :

- Systèmes
- Matériel
- Logiciel
- Données
- Procédures
- Facilités
- Personnes

Les blocs peuvent avoir plusieurs compartiments standards qui peuvent être rendus visibles sur diagrammes pour décrire les caractéristiques du Bloc , notamment :

- Propriétés (pièces, références, valeurs, ports)
- Opérations
- Contraintes
- Affectations depuis/vers d'autres éléments du modèle (tels que les activités)
- Exigences que le Bloc satisfait
- Compartiments définis par l'utilisateur

N'importe lequel des compartiments peut être supprimé. Aucune ligne de séparation n'est tracée pour un compartiment manquant. Si un compartiment est supprimé, aucune conclusion ne peut être tirée quant à la présence ou l'absence d'éléments qu'il contient. Dans ce diagramme un pendule a été modélisé et un certain nombre de compartiments ont été rendus visibles en vue de créer une simulation paramétrique.



Des compartiments supplémentaires peuvent être fournis sous forme d'extension d'outil pour afficher d'autres propriétés de modèle prédéfinies ou définies par l'utilisateur (par exemple, pour afficher les règles métier, les responsabilités, les variations, les événements gérés, déclenchés, etc.).

Chaque Bloc doit avoir un nom non nul qui est unique dans son espace de noms. La portée d'un nom est le Paquetage qui le contient et les autres Paquetages qui peuvent voir le Paquetage qui le contient.

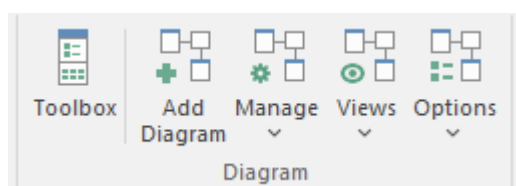
## Création d'un Interrompre lorsqu'une Variable Change de Valeur

Un diagramme de définition Bloc peut être créé à partir de plusieurs emplacements dans l' Interface Utilisateur en utilisant l'une de ces options :

- Ruban de conception - *Ajouter une icône Diagramme* sur le panneau *Diagramme*
- Barre d'outils de la fenêtre Navigateur - *Icône Nouveau Diagramme*
- Fenêtre Navigateur Menu Contexte - *Ajouter Diagramme*

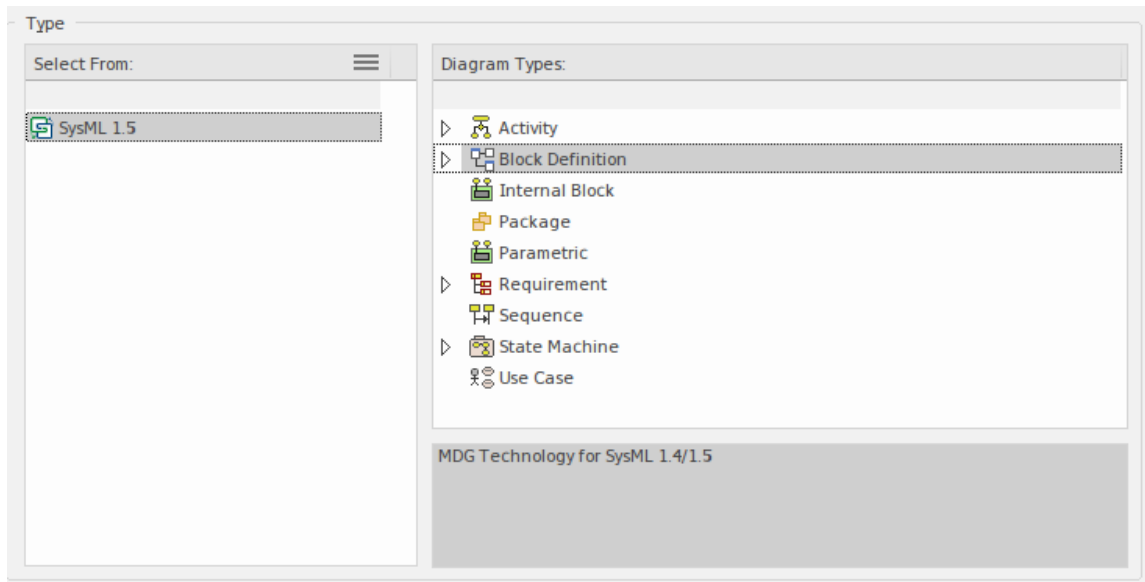
Nous allons utiliser le ruban Conception pour créer un diagramme de définition Bloc . Tout d'abord, sélectionnez l'emplacement dans la fenêtre du Navigateur où vous souhaitez placer le diagramme de définition Bloc . Comme pour tous diagrammes , celui-ci peut se trouver sous un Paquetage ou un élément, mais il est courant d'insérer diagrammes de définition Bloc dans un Paquetage . Une fois l'emplacement sélectionné dans la fenêtre Navigateur , sélectionnez l'option du ruban :

Conception > Diagramme > Ajouter Diagramme



La sélection de cette option ouvrira la dialogue « Nouveau Diagramme », qui vous permettra de nommer le diagramme . Le nom par défaut sera celui du Paquetage ou de l'élément qui contient le diagramme . Avec la perspective SysML choisie et la version de SysML sélectionnée, une liste de diagrammes s'affichera ; sélectionnez le diagramme Définition

Bloc et cliquez sur le bouton OK . Un nouveau diagramme Définition Bloc sera créé à l'emplacement sélectionné dans la fenêtre Navigateur . La Vue Diagramme s'ouvrira, vous permettant de commencer à ajouter des éléments et des connecteurs qui décrivent les blocs et d'autres éléments structurels importants tels que les ports, les interfaces et les types de valeur. Enterprise Architect affichera également les pages « Définition Bloc » de la boîte à outils Diagramme , qui contiennent les éléments et les relations définis par la spécification SysML comme applicables à la construction diagrammes Définition Bloc . N'importe quel nombre d'autres pages de la boîte à outils peut être ouvert si nécessaire, en plus des pages « Éléments communs » et « Relations communes » qui seront toujours disponibles.



Les éléments et connecteurs les plus importants utilisés avec le diagramme de définition Bloc sont :

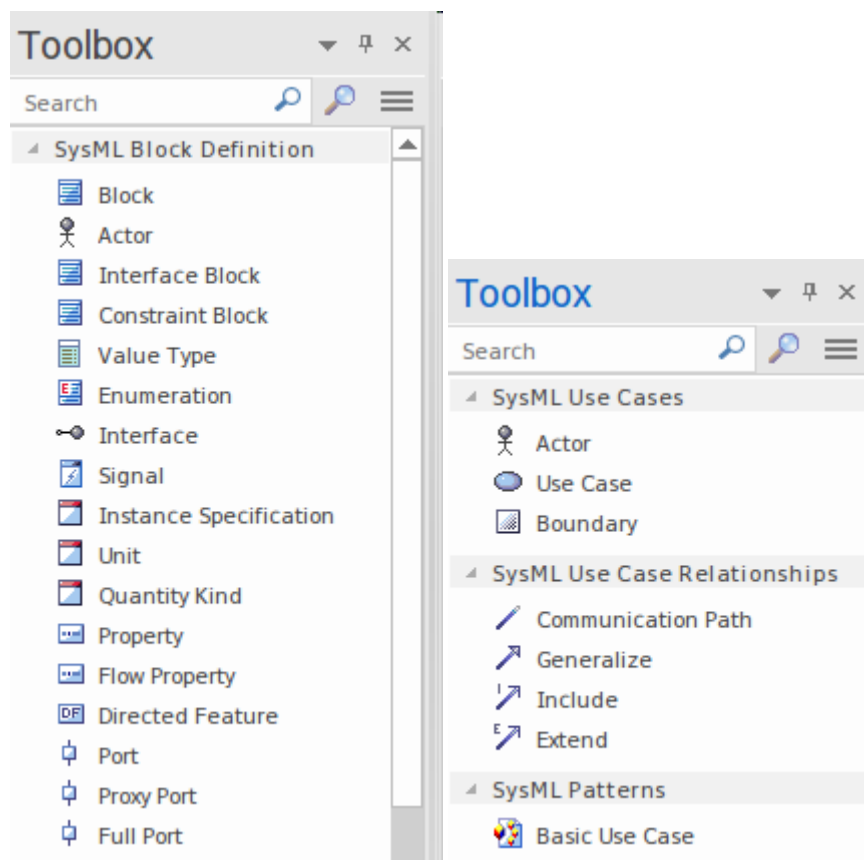
### Éléments

- Bloc
- Bloc de contrainte
- Type de valeur
- Propriété
- Unité
- Quantité Genre
- Port proxy
- Port complet

### Connecteurs

- Association de référence
- Association de pièces
- Association Partagée
- Généralisation
- Dépendance
- Flux Item

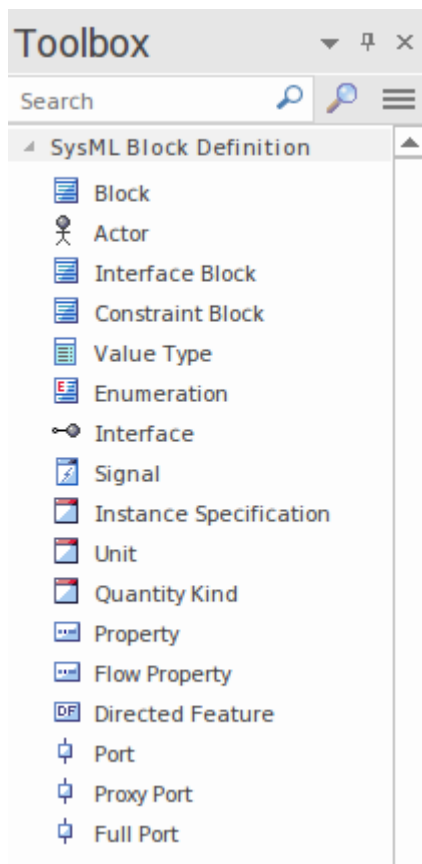




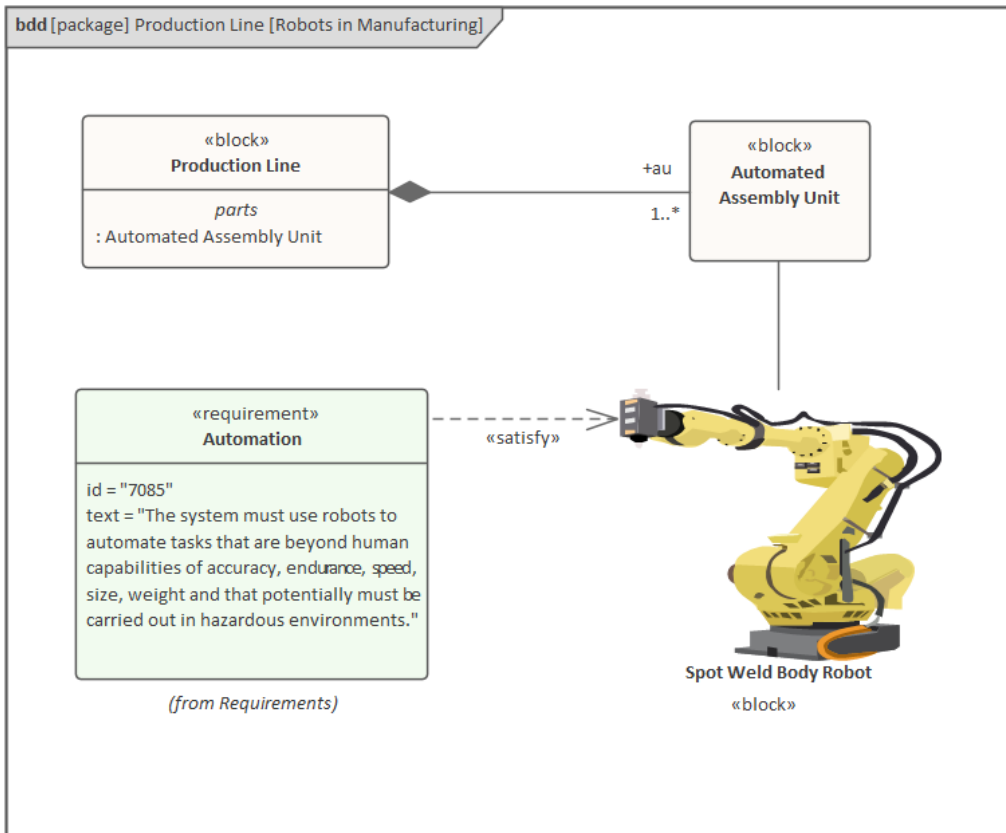
Des éléments peuvent être ajoutés au diagramme en les faisant glisser de la boîte à outils vers le Diagramme Vue . Pour plus d'informations, consultez [Block Definition Diagrams \(BDDs\)](#) Rubrique d'aide.

## Créer un élément Bloc

Les éléments Bloc , comme tout autre type d'élément, peuvent être créés à l'aide de l'option « Ajouter un élément » dans le menu contextuel d'un Paquetage ou en utilisant la page « Définition Bloc SysML » de la boîte à outils Diagramme pour placer un Bloc sur un diagramme de définition Bloc (BDD).



Il est courant que des blocs apparaissent sur plusieurs BDD, où chaque diagramme est conçu pour répondre aux préoccupations d'une partie prenante ou d'un groupe de parties prenantes particulier. Enterprise Architect propose une large gamme d'options d'affichage au niveau des blocs individuels (ou de tout élément) ou au niveau du diagramme . Celles-ci peuvent être utilisées pour décider, par exemple, des compartiments à afficher ou même des fonctionnalités à afficher pour des éléments individuels. Il existe également une large gamme de paramètres d'élément et diagramme génériques pour styliser à la fois l'élément et le diagramme . Par exemple, il est possible de définir les couleurs de l'élément, y compris le remplissage, les bordures et le texte, ou de modifier l'apparence d'un élément en appliquant une image graphique qui traduit mieux la fonction du Bloc . Dans cet exemple, un modélisateur a décidé d'utiliser une image alternative pour un robot de soudage par points afin de transmettre plus clairement l'automatisation en cours sur la ligne de production.

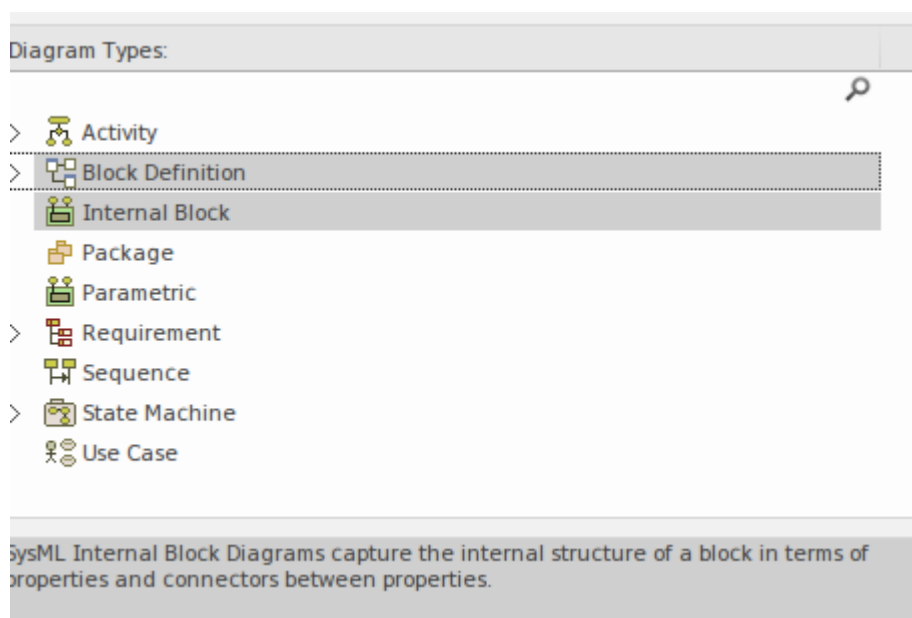


## Les éléments structurels fondamentaux

Les blocs sont les unités modulaires fondamentales et discrètes de description de système. Un Bloc définit un ensemble de fonctionnalités qui sont utilisées pour définir un aspect d'un système ou un système lui-même. Les fonctionnalités sont de deux types fondamentaux : les fonctionnalités structurelles (de quoi un Bloc est constitué) et fonctionnalités comportementales (ce qu'il fait). Les relations d'un Bloc avec d'autres blocs (y compris lui-même) et avec des éléments d'autres types aident à décrire la structure d'un système, d'un sous-système ou d'un composant.

Les modélisateurs de systèmes utilisent Diagrammes de définition Bloc (BDD) pour définir la structure des blocs et Diagrammes Bloc internes (IBD) pour décrire leur utilisation.

Ces diagrammes peuvent être créés à partir de la dialogue ' Nouveau Diagramme ', accessible depuis la barre d'outils de la fenêtre Navigateur .



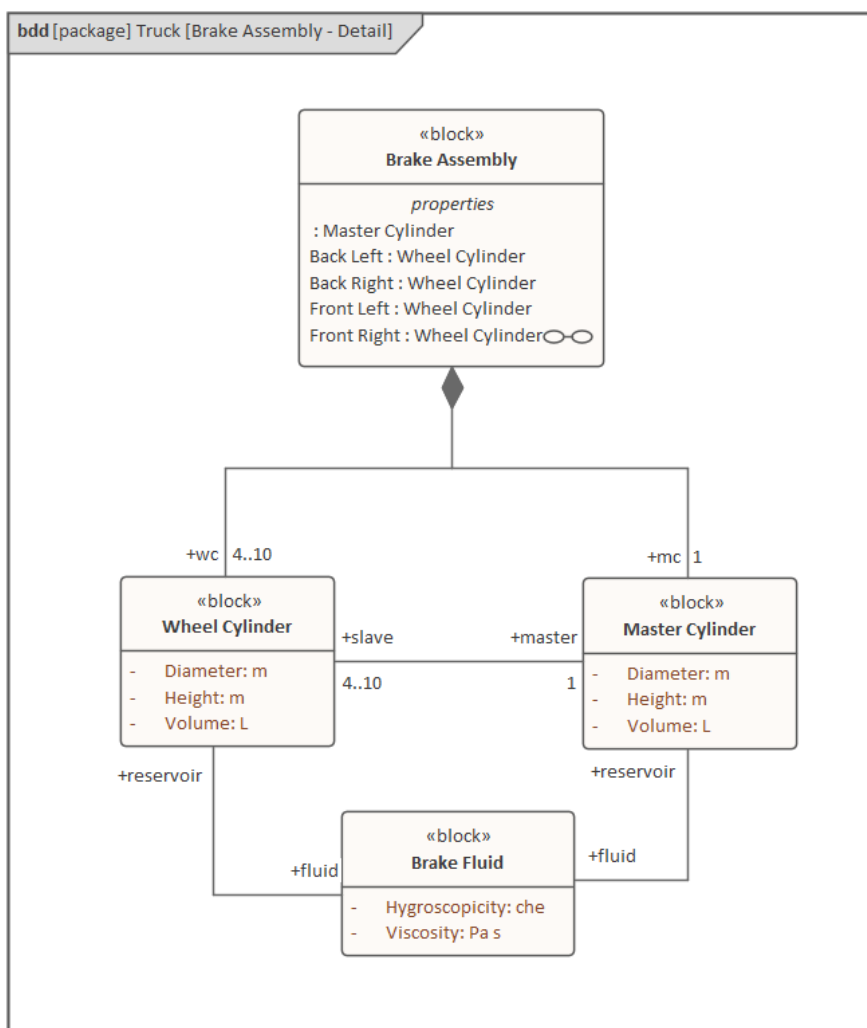
## Modélisation Fonctionnalités Structurelles

Les blocs sont généralement définis à l'aide d'une série de fonctionnalités structurelles. Il s'agit des propriétés du Bloc et elles définissent la nature du Bloc . Par exemple, une locomotive (matériel roulant) aura des propriétés telles que la classe de locomotive, le numéro d'identité, le nombre d'ensembles de roues, la force motrice, les moteurs et une série d'autres propriétés. Il est important de se rappeler que le Bloc est un classificateur qui décrit un ensemble de locomotives. La locomotive à l'avant du train dans lequel vous embarquez pour vos vacances d'été est une instance d'une locomotive et elle aura une classe particulière, par exemple la classe OSE 660, et un identifiant SM-09873, et 8 ensembles de roues.

Enterprise Architect supporte trois types de fonctionnalité structurelles de base, chacune étant importante pour modélisation de différents aspects de la structure d'un Bloc . Nous examinerons chacune d'elles dans ces sections.

- Pièces - *un bloc est composé de pièces*
- Références - *se référer aux fonctionnalités d'autres blocs*
- Valeurs - *décrire les quantités*

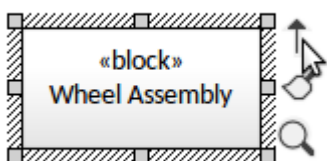
Ce diagramme montre les trois types de fonctionnalité structurelles.



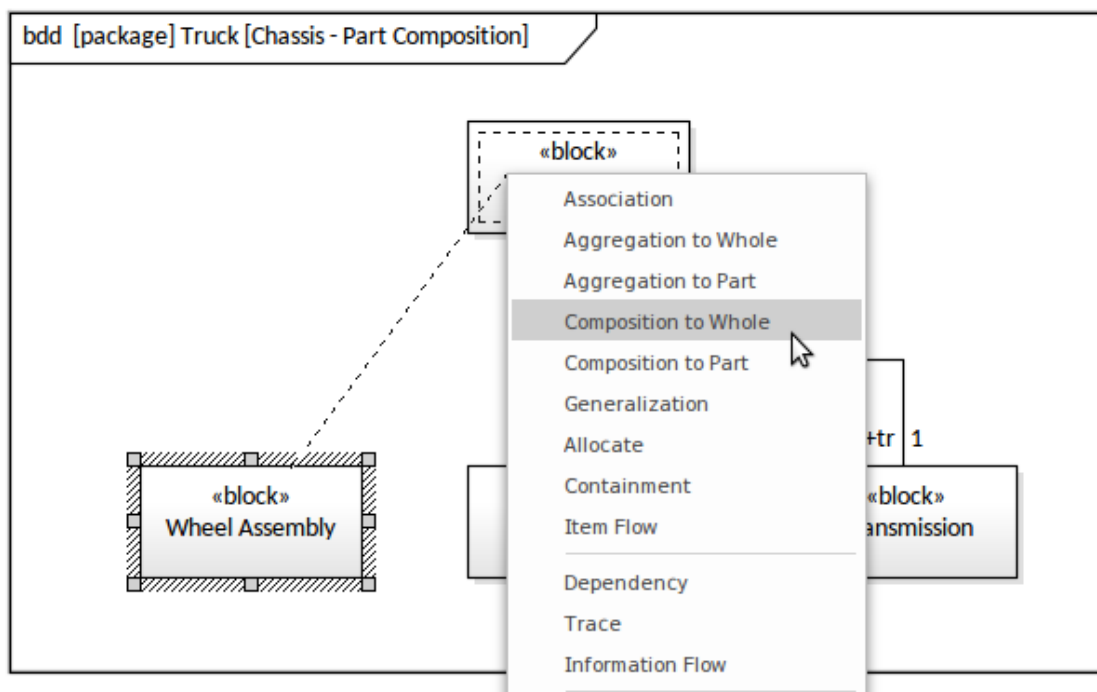
Le système de freinage est constitué d'un certain nombre de pièces, dont deux sont représentées sur ce diagramme . L'association de pièces a été utilisée pour indiquer que le *maître-cylindre* et les *cylindres de roue* sont des constituants fondamentaux du système de freinage. Une association de référence a été utilisée pour montrer à la fois une relation entre les deux types de cylindres et également entre les cylindres et le *liquide de frein*. Les valeurs qui ont été saisies comme attributs sont affichées avec leurs types de valeur associés ; par exemple, le volume a un Type de valeur de L, qui est le symbole de la dimension du volume dont l'unité SI est le litre.

## Blocs composés de pièces

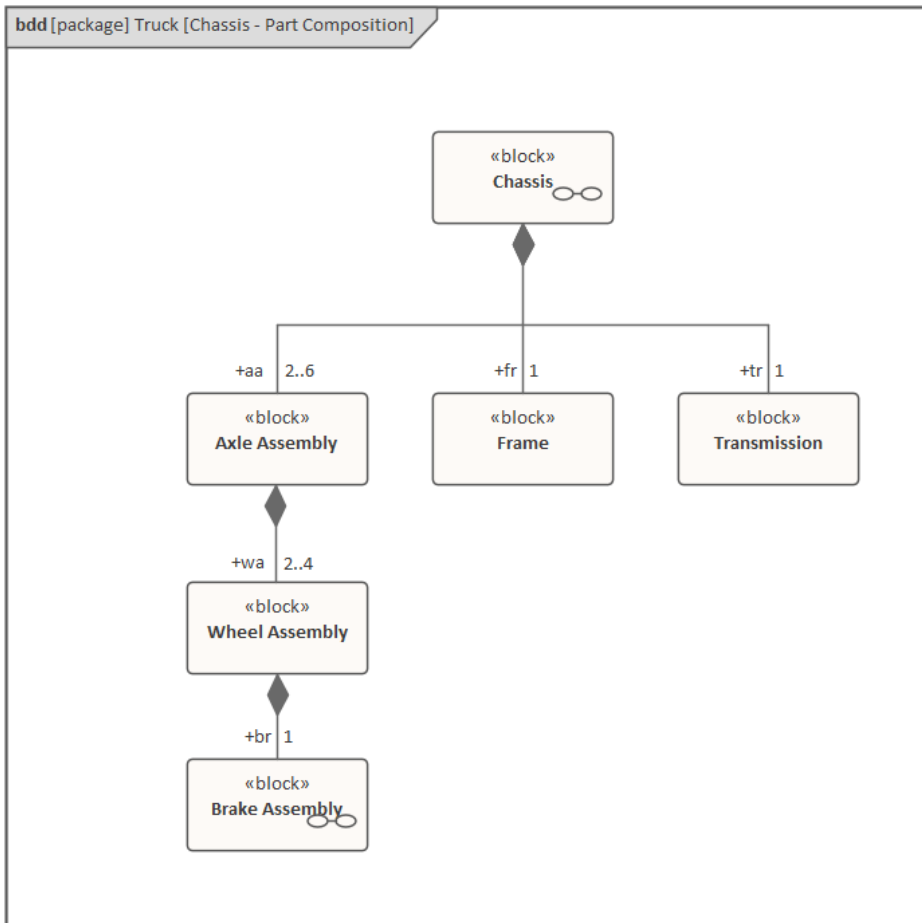
Une partie est une Fonctionnalité structurelle d'un Bloc et forme l'une des relations les plus fortes entre un Bloc et ses propriétés. Il est important de comprendre qu'une instance d'un Bloc peut avoir plusieurs instances d'une partie ; par exemple, un camion peut avoir plusieurs assemblages de roues et, selon la taille et le type de camion, ce nombre peut être aussi bas que 2 ou aussi élevé que 10. Ces configurations possibles peuvent être spécifiées dans la définition du Bloc et de ses parties, qui sont officiellement connues sous le nom de multiplicités - le nombre le plus bas est appelé limite inférieure et le nombre le plus élevé limite supérieure. Une partie sera généralement typée par un autre Bloc, ainsi dans l'exemple le type de la partie sera un autre Bloc nommé « assemblage de roues », qui comprendrait lui-même généralement un essieu et deux assemblages de roues. Ainsi, chaque partie sera définie dans le Bloc avec un nom, un type et une multiplicité. L'outil permet de créer la relation de *composition de pièce* de plusieurs manières, mais la manière la plus immédiate est peut-être de faire glisser à la fois le Bloc de *châssis* (l'ensemble) et le Bloc d'*assemblage de roues* (la pièce) sur le diagramme, puis d'utiliser le Quick Linker pour faire glisser de la pièce (*assemblage de roues*) vers l'ensemble (*châssis*).



Le déplacement de l' object source vers la cible affichera un menu de connecteurs possibles et l'ingénieur choisira le connecteur *Composition vers tout*. Le résultat sera une relation avec le marqueur en losange à l'extrémité *Châssis* de la ligne, indiquant qu'il s'agit de l'ensemble et que l'élément à l'extrémité *Ensemble de roues* est la Pièce.

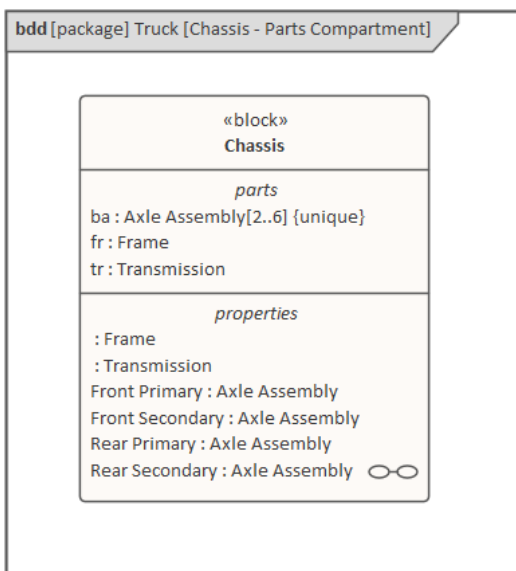


Les propriétés du connecteur vous permettront de définir le rôle source et les multiplicités qui, comme indiqué, spécifient le nom et le nombre possible de pièces pour chaque instance d'un *châssis*.



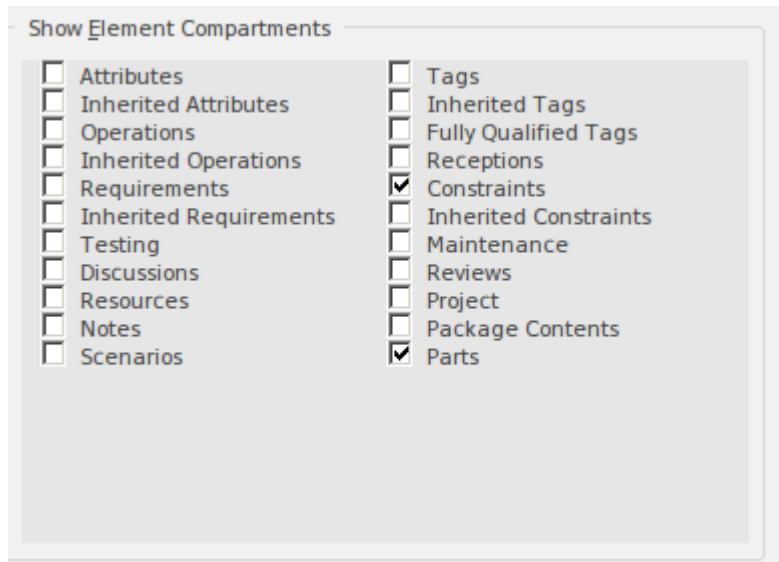
Dans le diagramme le modélisateur a expressément défini les pièces en utilisant l'association de pièces, disponible dans la boîte à outils de définition Bloc SysML.

Dans ce diagramme le modélisateur a utilisé le compartiment Pièces du Bloc propriétaire pour afficher les pièces appartenant au Bloc châssis.

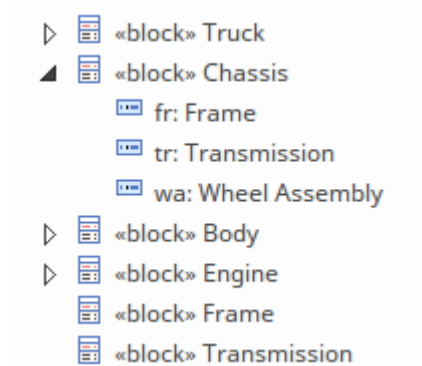


Le compartiment Pièces s'affiche par défaut, mais sa visibilité peut être contrôlée au niveau du diagramme à l'aide des Propriétés diagramme ou au niveau d'un élément individuel à l'aide de l'option « Visibilité du compartiment » de l'élément dans le menu contextuel de l'élément. La définition de la visibilité au niveau du diagramme aura pour conséquence que tous les éléments du diagramme seront conformes à la visibilité spécifiée (affichés ou non comme

spécifié), tandis que la définition de la visibilité au niveau de l'élément n'affectera que l'élément sélectionné.



Les éléments du référentiel seront mis à jour, qu'ils soient modifiés dans le diagramme, dans le Navigateur ou dans toute autre fenêtre. Dans l'exemple, l'ingénieur a créé les pièces dans le diagramme en faisant glisser une association de pièces depuis la boîte à outils. En réponse à cela, Enterprise Architect crée trois nouvelles pièces, qui sont placées sous le nœud Châssis dans le Navigateur comme indiqué dans cette capture d'écran.



L'association de parties est le type de relation d'association le plus fort - le continuum de force du plus faible au plus fort étant :

1. Association de référence
2. Association Partagée
3. Association de pièces

Nous explorerons les autres relations dans les sections ultérieures de ce guide.

## Références à d'autres blocs

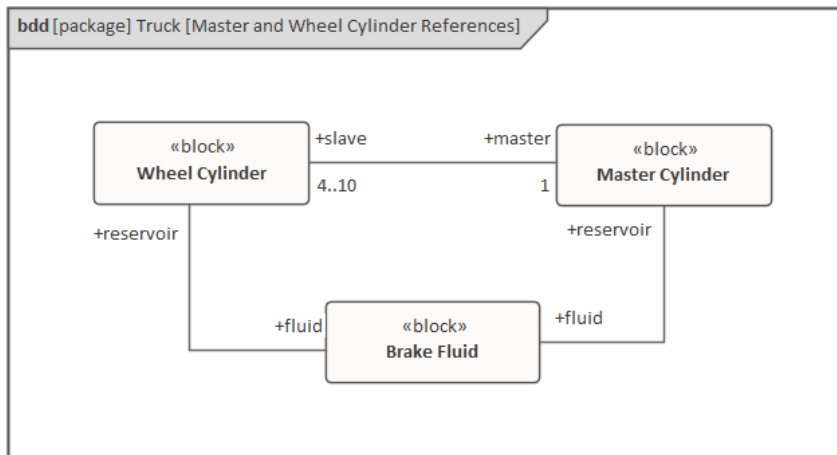
Comme indiqué précédemment, l'association de parties est le type de relation le plus fort dans le SysML et implique un sens des responsabilités de la part de l'ensemble :

- Il est responsable de la durée de vie des pièces qui le composent
- Une partie ne peut participer à une composition de partie qu'avec un seul bloc

La deuxième condition signifie que la multiplicité à la fin d'une composition de partie est toujours 1..1, ce qui peut être abrégé en 1.

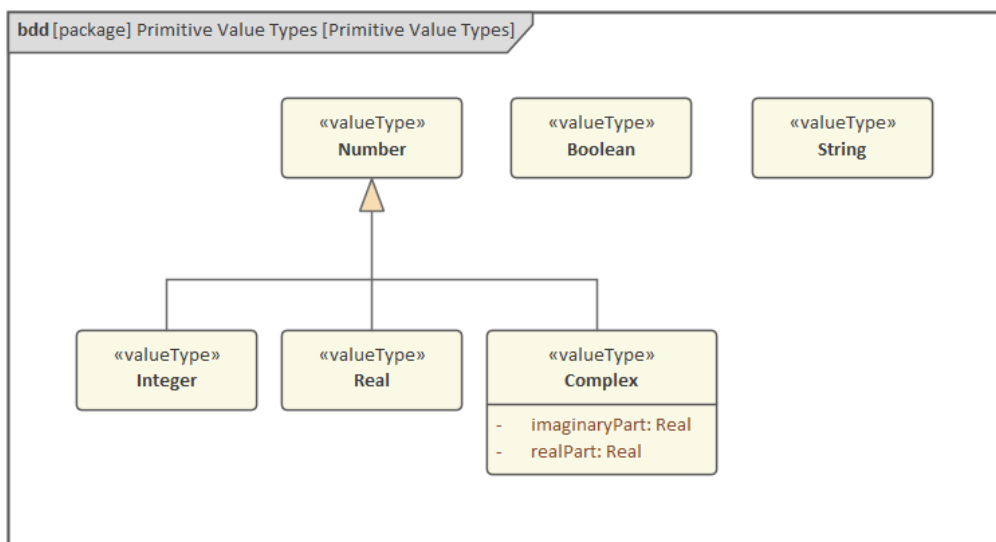


Il existe cependant une autre relation, l'association de référence (ou référence en abrégé) qui peut être utilisée pour spécifier les relations entre les blocs indépendamment de la composition ou de la notion d'un bloc faisant partie d'un autre. Cela fournit un mécanisme très utile pour créer des relations entre des blocs qui font partie de différentes hiérarchies de pièces ou entre deux blocs quelconques qui sont liés l'un à l'autre. Par exemple, le maître-cylindre et les cylindres de roue ont tous deux une relation avec le liquide de frein qui est utilisé pour remplir leurs réservoirs. Le cylindre de roue pourrait à son tour avoir une relation avec un mécanicien qui vérifie périodiquement le cylindre pour détecter les fuites qui compromettraient l'efficacité du système de freinage.



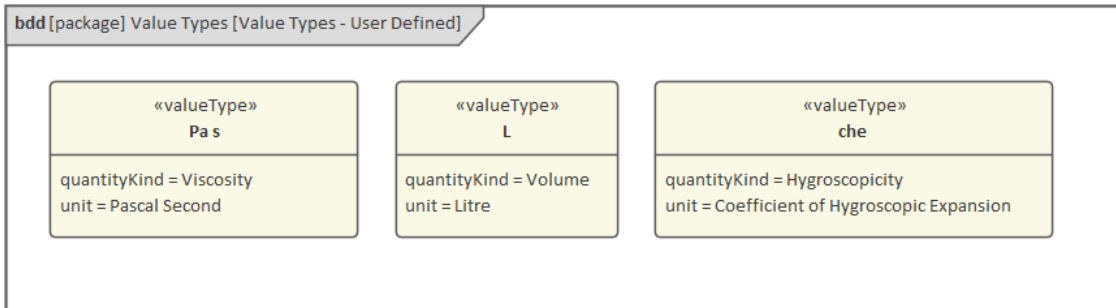
### Valeurs utilisées pour décrire les quantités

Les blocs peuvent avoir des propriétés avec un certain type de valeur quantifiable ; par exemple, un moteur a une puissance de sortie, un réservoir a un volume, une automobile a une couleur, un wagon de chemin de fer a un nombre de bogies. Les types peuvent être un type primitif défini comme Number, Integer, Real, Complex, Boolean ou String, comme illustré dans ce diagramme.

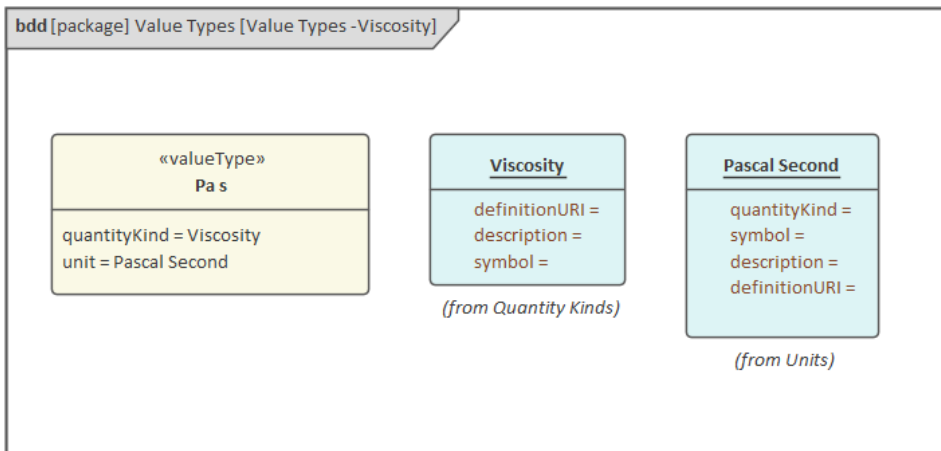


Un ingénieur, une équipe ou une communauté de pratique peut également définir un nombre quelconque de types de valeur, simples ou structurés. Ceux-ci peuvent être basés sur un nombre quelconque de systèmes d'unités, tels que le Système international d'unités (SI). Un ingénieur automobile qui conçoit un système de freinage peut se retrouver à utiliser un certain nombre de types de valeur SI standard et un certain nombre de types dérivés, ainsi que d'autres valeurs

non définies dans le cadre de cette norme. Ce diagramme illustre comment ces valeurs peuvent être définies, à l'aide de l'élément Type de valeur disponible dans la boîte à outils de définition Bloc SysML.



Le Type de valeur possède deux propriétés définies : la quantité (quantité) et l'unité (unité). Celles-ci peuvent également être modélisées dans Enterprise Architect et confèrent de la rigueur à l'application du Type de valeur. Un ingénieur saura que le type est basé sur une quantité (dimension) et une unité définie. Ce diagramme montre ces éléments pour le Type de valeur (viscosité).



## Modélisation Comportementale Fonctionnalités

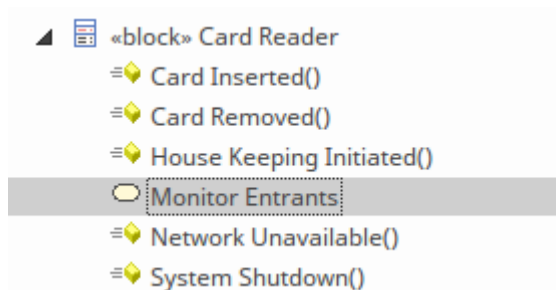
Lorsqu'un système est en fonctionnement, des instances des blocs qui ont été définies dans le cadre de l'architecture et de la conception détaillée sont instanciées. À ce stade, si un Bloc a un comportement de classificateur défini, ce comportement commencera généralement et continuera à fonctionner jusqu'à ce que le Bloc soit détruit. Ainsi, dans l'exemple de notre système de parking, lorsque le système a été activé, le lecteur de cartes commencera à fonctionner et son comportement principal entrera en vigueur. En plus de cela, un Bloc (même s'il est fondamentalement de nature structurelle) possède fonctionnalités comportementales qui seront appelées à effectuer des travaux. En résumé, il existe deux définitions fondamentales du comportement qui sont définies dans le contexte d'un Bloc , à savoir :

- Comportement du classificateur - *le comportement natif qui est initié lorsqu'un Bloc est instancié*
- Fonctionnalités de comportement - *ce sont les opérations et les réceptions (et leurs signaux associés)*

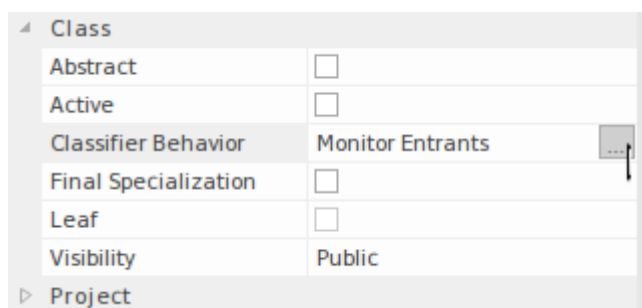
Nous examinerons ces différents comportements dans les prochaines sections du Guide, mais il est important de comprendre qu'ils fonctionneront à l'unisson, coordonnés par des interactions système qui garantiront que les opérations sont appelées en séquence et que les signaux sont reçus et traités par les réceptions.

### Comportement du classificateur de blocs A

Un Bloc a le potentiel d'effectuer un travail, mais en soi, il s'agit d'une entité quelque peu latente et doit être commandé pour agir par un type d'appel à ses opérations ou par la réception d'un signal, d'un changement d'état ou d'un autre déclencheur comportemental. Un Bloc a un concept de son comportement natif ou de classificateur, comme on l'appelle formellement. Ce diagramme montre un Bloc dans la fenêtre Navigateur qui a une Activité imbriquée qui sera définie comme le Comportement de Classificateur pour le Bloc .



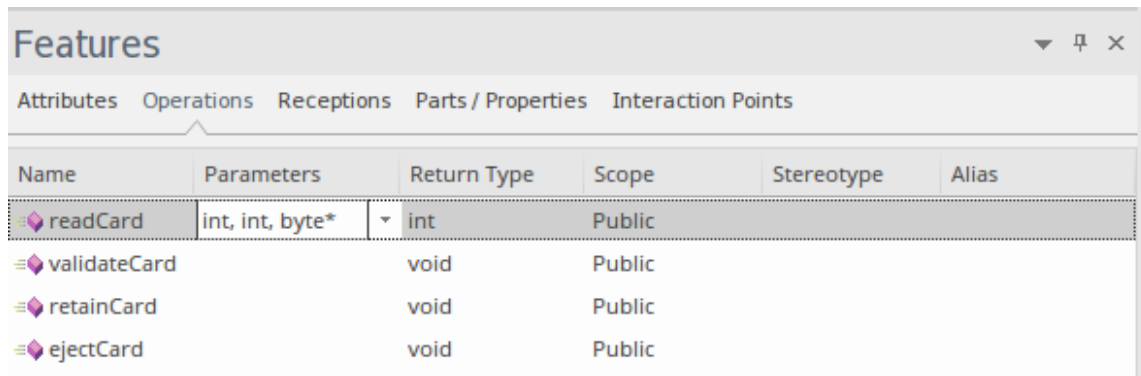
Pour sélectionner ce comportement pour le comportement du classificateur, ouvrez la fenêtre Propriétés et modifiez la propriété Comportement du classificateur en sélectionnant l'icône [...] et en localisant le comportement (activité) approprié comme indiqué dans cette illustration.



### Les opérations comme Fonctionnalités Comportementale

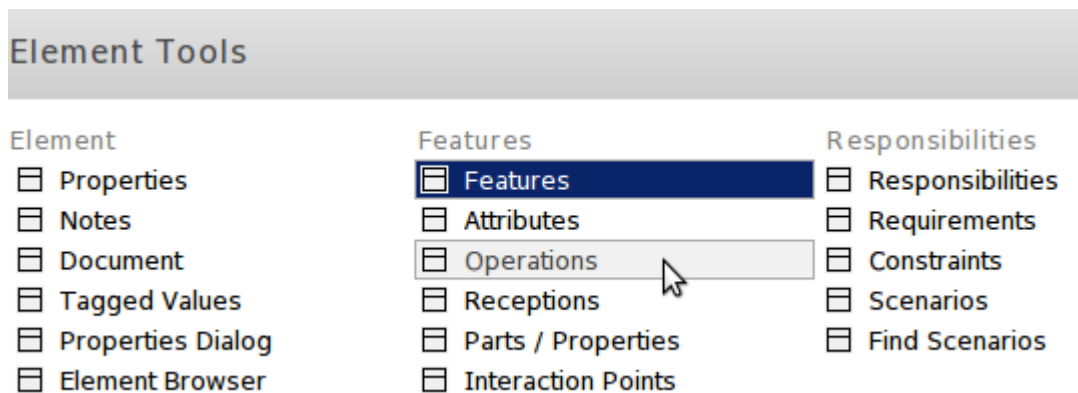
Les blocs peuvent définir des opérations essentiellement comme les « muscles » du Bloc , car ce sont les opérations qui

effectuent la majeure partie du travail requis par le système. Dans Enterprise Architect un ingénieur peut accéder aux opérations à partir de plusieurs points de l'interface utilisateur, mais tous ces points ouvriront la fenêtre Fonctionnalités , qui répertorie les opérations sur l'onglet « Opérations », comme indiqué ici :

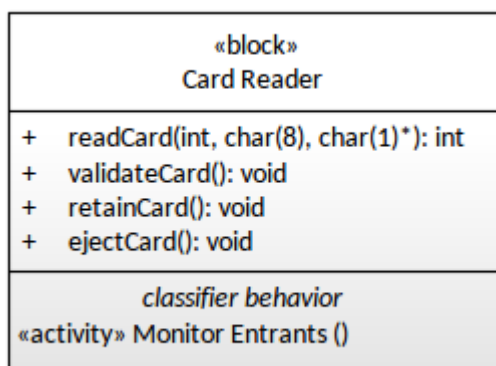


La fenêtre Fonctionnalités est utile comme résumé de toutes les fonctionnalités structurelles et comportementales, y compris les pièces et les points d'interaction appartenant au Bloc . Le moyen le plus simple de créer une opération est de sélectionner le Bloc dans un diagramme ou dans la fenêtre Navigateur et de cliquer sur l'élément du ruban :

Conception > Élément > Editeur > Fonctionnalités > Opérations

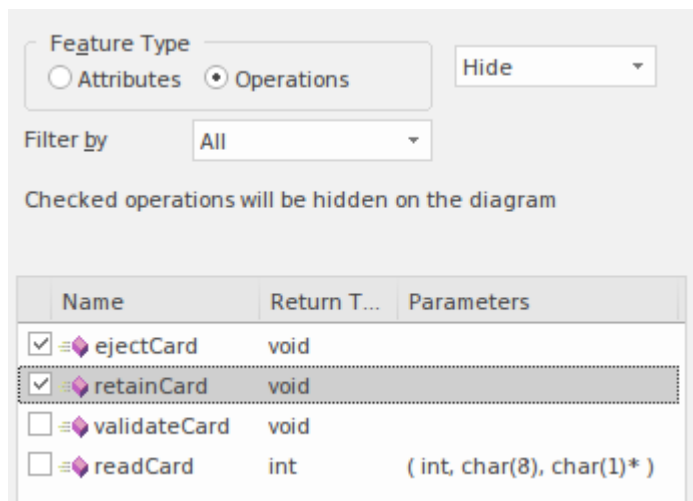


Les opérations sont simplement créées en sélectionnant l'onglet « Opérations » et en ajoutant le nom et d'autres détails dans une ligne de la fenêtre. N'importe quel nombre d'opérations peut être créé, et chaque opération peut définir n'importe quel nombre de paramètres, qui spécifient les entrées et les sorties de l'opération. Leur importance sera abordée plus loin dans cette section lorsque nous décrirons la relation entre les paramètres d'activité et Pins Action . Les opérations peuvent également être affichées dans un diagramme , soit seules, soit avec d'autres fonctionnalités , dont chaque type est affiché dans un compartiment séparé de l'élément parent.

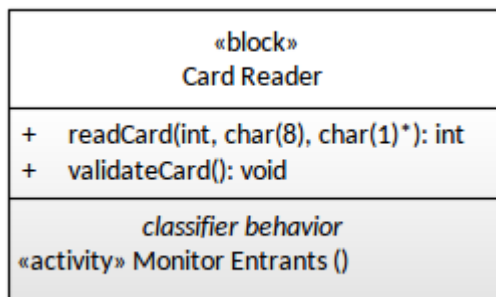


Il existe une large gamme d'options qui régissent la manière dont les opérations sont affichées, y compris la possibilité de

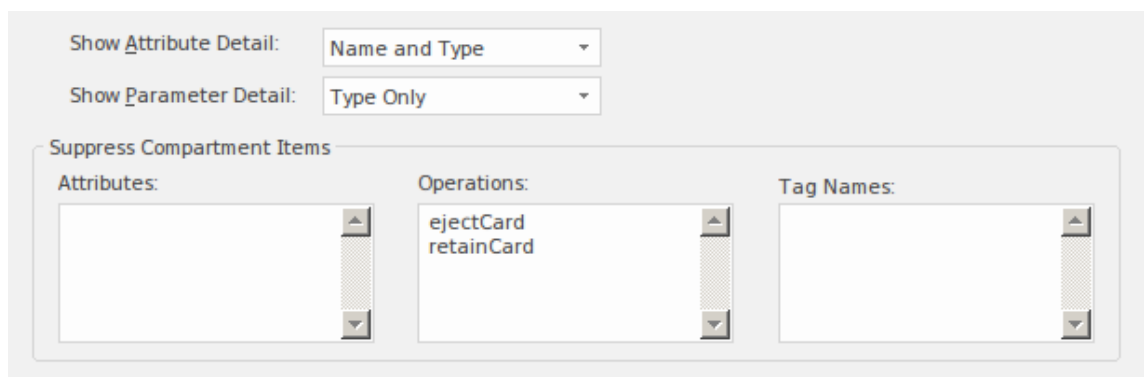
ne pas afficher l'intégralité du compartiment ou d'afficher uniquement des opérations particulières en supprimant les autres de l'affichage.



Cela aura pour conséquence de masquer les opérations sélectionnées sur le diagramme , ce qui est un outil de présentation très utile car il aide un ingénieur à créer un diagramme axé sur un aspect particulier du Bloc , supprimant ou masquant le contenu non pertinent et distrayant. Ce fragment diagramme montre le résultat de la suppression des opérations :



La même chose peut être faite pour les attributs au niveau d'un élément, et une fonction similaire est disponible pour supprimer des opérations, des attributs et Valeur Étiquetés particuliers au niveau d'un diagramme . Un ingénieur peut utiliser la fonction au niveau du diagramme lorsqu'une opération particulière apparaît sur plusieurs blocs et qu'il souhaite la supprimer pour chaque élément du diagramme .



Les opérations peuvent être invoquées selon deux modes, de manière synchrone ou asynchrone, et peuvent être initiées de différentes manières en fonction du type de comportement qui orchestre le comportement du système, notamment :

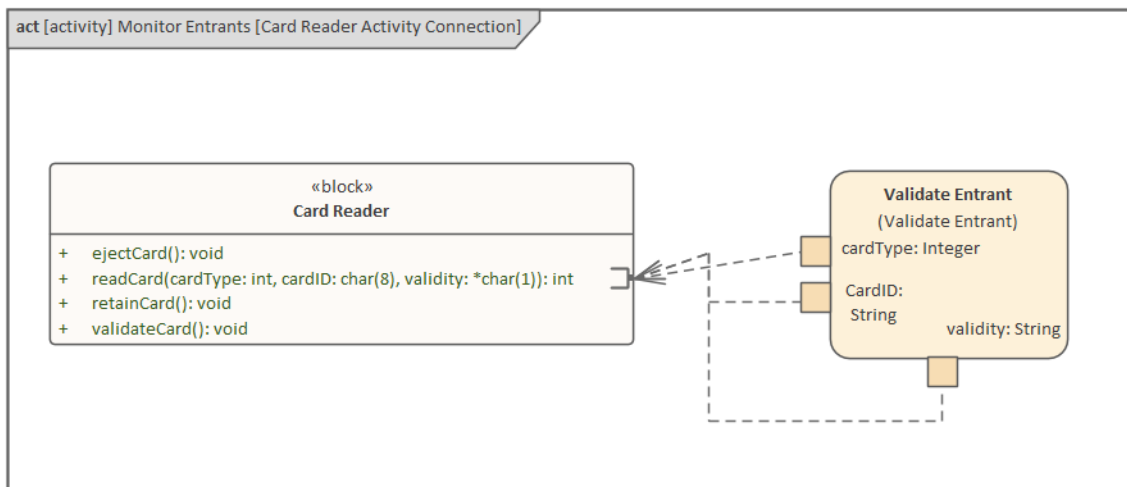
- Une Action d'appel (invocation d'une activité)

- Un message dans le cadre d'une interaction ( diagramme Séquence )
- Une State machine

Cela signifie que l'opération peut être visualisée dans une gamme de diagrammes SysML et apparaîtra différemment dans différents contextes. Par exemple, dans un diagramme Séquence où des messages sont envoyés entre des instances de blocs ou d'autres classificateurs, l'opération apparaîtra comme une annotation à l'un des messages entrants du Bloc pour montrer que l'opération sera initialisée à la suite du message. Enterprise Architect permet à un ingénieur d'accéder à la liste des opérations du Bloc directement à partir de ce diagramme et permettra également de créer des opérations directement à partir du diagramme .

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

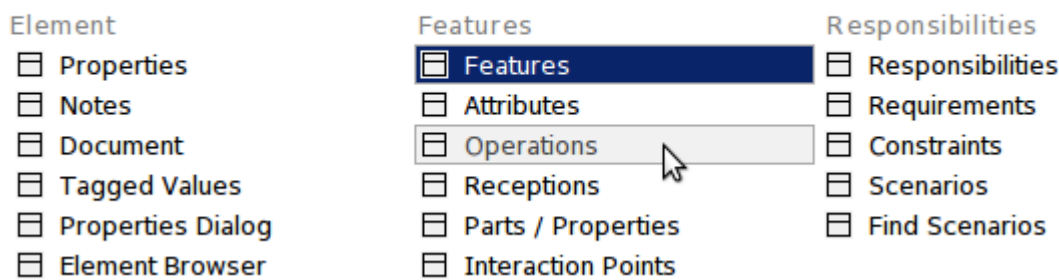
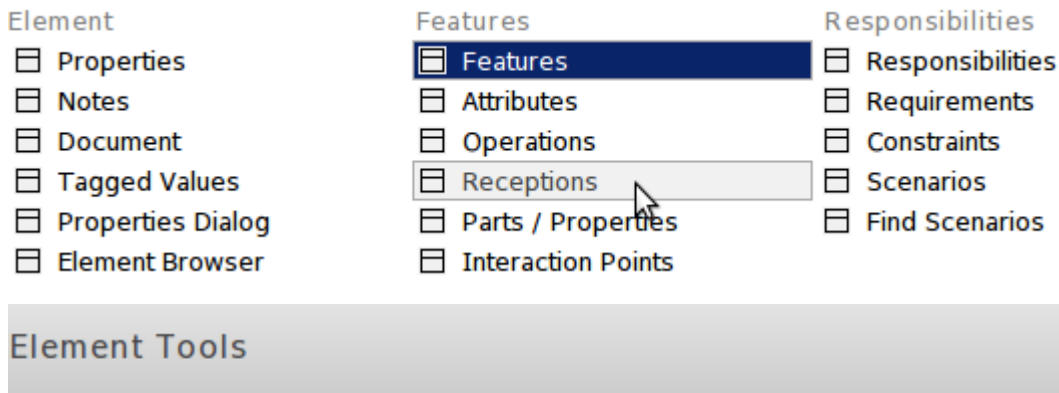
Dans le cas de l' Action Appel d'opération, les Pins de l'élément doivent être alignés par type et par nom sur les paramètres de l'opération appelée ; Enterprise Architect vous aide à visualiser ce mapping sur un diagramme , en utilisant la facilité 'Lien vers Fonctionnalité '.



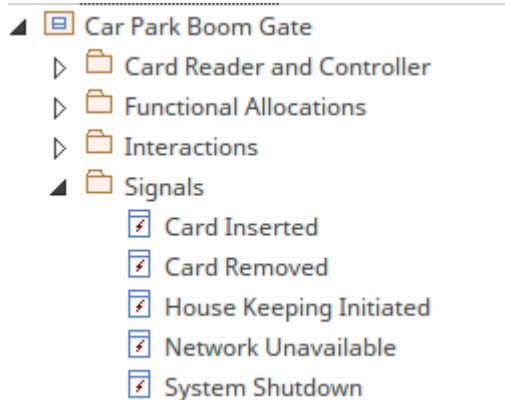
## Les réceptions comme Fonctionnalités Comportementale

Les réceptions sont une autre fonctionnalité comportementale d'un Bloc mais, contrairement à une opération, les réceptions ne peuvent être appelées que de manière asynchrone. Les réceptions fonctionnent également différemment des opérations dans la mesure où un appel d'opération identifie spécifiquement une opération à invoquer, tandis que la réception d'une instance d'un signal est considérée comme une demande de toute réception de l' object récepteur qui référence ce signal ou toute généralisation directe ou indirecte de celui-ci. De cette façon, il existe un niveau d'indirection entre l'élément appelant et la réception. Une réception possède des paramètres correspondant aux attributs du signal référencé par la réception, et ceux-ci sont considérés comme des paramètres « entrants » de la réception.

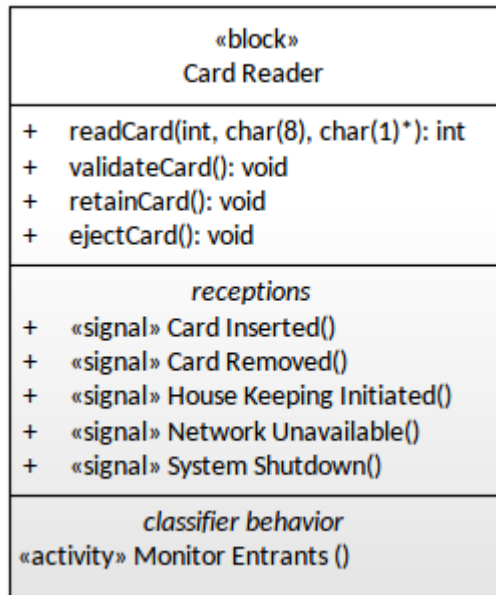
La manière la plus simple de créer une réception est de cliquer sur le Bloc dans un diagramme ou dans la fenêtre Navigateur et de sélectionner l'élément de ruban « Conception > Élément > Éditeur > Réceptions ».



Pour créer une nouvelle réception, vous devez d'abord avoir créé le signal approprié auquel associer la réception. Lorsque vous créez la réception, vous serez invité à localiser le signal approprié dans la fenêtre Navigateur comme indiqué ici :



Les réceptions, comme les opérations, peuvent être affichées dans un compartiment spécialisé dans un Bloc sur un diagramme . Il est possible de personnaliser l'affichage et de supprimer toutes les réceptions ou de configurer les réceptions particulières à afficher. Dans cette capture d'écran, l'ingénieur a décidé de rendre toutes les réceptions visibles, mais chaque diagramme et chaque Bloc au sein d'un diagramme peuvent être configurés différemment.



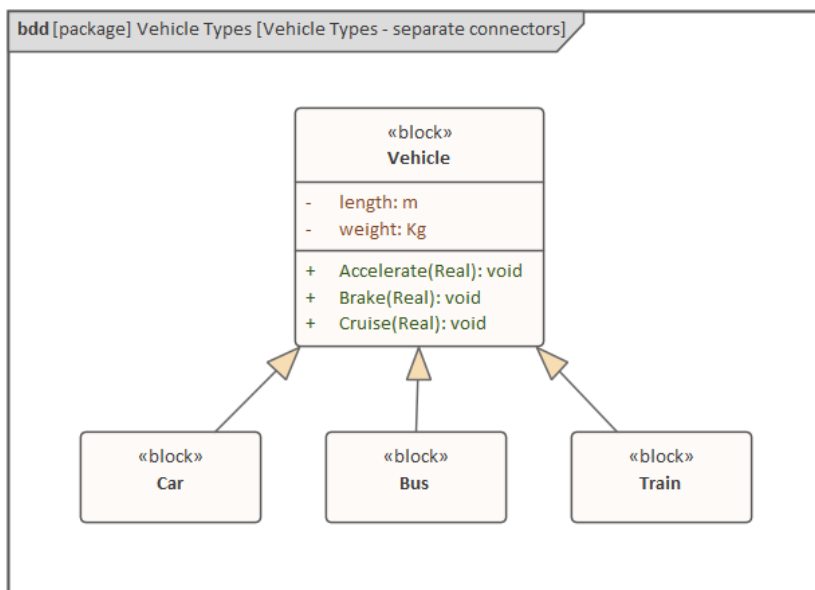


## Autres Relations Bloc

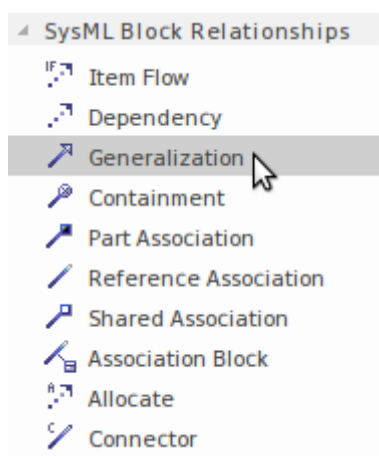
Les blocs sont les éléments structurels clés du SysML et peuvent participer à une variété de relations, dont certaines ont été abordées dans les sections précédentes du Guide lorsque nous avons abordé les associations. Il existe un certain nombre d'autres relations qui peuvent être utilisées lors de la définition des blocs.

### Généralisation d'une relation de famille

Dans une section précédente, nous avons parlé de l'association de parties comme étant le type de relation d'association le plus fort, mais il existe une autre relation - la généralisation - qui est également très forte et qui est essentiellement utilisée pour modéliser le fait que les blocs (et d'autres classificateurs) appartiennent à la même famille. Le mot « classificateur » vient de nos langues naturelles, comme le chinois et le thaï, qui ont une manière abstraite de classer ou de regrouper des classes de noms qui ont des caractéristiques similaires ; par exemple, une ceinture et une route sont des choses longues et fines, tandis qu'une baie et une balle sont des choses round . De même, avec le SysML, la relation de généralisation est utilisée pour classer les choses et la structure peut avoir une profondeur arbitraire. À bien des égards, il est plus naturel pour les ingénieurs de lire la relation à l'envers et de dire que quelque chose est une version spécialisée de quelque chose.

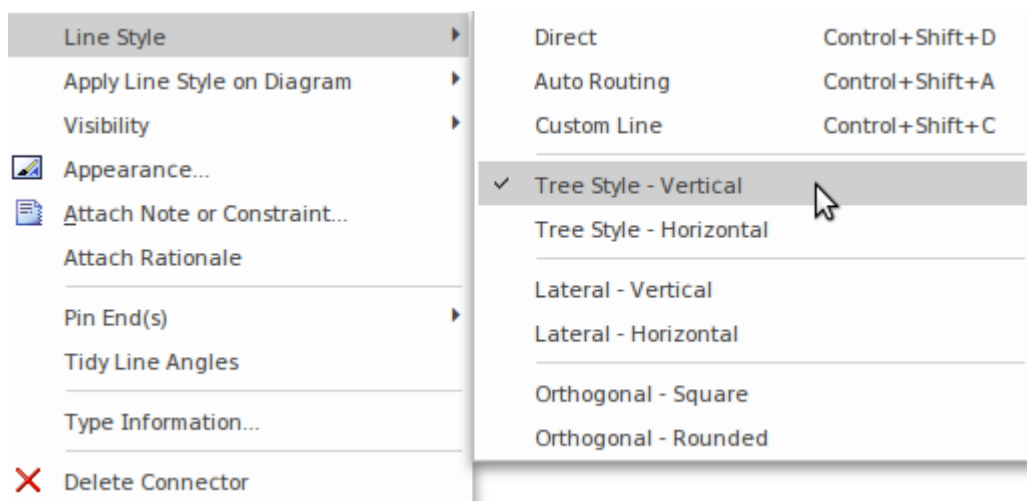


Enterprise Architect permet à un ingénieur de créer ces hiérarchies de classification pour les blocs, les types de valeur, les signaux, les interfaces, les activités, etc. Un diagramme contient généralement une seule famille.

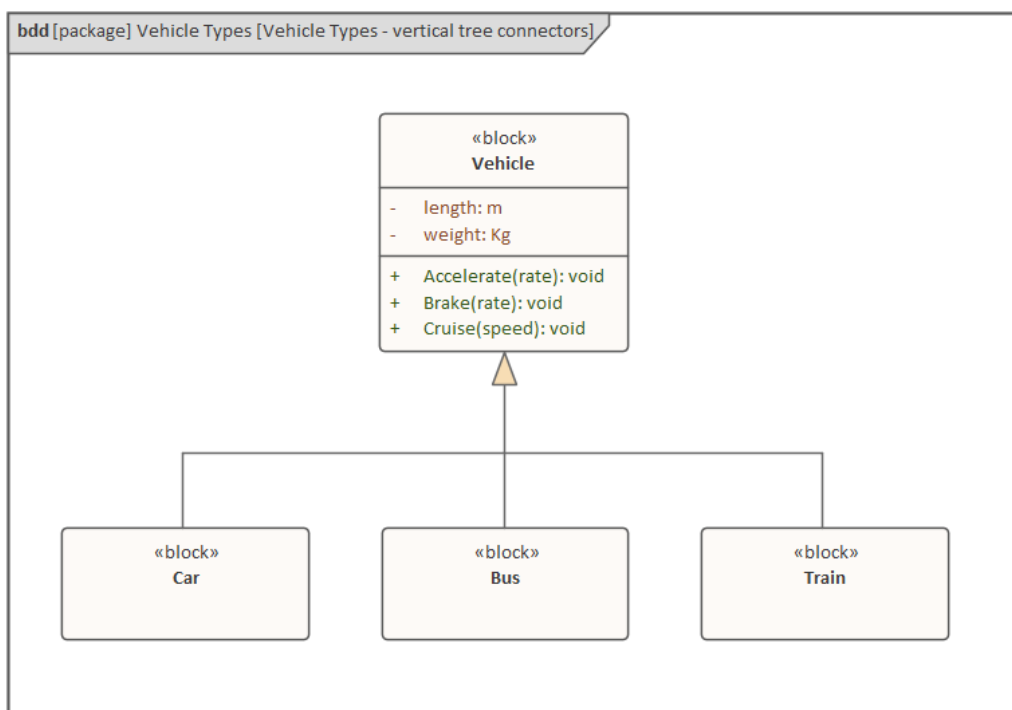


La relation peut être établie en sélectionnant d'abord l'icône « Généralisation » dans la boîte à outils, puis en effectuant

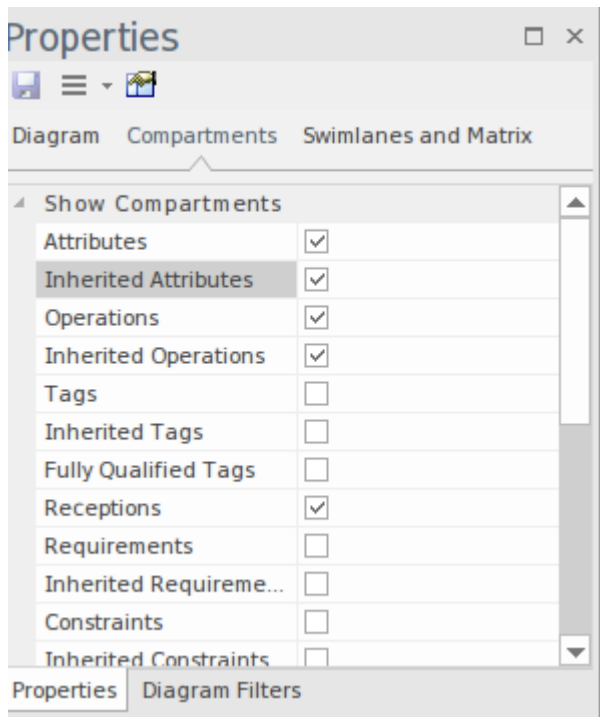
un glisser-déposer de l'élément le plus spécialisé vers l'élément le plus généralisé. Vous pouvez également le faire à l'aide du Quick Linker .



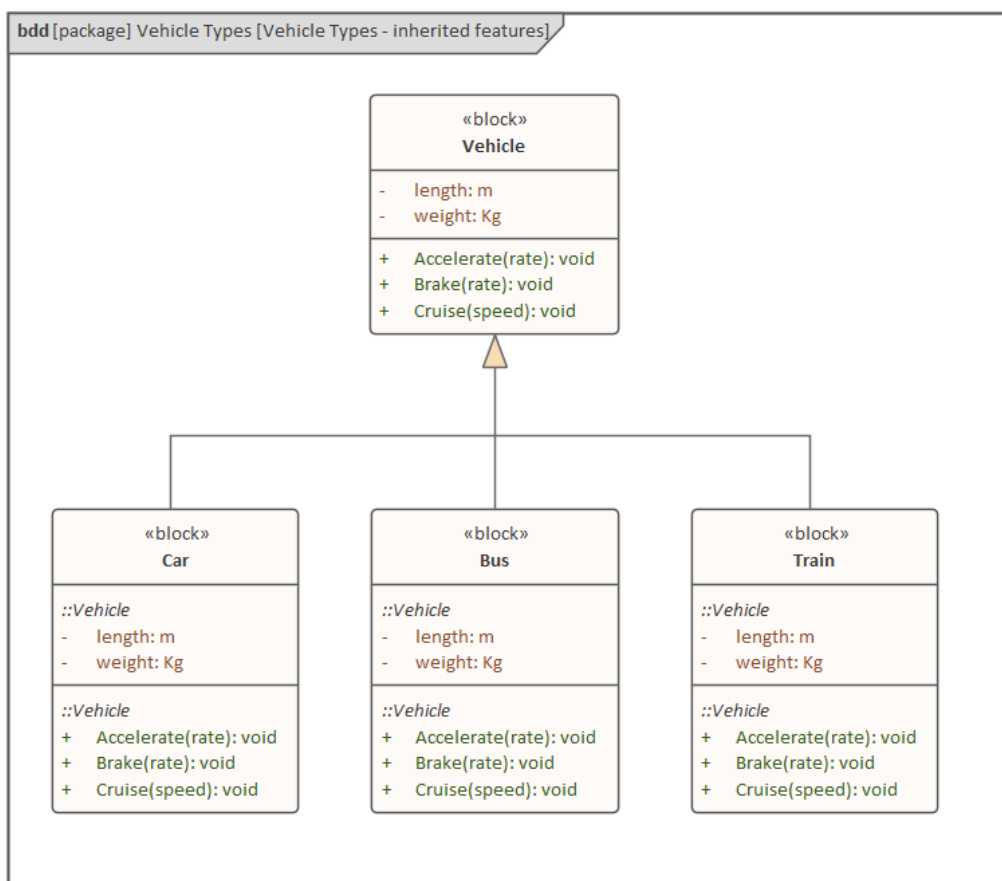
Lorsqu'un Bloc participe à une hiérarchie de généralisation et possède un certain nombre de spécialisations, les connecteurs émanant du Bloc peuvent devenir désordonnés. Enterprise Architect fournit un mécanisme permettant de modifier le style de ligne en fonction d'un certain nombre de styles, mais le style le plus utile est probablement le style d'arborescence orientée verticalement, qui regroupe les têtes de la relation et permet d'aligner leurs queues en parallèle.



L'un des mécanismes de langage utiles résultant de la généralisation est que les éléments spécialisés héritent des fonctionnalités structurelles et comportementales de l'élément généralisé. Jusqu'à présent, dans les diagrammes d'exemple, l'ingénieur a choisi de ne pas afficher ces fonctionnalités héritées, mais elles peuvent être configurées pour être affichées à l'aide des sections de compartiments de la feuille de propriétés de l'élément.



Le résultat sera que les blocs spécialisés afficheront les attributs et les opérations qui ont été hérités du Bloc parent. Ceux-ci seront affichés groupés par le nom du Bloc parent. Ce mécanisme est largement utilisé en ingénierie logicielle, mais il est également utile pour l'ingénieur système où le Bloc spécialisé hérite automatiquement des fonctionnalités de son parent en vertu de son statut de « membre de la famille ». Tout comme dans une famille humaine, un Bloc spécialisé (enfant) peut remplacer les fonctionnalités structurelles ou comportementales héritées d'un parent.



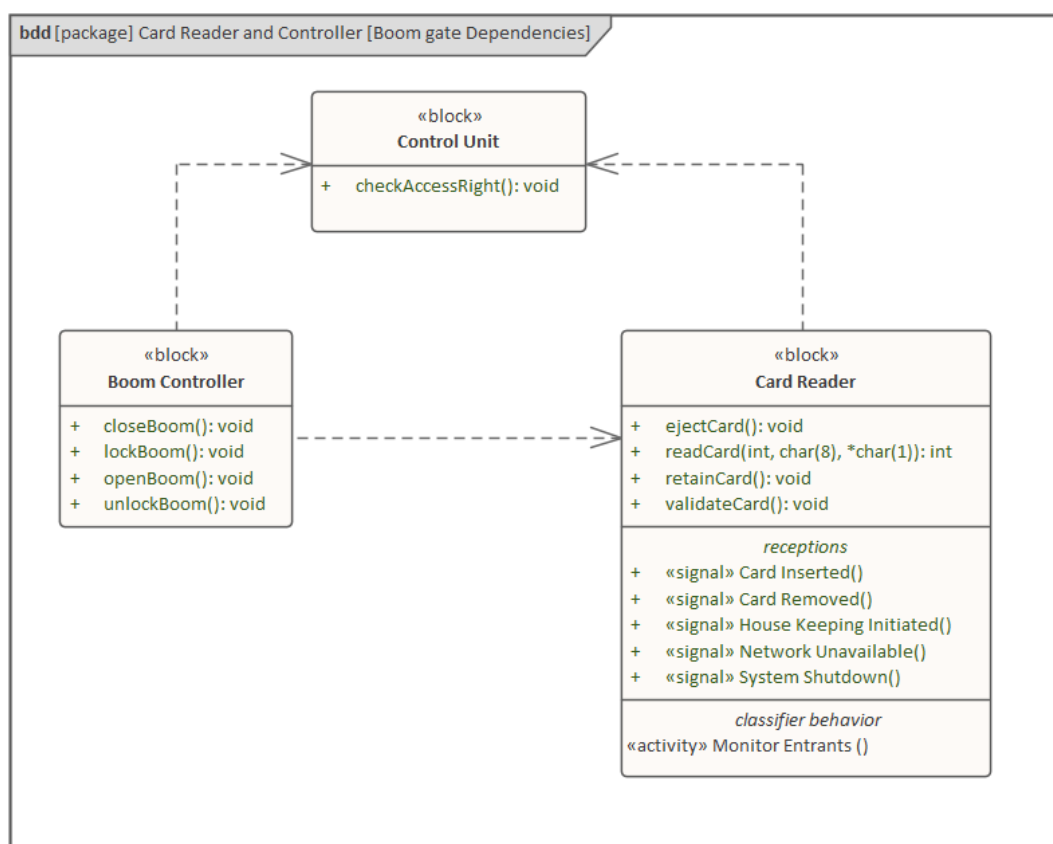
Les blocs appartiennent à des familles en fonction de certains critères, et cela peut être modélisé à l'aide de l'ensemble de

généralisation, qui est un mécanisme utilisé pour définir la base de l'appartenance à une famille.

## Dépendance

La dépendance est une relation utile mais sémantiquement faible. C'est le « pion » de la boîte à outils des ingénieurs en matière de relations, souvent utilisée au début du processus modélisation lorsque les détails des relations entre les éléments du système n'ont pas été analysés ou ne sont tout simplement pas connus. Elle modélise le fait que l'élément (client) à l'extrémité de la relation dépend d'une certaine manière de l'élément (fournisseur) à l'extrémité de la flèche de la relation. On peut pardonner aux modélisateurs novices de dessiner cette relation dans le sens inverse, car on pense souvent que le matériel passe du fournisseur au client. Une fois que la sémantique de la relation est comprise et que l'on se rend compte que la relation ne dit rien sur la direction du flux, l'erreur ne sera pas commise.

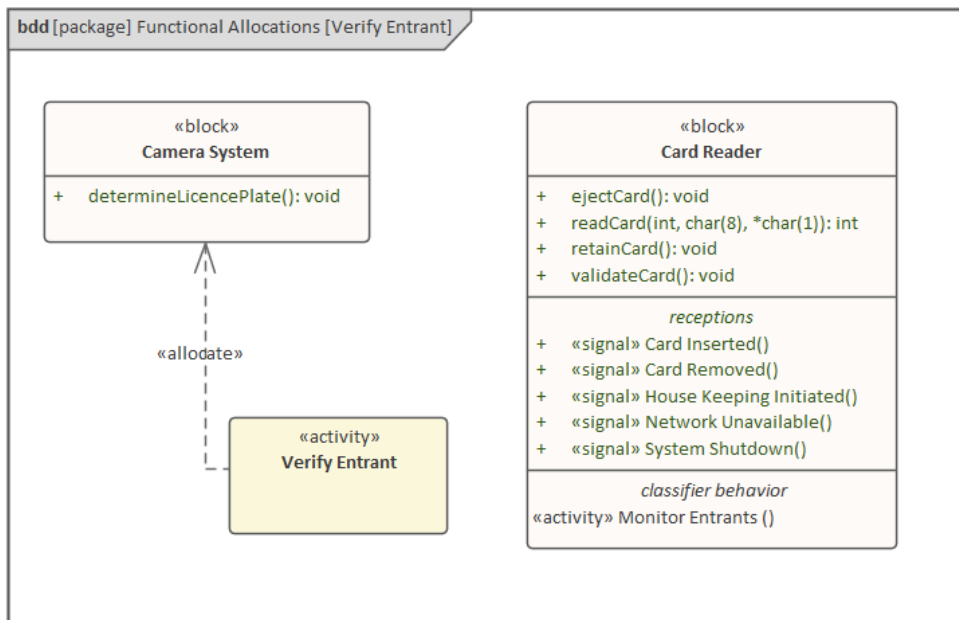
Il existe plusieurs types de dépendance, tous pris en charge par Enterprise Architect. Le connecteur peut être créé en sélectionnant l'icône « Dépendance » dans la page « Relations Bloc SysML » de la boîte à outils, puis en cliquant sur l'élément client (extrémité de la queue) et en faisant glisser le curseur jusqu'à l'élément fournisseur (extrémité de la pointe de flèche). Le connecteur peut également être créé à l'aide de la flèche Quick Linker dans le coin supérieur droit d'un élément diagramme sélectionné. Une fois la relation créée, un stéréotype peut être choisi dans la fenêtre Propriétés du connecteur pour rendre la dépendance plus spécifique. Cette capture d'écran montre tous les stéréotypes disponibles, dont certains sont utilisés entre différents types d'éléments autres que les blocs ; par exemple, Paquetages et Exigences .



## Répartition entre les blocs et les activités

La relation d'allocation peut être utilisée dans diverses circonstances, mais elle est particulièrement utile pour exprimer une relation fondamentale entre les deux éléments Comportementale et structurels les plus canoniques, à savoir l'activité et le Bloc. Cela ressemble à nos langues naturelles, où un verbe n'a pas de sens sans la présence d'un nom qui exécute

l'action décrite par le verbe. Ce type d'allocation est appelé allocation fonctionnelle, et l'ingénieur comble le fossé entre ces deux aspects d'un système en trouvant un Bloc qui peut exécuter le comportement décrit par une activité.



Dans ce diagramme l'ingénieur a créé deux relations d'allocation fonctionnelle qui décrivent comment le travail spécifié dans l'activité *Vérifier l'entrée* sera effectué. Une relation cible le système de caméra qui est utilisé pour capturer la plaque d'immatriculation du véhicule afin de déterminer si le véhicule particulier a été autorisé à entrer. L'autre relation cible le Bloc lecteur de carte qui est utilisé pour déterminer que le propriétaire de la carte a une relation avec la station de stationnement.

## Points d'interaction Modélisation

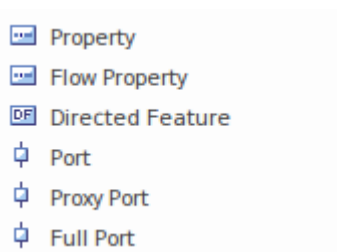
Les blocs - et les parties typées par les blocs - résident dans un environnement et interagissent avec cet environnement et les autres éléments qu'il contient. En termes généraux, le SysML fournit une construction de langage appelée Points d'interaction, qui sont des emplacements sur la bordure d'un élément qui agissent comme des points d'entrée et de sortie pour la communication avec l'élément propriétaire. Les ports sont un type de point d'interaction et fournissent un mécanisme permettant au Bloc d'exposer ses comportements - soit ceux qu'il possède de manière innée, soit ceux qui sont fournis par ses parties. Le port est représenté par un petit rectangle (généralement un carré) monté sur la bordure d'un Bloc ou d'une partie. SysML prend actuellement supporte deux types de port qui sont destinés à remplacer à terme les concepts antérieurs de port de flux et de port standard :

- *Port proxy* - Agit comme un relais pour exposer les fonctionnalités comportementales fournies par le Bloc propriétaire et est typé par une interface qui décrit ces services
- *Port complet* - Agit comme une partie et est typé par un Bloc , ce qui signifie qu'il peut lui-même contenir des parties

Enterprise Architect prend entièrement support ces deux types de ports et offre une compatibilité descendante avec les ports Standard et Flow antérieurs (qui sont toujours disponibles mais seront obsolètes dans les versions ultérieures de la norme).

Une fois qu'un diagramme de définition Bloc a été créé et qu'un Bloc a été placé sur le diagramme , les ports peuvent être créés soit :

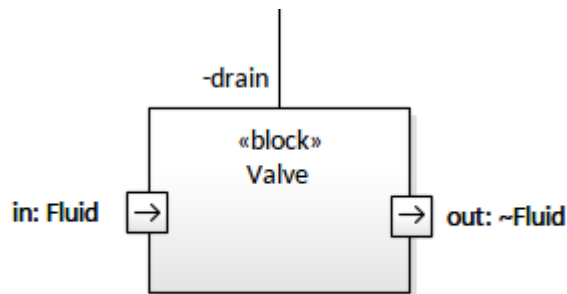
- Glissez-déposez le port approprié de la boîte à outils sur le Bloc - ce diagramme montre la section de la boîte à outils de définition Bloc qui répertorie les ports



- Sélectionnez l'option « Nouvel élément enfant » dans le menu contextuel du Bloc et sélectionnez le type de port approprié, comme indiqué :



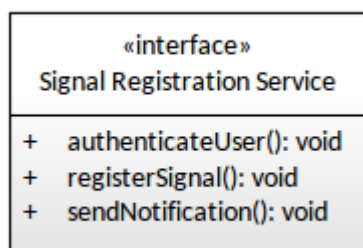
Quelle que soit la méthode choisie, le port sera alors automatiquement placé sur la bordure du Bloc et pourra être déplacé à la position requise et nommé. Note également que le port peut être ajouté à partir du panneau « Élément » du ruban « Conception ». Cette image d'écran est celle d'une section de diagramme montrant deux ports avec des indicateurs de direction à l'intérieur de l'élément Port. Les ports ont été nommés respectivement « in » et « out » et ont été typés par « Fluid », qui indique le type de l'élément arrivant au port.



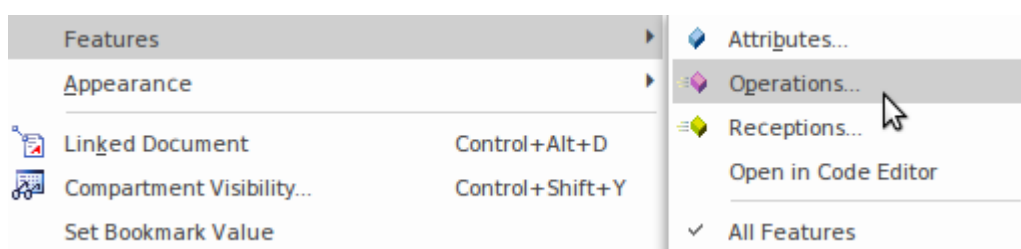
Les ports peuvent contenir des interfaces ainsi que des ports imbriqués ; Enterprise Architect fournit une fonctionnalité utile qui permet à l'ingénieur de personnaliser la taille du port, en transformant efficacement les petits carrés en petits rectangles.

## Interfaces et ports

Une interface est un moyen utile d'encapsuler un groupe de services fournis par un Bloc , offrant ainsi un moyen simple d'exposer ces services aux clients. L'interface a la même apparence qu'un Bloc et peut avoir des opérations et des réceptions définies, mais pas d'attributs ( Propriétés ).

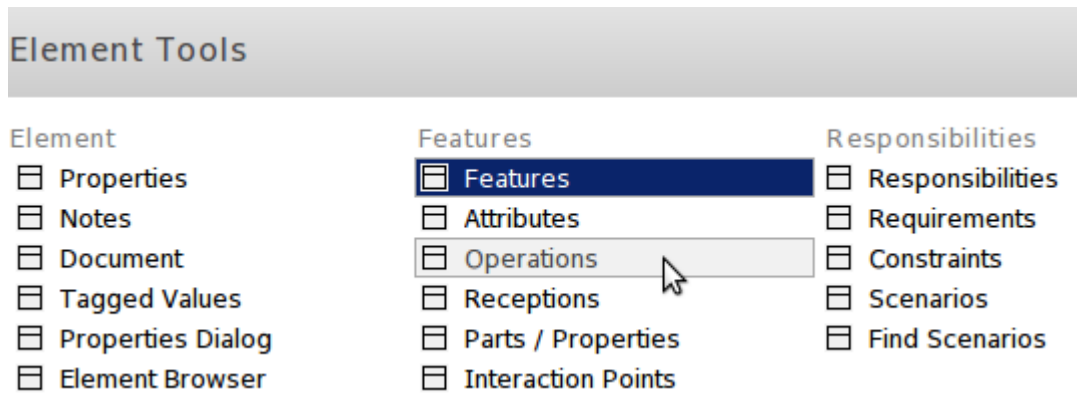


Une interface peut être créée en faisant simplement glisser l'icône « Interface » depuis la page Boîte à outils. fonctionnalités Comportementale peuvent être ajoutées en sélectionnant la Fonctionnalité appropriée dans le menu contextuel de l'interface.



Une autre façon simple de créer les opérations d'une interface est de sélectionner le Bloc dans un diagramme ou dans la fenêtre Navigateur et de cliquer sur l'élément du ruban :

Conception > Élément > Editeur > Fonctionnalités > Opérations



Les opérations sont créées simplement en sélectionnant l'onglet « Opérations » et en ajoutant le nom et d'autres détails dans une ligne de la fenêtre. N'importe quel nombre d'opérations peut être créé, et chaque opération peut définir n'importe quel nombre de paramètres qui spécifient les entrées et les sorties de l'opération. Les réceptions - l'autre fonctionnalité comportementale - peuvent être ajoutées de manière similaire à l'aide de l'onglet « Réceptions ». Chacun de ces éléments peut être réorganisé à l'aide des touches du clavier <Ctrl>+flèche vers le haut et vers le bas.

Des interfaces peuvent être ajoutées aux ports, qui constituent un mécanisme courant dans le Unified Modeling Language et permettent de publier les services disponibles sur un port. Les interfaces sont de deux types fondamentaux :

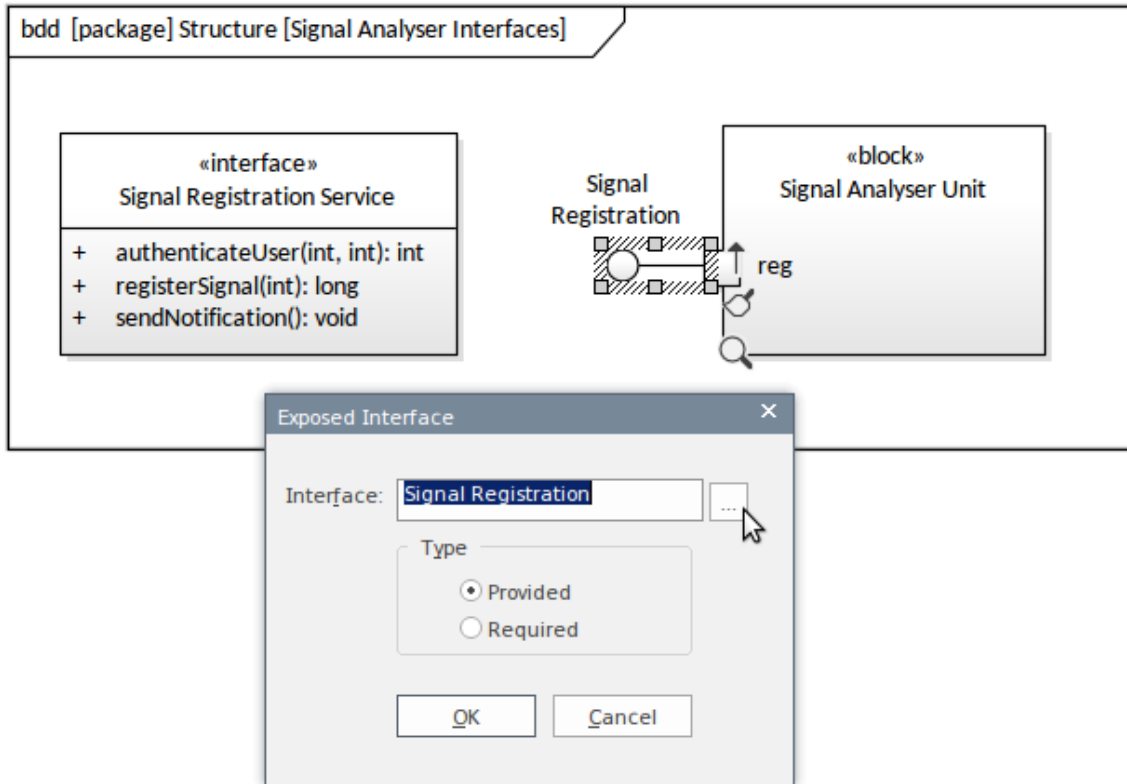
- *Fourni* - disponible pour utilisation
- *Obligatoire* - requis pour l'utilisation

Avec un Port sélectionné sur la bordure d'un Bloc dans un diagramme , une Interface peut être ajoutée comme pour tout autre élément structurel - à partir des éléments du menu contextuel du Port :

- Nouvel élément enfant | Fourni
- Nouvel élément enfant | Obligatoire

Les interfaces ajoutées de cette manière doivent être typées par une interface propre (un élément stéréotypé comme <<interface>>). Pour cela, sélectionnez d'abord l'interface dans le diagramme , puis sélectionnez la fenêtre Propriétés dans le menu contextuel de l'interface. Vous pouvez ensuite nommer l'interface et utiliser l'icône [...] pour naviguer ou rechercher l'élément Interface. Ce diagramme illustre l'étape pour une interface d'enregistrement de signal.



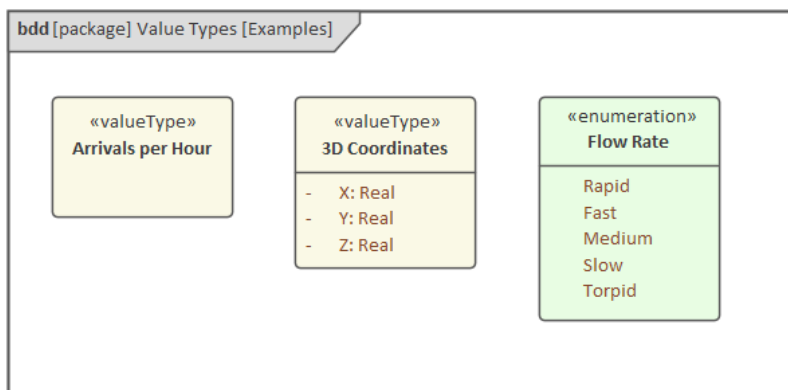


## Modélisation de la quantité à l'aide de types de valeur

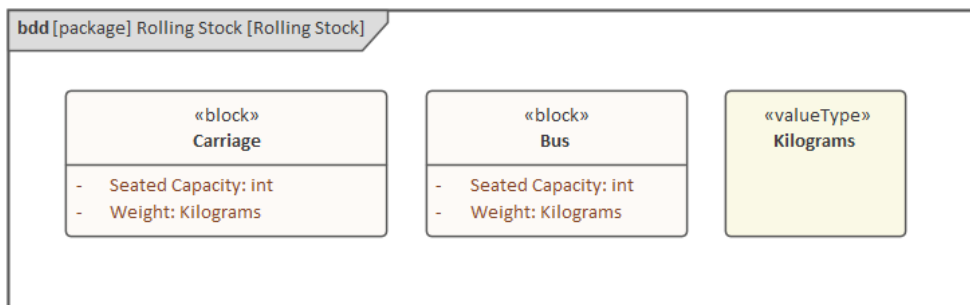
Un Bloc peut avoir des propriétés simples appelées Propriétés de valeur, qui définissent les attributs d'un Bloc qui représentent une quantité scalaire ou vectorielle. Les valeurs sont utilisées pour exprimer des informations sur un Bloc et fournir un emplacement pour une valeur réelle à saisir dans une instance d'un Bloc . Par exemple, un Réservoir peut avoir des propriétés de diamètre, de hauteur et de volume définies. Les types valeur permettent à un ingénieur de créer une manière universelle de quantifier une propriété ; par exemple :

- Une centrifugeuse a une vitesse maximale spécifiée en tours par minute (tr/min).
- Un wagon de train a un poids spécifié en kilogrammes (kg)
- Un réservoir a un volume spécifié en litres (l)
- Une source lumineuse a une intensité lumineuse spécifiée en Candela (cd)
- Une machine de dialyse a un débit sanguin spécifié en millilitres par minute (ml/min)

Ce diagramme montre un certain nombre de types de valeur différents qui peuvent être définis dans Enterprise Architect , puis appliqués à n'importe quel nombre d'attributs définis dans des blocs.



L'objectif du Type de valeur est de permettre à un ingénieur, à une équipe ou à une industrie de définir des types standard qui peuvent être réutilisés pour caractériser les propriétés valeur définies pour un nombre quelconque de blocs. Par exemple, le type de valeur « Kilogramme » peut être appliqué à une propriété valeur spécifiant le poids d'un train ou le poids d'un bus ou la capacité d'accueil de l'un ou l'autre.



Ces types valeur , comme leur nom l'indique, ont une valeur qui décrit la quantité de la propriété ; ils peuvent être définis à l'aide de l'onglet « Attribut » de la fenêtre Fonctionnalités , comme illustré ici.

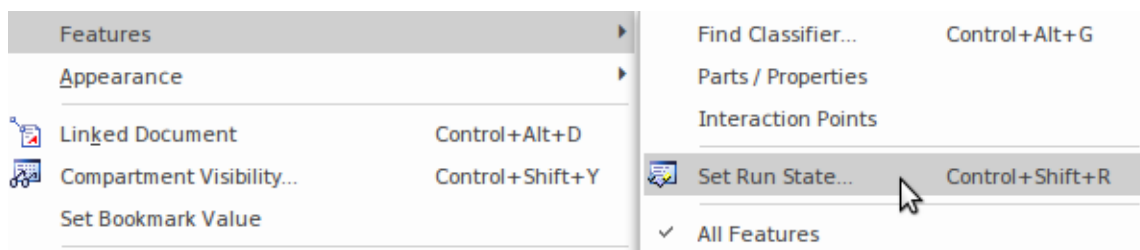
Name	Type	Scope	Stere...	Alias	Initial ...
Seated Capacity	int	Private			
Weight	Kilograms	Private			

New Attribute...

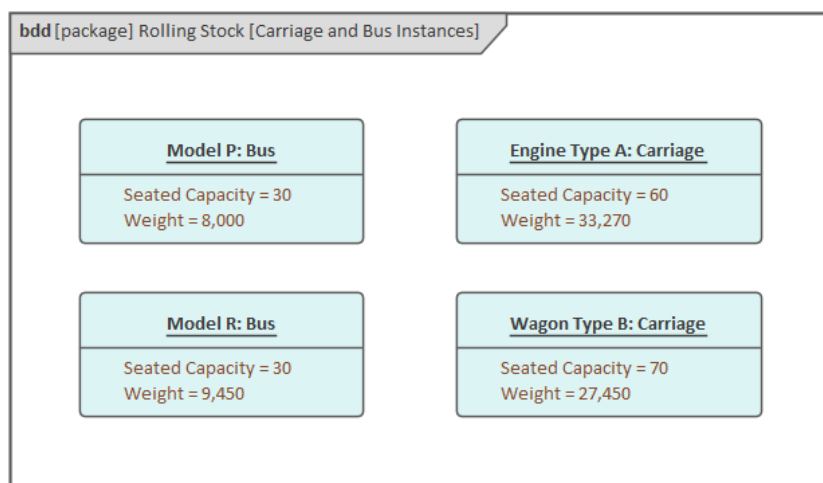
Par exemple, deux blocs représentant un train et un bus peuvent avoir une propriété « Poids » définie et typée par le type valeur « Kilogramme ». Si un attribut tel que « Nombre de places assises » a un type simple tel que « Integer », celui-ci peut être directement sélectionné dans la liste déroulante « Type », mais si « Type » est basé sur un Type de valeur, celui-ci peut être sélectionné à l'aide de l'option « Sélectionner le type... » de la liste déroulante.

Les instances de blocs qui ont un attribut (propriété de valeur) défini dans Enterprise Architect peuvent avoir une valeur réelle spécifiée pour l'attribut. Par exemple, chaque instance de bus et de train avec, par exemple, un numéro de modèle particulier peut avoir un poids différent défini. D'autres propriétés telles que « Capacité de sièges » peuvent avoir un type primitif « Integer » défini, et celles-ci peuvent également être définies pour des instances particulières de « Voiture ».

Enterprise Architect permet à un ingénieur de définir les valeurs de chacun des attributs définis en utilisant l'option « Définir State Exécuter » du sous-menu « Fonctionnalités » pour une instance Bloc .



Les valeurs d'attribut (emplacements) peuvent être affichées sur un diagramme , permettant à un ingénieur de créer des exemples convaincants ou des catalogues d'instances Bloc , comme illustré ici.



## Utilisation Propriétés et des pièces pour l'utilisation Modèle Bloc

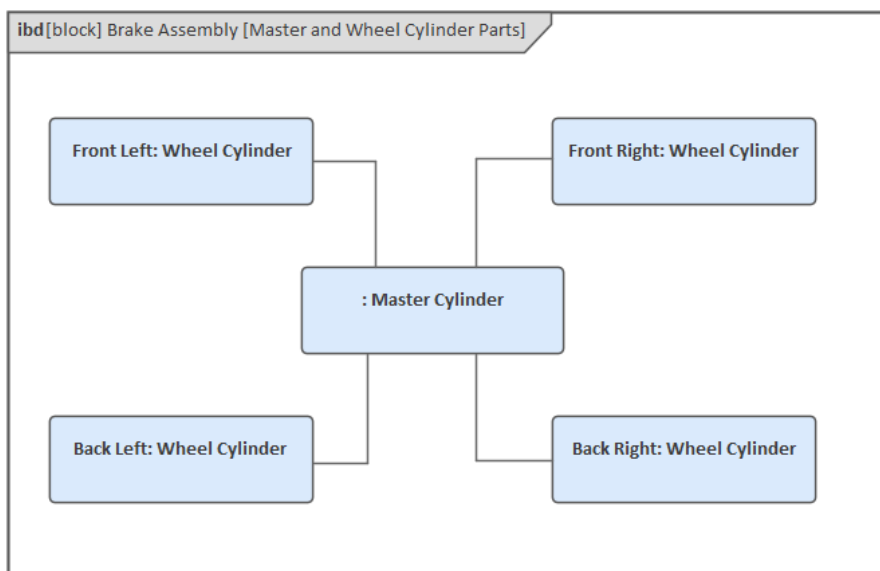
Les blocs sont des classificateurs et décrivent les caractéristiques d'un ensemble d'éléments qui représentent la manière dont le Bloc est utilisé dans un contexte. Lorsque le Bloc possède des attributs ( valeur propriétés) définis, des valeurs spécifiques sont attribuées dans les instances du Bloc . En effet, chaque instance Bloc possède une identité et se voit généralement attribuer des valeurs différentes qui définissent l'état du Bloc . Enterprise Architect permet de spécifier ces valeurs à l'aide d'une option *Set Exécuter State* disponible dans le menu contextuel.

diagrammes Bloc internes montrent souvent comment les parties d'un Bloc sont connectées entre elles dans un contexte d'utilisation. Enterprise Architect permet de faire glisser les blocs de la fenêtre Navigateur vers un diagramme et de les déposer en tant que Propriétés de partie. Il s'agit en fait de pièces et elles représentent des instances du classificateur Bloc . Un ingénieur a la possibilité de les nommer dans le contexte du diagramme . Par exemple, ce diagramme représente un ensemble de freins qui a été modélisé, ce qui est indiqué dans le cadre diagramme dans ce format :

ibd [bloc] Ensemble de frein [pièces du maître-cylindre et du cylindre de roue]

- *ibd* - signifie qu'il s'agit d'un diagramme Bloc internes
- *bloc* - indique le type d'élément dû
- *Ensemble de frein* - est le nom de l'élément
- *Pièces du maître-cylindre et du cylindre de roue* - est le nom du diagramme

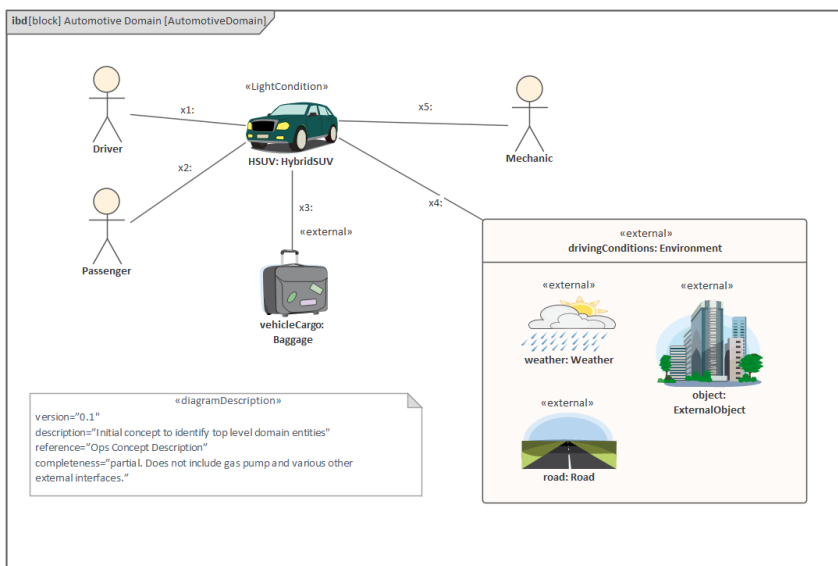
L'ingénieur a nommé chacune des pièces du cylindre de roue (avant gauche, avant droit, arrière gauche, arrière droit) car elles doivent être identifiées par rapport à leur emplacement dans le véhicule, mais a décidé de ne pas nommer le maître-cylindre car aucune qualification supplémentaire n'est requise.



## Présentation des Diagrammes Fonctionnels Internes

Un diagramme Bloc interne permet de visualiser la structure interne d'un Bloc , y compris ses Propriétés et ses parties, ainsi que la manière dont ces parties sont liées les unes aux autres. Le diagramme n'est pas tenu d'afficher toutes les parties qui composent un Bloc donné et il est courant pour un ingénieur de créer un diagramme qui se concentre sur un aspect particulier d'un système ou d'un sous-système.

Le cadre d'un diagramme Bloc interne représente le Bloc propriétaire, il sera donc nommé comme tel et les éléments qui apparaissent sur le diagramme seront des parties qui sont des instances des blocs dont le Bloc propriétaire est composé. Ce diagramme Bloc interne montre une instance d'un véhicule dans un contexte donné ; il utilise un certain nombre d'images à la place des symboles conventionnels du langage SysML afin d'ajouter de l'attrait et de rendre le diagramme plus convaincant pour un public non ingénieur. Pour plus d'informations, consultez la rubrique d'aide sur le [Internal Block Diagrams](#) .



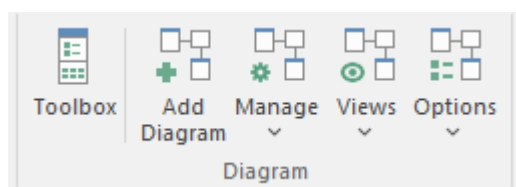
## Création d'un Diagramme de Principe Interne

Un diagramme Bloc internes peut être créé à partir de plusieurs emplacements dans l' Interface Utilisateur , tels que :

- Ruban de conception - *Diagramme* Icône de diagramme sur le panneau *Diagramme*
- Barre d'outils de la fenêtre Navigateur - Icône *Nouveau Diagramme*
- Menu contextuel de la fenêtre Navigateur - *Ajouter Diagramme*

Nous allons utiliser le ruban Conception pour créer un diagramme Bloc internes. Tout d'abord, sélectionnez l'emplacement dans la fenêtre Navigateur où vous souhaitez placer le diagramme . Contrairement à la plupart des autres diagrammes SysML, le diagramme Bloc internes est généralement inséré sous son propre Bloc . Une fois l'emplacement sélectionné dans la fenêtre Navigateur , sélectionnez l'élément du ruban :

Conception > Diagramme > Ajouter Diagramme

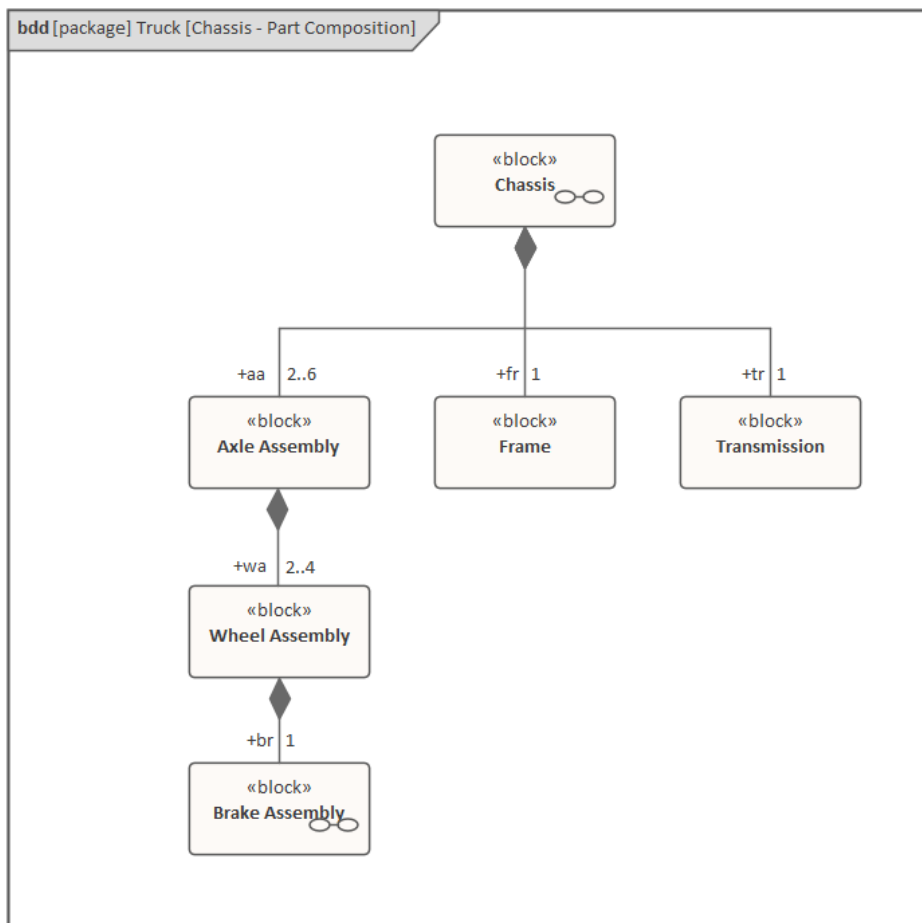


La sélection de cette option ouvre la dialogue *Nouveau Diagramme* , qui vous permet de nommer le diagramme ; le nom diagramme est par défaut le nom du Bloc qui contient le diagramme . Avec la perspective SysML choisie et la version de SysML sélectionnée, une liste de types diagramme s'affiche à partir de laquelle vous sélectionnez le diagramme Bloc

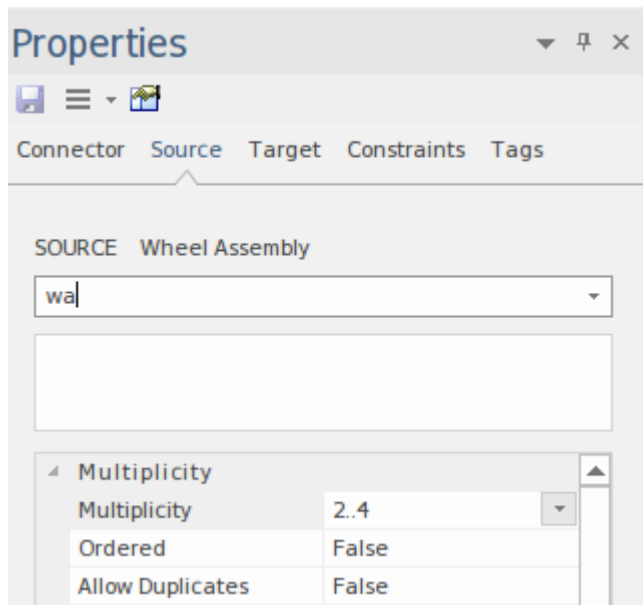
interne. Cliquez sur le bouton OK pour créer un nouveau diagramme Bloc interne à l'emplacement sélectionné dans la fenêtre Navigateur . Le canevas diagramme s'ouvre, vous permettant de commencer à ajouter des éléments et des connecteurs qui décrivent la structure interne du Bloc . Enterprise Architect affiche également la page « Bloc interne » de la boîte à outils, qui contient les éléments et les relations définis par la spécification SysML comme applicables à la construction de ce type de diagramme . N'importe quel nombre d'autres pages de la boîte à outils peut être ouvert si nécessaire, en plus des pages « Éléments communs » et « Relations communes » qui sont toujours disponibles.

## Modélisation et assemblage des pièces

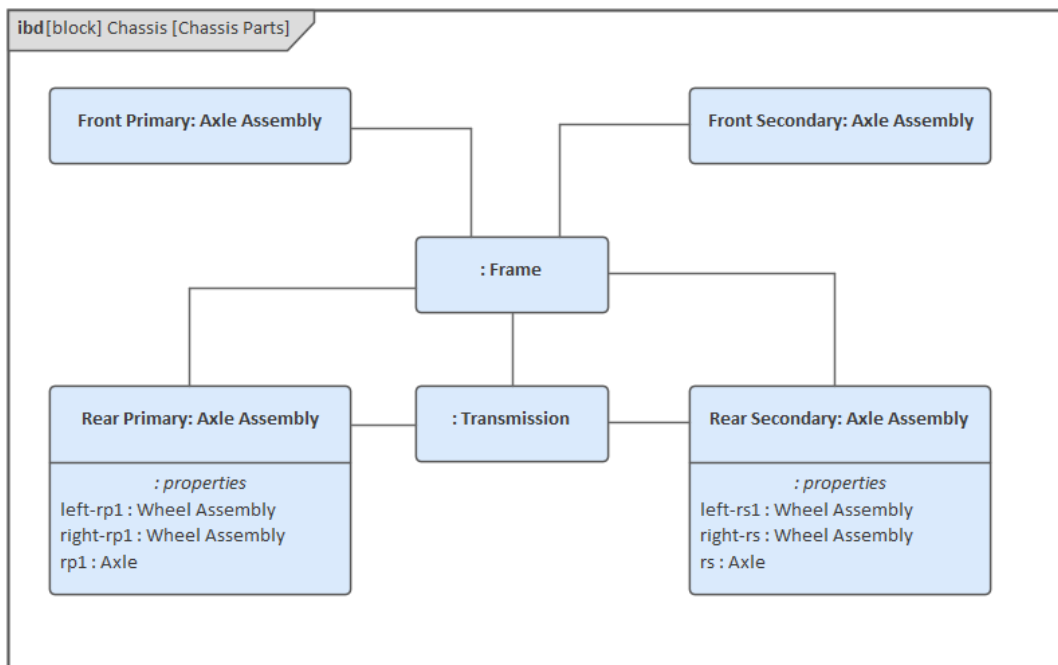
Lorsqu'un Bloc est composé d'un certain nombre d'autres Blocs, il arrive généralement que lorsque le système est instancié, les Blocs contenus effectuent une grande partie du travail requis du Bloc propriétaire. Le diagramme Bloc interne fournit un mécanisme de langage pour visualiser la manière dont les pièces interagissent, pour montrer la structure des Blocs dans le contexte et pour fournir le comportement global spécifié par le Bloc propriétaire. Dans une rubrique précédente, nous avons vu un diagramme de définition Bloc du châssis d'un camion, décrivant les Blocs qui composent le châssis en fonction des Blocs.



Le diagramme inclut des multiplicités aux extrémités des parties de l'association, montrant combien d'éléments d'une partie particulière peuvent être inclus dans une seule instance du Bloc propriétaire. Ces nombres représentent la cardinalité exprimée sous forme de limite supérieure et inférieure, qui peut être définie dans la fenêtre Propriétés du connecteur. Cette capture d'écran montre une partie de la fenêtre Propriétés utilisée pour définir les multiplicités et autres propriétés de l'extrémité de l'association, qui ajoutent toutes une sémantique riche à l'association.



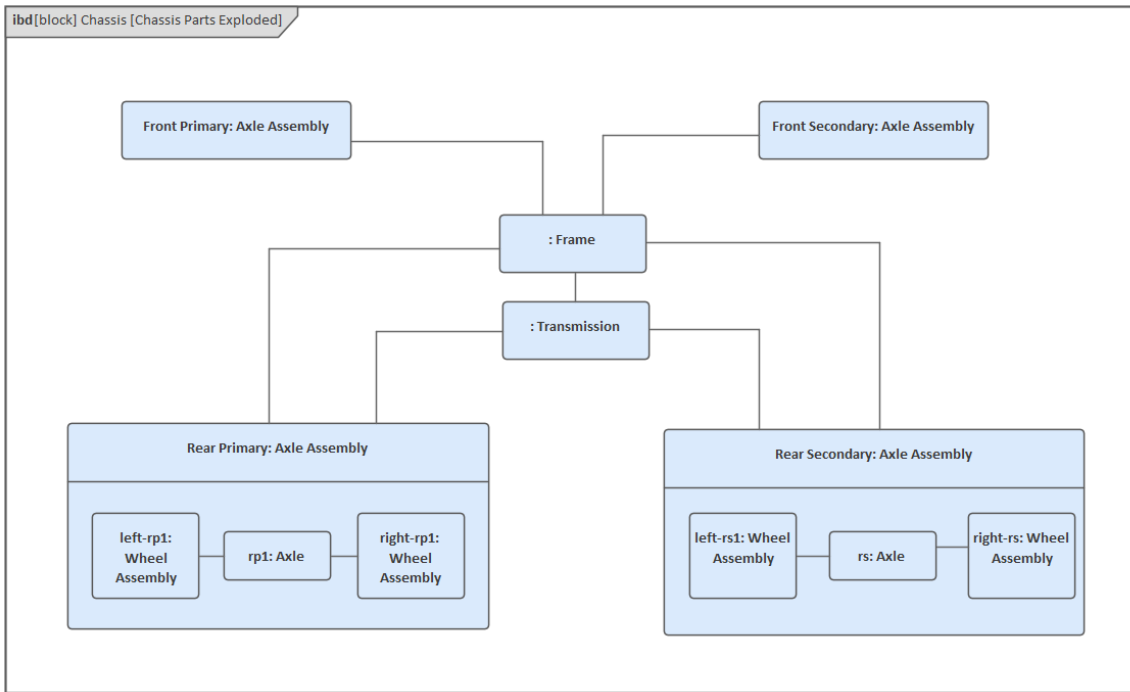
Ce diagramme Bloc interne du châssis montre les pièces qui le composent dans un contexte réel. Le nombre d'essieux est défini pour le cas particulier du camion.



Enterprise Architect permet à un ingénieur de créer diagrammes avec des pièces imbriquées à n'importe quel niveau, ce qui permet de démontrer la structure d'un Bloc et la manière dont les pièces seraient connectées dans un contexte réel.

Ce diagramme montre des parties imbriquées sur deux niveaux, mais n'importe quel nombre de niveaux est possible et peut être créé sur un diagramme . Ce type d'expression peut conduire à diagrammes assez volumineux, et Enterprise Architect supporte des formats de papier allant jusqu'à A0, ce qui permet de créer et d'imprimer de grands diagrammes .





Dans le diagramme les ensembles d'essieux primaires et secondaires arrière ont été représentés en détail, où chaque essieu est composé d'un ensemble de roues droite et gauche, qui pourraient eux-mêmes être représentés comme une structure imbriquée comprenant l'ensemble de frein, qui à son tour pourrait montrer les ensembles de cylindres de roue.

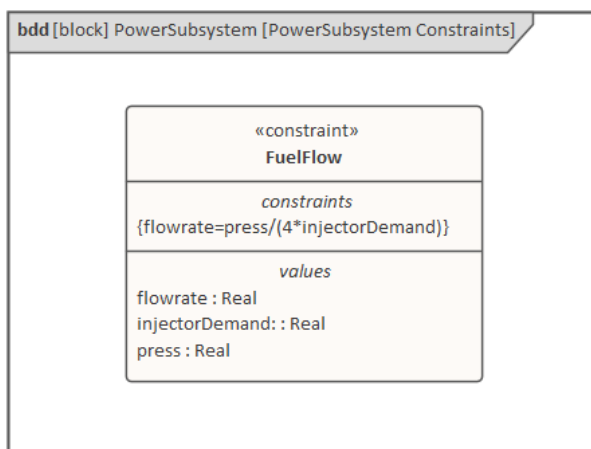
Dans le cas où une pièce est ajoutée au diagramme mais que le modelleur doit pour une raison quelconque modifier le classificateur Bloc qui saisit le Bloc , cela peut être fait à partir du menu contextuel « Pièces » qui fournit une option permettant de modifier le Type de propriété. Cela peut être pratique lorsqu'un diagramme élaboré a été créé et qu'il est plus facile de laisser l'élément visuel en place et de simplement mettre à jour le Bloc sur lequel il est basé.



## Modélisation Paramétriques Equations

Les ingénieurs sont chargés de trouver des solutions aux problèmes et aux opportunités, et utilisent des modèles comme moyen de visualiser les simplifications du système considéré et du contexte mondial dans lequel le système devra fonctionner. Les modèles d'ingénierie système créés dans Enterprise Architect fournissent un outil précieux pour l'analyse, la conception, architecture, les tests et la visualisation. Cela inclut la capacité à prédire le comportement d'un système dans un contexte donné, en équilibrant les exigences concurrentes et les considérations de conception sous la forme de négociations entre les parties prenantes et d'analyses de compromis. diagrammes Paramétriques sont un outil avancé qui peut aider l'ingénieur à répondre à ces préoccupations dans un modèle et à représenter de manière préventive le comportement probable d'un système.

Dans une rubrique précédente, nous avons appris comment les équations peuvent être modélisées à l'aide du diagramme de définition Bloc, avec la relation d'association de pièces articulant les variables (paramètres) de l'équation. Cela fournit un mécanisme essentiel pour la réutilisation. Pour vous rappeler comment nous utilisons un Bloc de contraintes pour modéliser les équations, reportez-vous à ce diagramme, qui utilise une contrainte (qui est un Bloc stéréotypé) pour modéliser le carburant d'un véhicule. Le débit de carburant est basé sur une équation dont la pression de carburant (press) et la demande de carburant sont des variables (paramètres).



Toute équation ou système d'équations peut être modélisé à l'aide de la contrainte.

Cette contrainte peut potentiellement être réutilisée dans de nombreux contextes différents. C'est sur le diagramme Paramétriques que nous voyons comment elle est utilisée. Pour plus d'informations, consultez la rubrique d'aide [Parametric Diagrams](#).

# Présentation Diagrammes Paramétriques

Un diagramme Paramétriques permet de visualiser les équations et leurs paramètres dans un contexte particulier sous la forme de propriétés de contrainte. Chacune de ces propriétés représente une utilisation d'un bloc de contrainte qui a généralement été défini sur un diagramme de définition Bloc .

Le cadre d'un diagramme Paramétriques représente le ConstraintBlock propriétaire, il sera donc nommé comme tel et les éléments qui apparaissent sur le diagramme seront des propriétés de contrainte, qui sont des instances des ConstraintBlocks dont le Bloc propriétaire est composé, montrant ainsi la composition de la contrainte.

## Création de Diagrammes Paramétriques

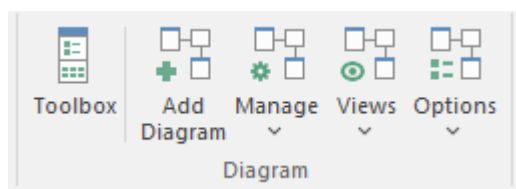
Un diagramme Paramétriques peut être créé à partir de plusieurs emplacements dans l' Interface Utilisateur , en utilisant l'une de ces options :

- Ruban de conception - *Ajouter une icône Diagramme* sur le panneau *Diagramme*
- Barre d'outils de la fenêtre Navigateur - *Icône Nouveau Diagramme*
- Menu contextuel de la fenêtre Navigateur - *Ajouter Diagramme*

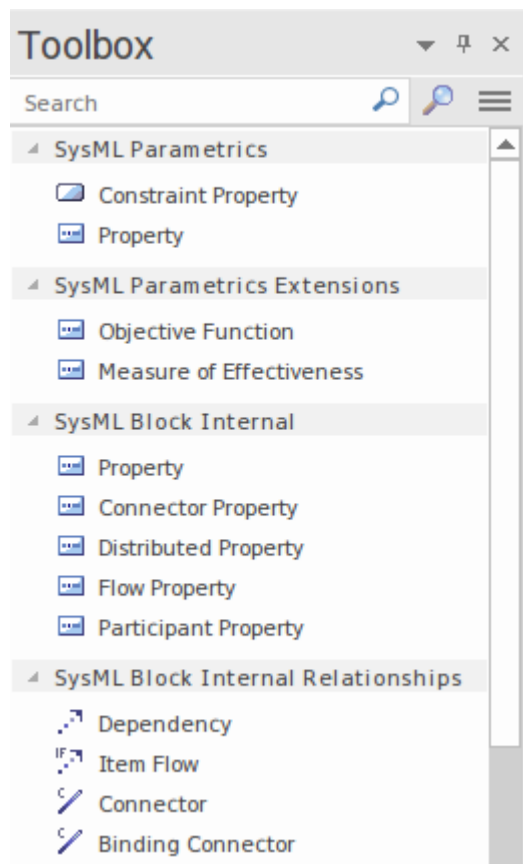
Nous allons utiliser l'option 'Conception' du ruban pour créer un diagramme Paramétriques . Tout d'abord, vous sélectionnez le Paquetage dans la fenêtre Navigateur où vous souhaitez placer le diagramme .

Contrairement à la plupart des autres diagrammes SysML, le diagramme Paramétriques est généralement inséré sous son propre Bloc de contraintes. Une fois l'emplacement sélectionné dans la fenêtre Navigateur , sélectionnez :

Conception > Diagramme > Ajouter Diagramme



La sélection de cette option ouvre la dialogue « Nouveau Diagramme », qui vous permet de nommer le diagramme ; le nom sera par défaut celui du Bloc de contraintes auquel appartient le diagramme . Avec la perspective SysML choisie et la version de SysML sélectionnée, une liste de diagrammes s'affiche, vous permettant de sélectionner le diagramme Paramétriques . Cliquez sur le bouton OK pour créer un nouveau diagramme Paramétriques à l'emplacement sélectionné dans la fenêtre Navigateur . La Vue Diagramme s'ouvre, vous permettant de commencer à ajouter des éléments et des connecteurs qui décrivent les équations et les paramètres. Enterprise Architect affiche également les pages « Paramétriques » de la boîte à outils Diagramme , qui contiennent les éléments et les relations définis par la spécification SysML comme applicables à la construction de ce type de diagramme . N'importe quel nombre d'autres pages de la boîte à outils peut être ouvert si nécessaire, en plus des pages « Éléments communs » et « Relations communes » qui sont affichées par défaut et qui permettent d'ajouter notes diagramme , des légendes et d'autres éléments communs.



Les éléments et connecteurs les plus importants utilisés avec le diagramme Paramétriques sont :

### Éléments

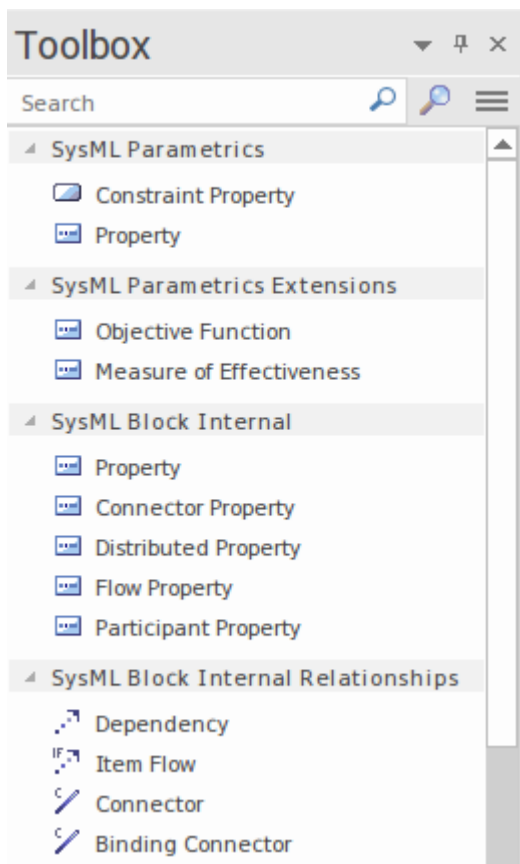
- Propriété de contrainte
- Propriété

### Extensions d'éléments

- Fonction objective
- Mesure de l'efficacité

### Connecteurs

- Dépendance
- Flux Item
- Connecteur
- Connecteur de liaison

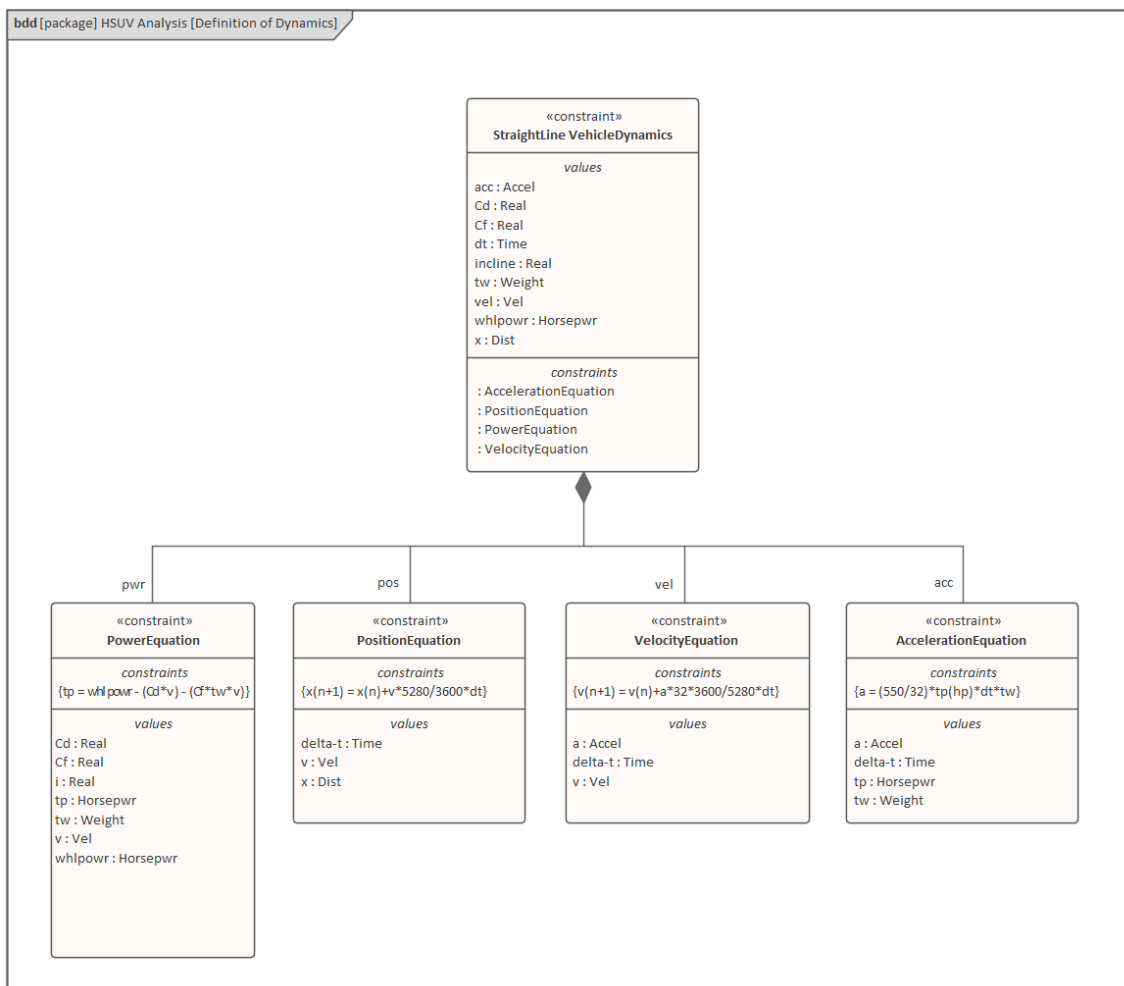


Des éléments peuvent être ajoutés au diagramme en les faisant glisser et en les déposant depuis la page Boîte à outils sur le canevas diagramme .

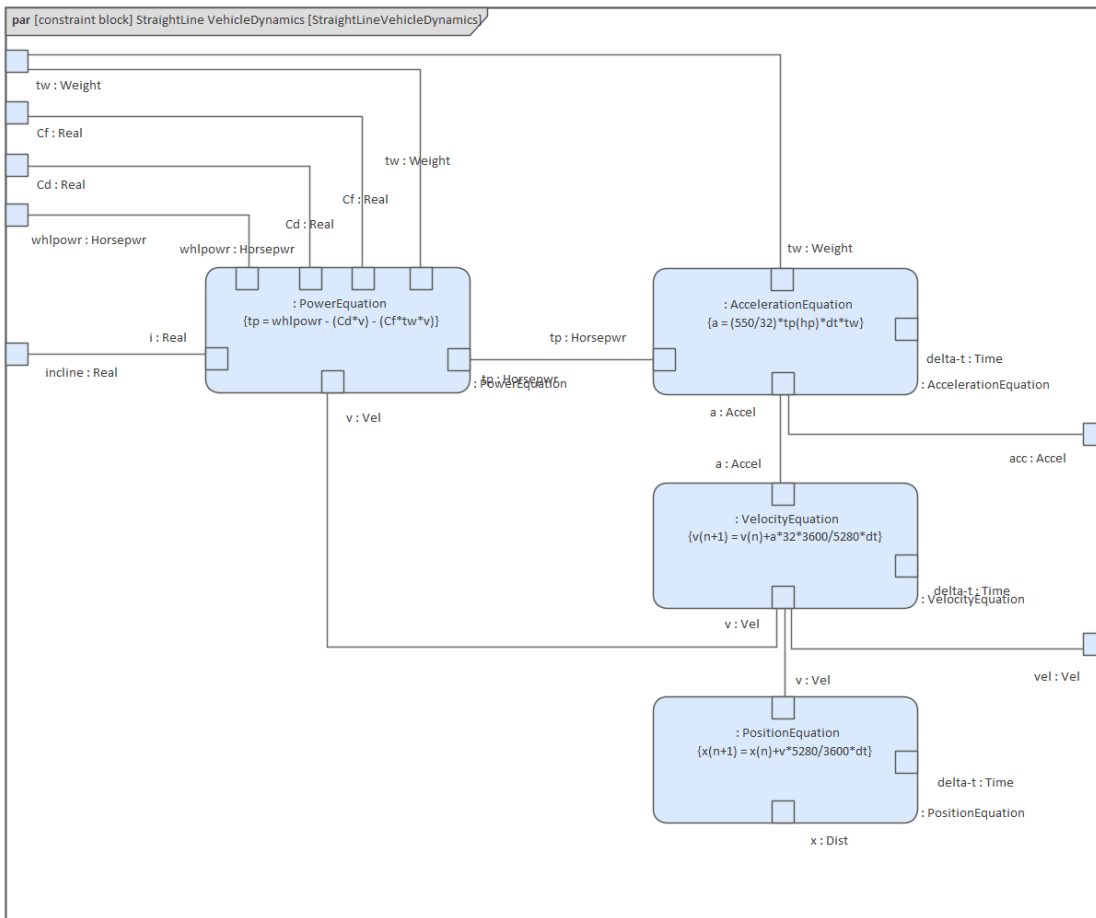
# Systemes d'equations utilisant des associations de parties

Les problemes et systemes d'ingenierie necessitent generalement une analyse detaillee pour determiner les performances d'une solution proposee particuliere. L'analyse peut impliquer un nombre quelconque d'equations qui sont souvent liees les unes aux autres pour determiner une valeur particuliere. Enterprise Architect permet a un ingenieur de construire des systemes d'equations a l'aide d'un diagramme de definition Bloc , puis d'utiliser ces equations dans plusieurs diagrammes Parametriques pour decrire les solutions proposees.

Ce diagramme de definition Bloc decrit les parametres de la dynamique en ligne droite d'un vehicule - le HSUV - et comprend un certain nombre d'equations qui sont representees sur le diagramme sous forme de contraintes, qui sont un type de Bloc .

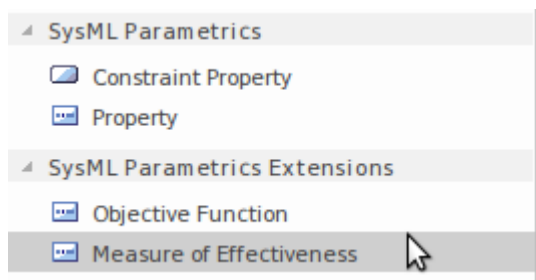


Le diagramme suivant, un diagramme Parametriques , montre comment les ConstraintBlocks sont utilises dans un contexte particulier, representes sur le diagramme sous la forme de ConstraintProperties. Nous pouvons visualiser comment le parametre de puissance totale est calcule, avec une connexion entre l'equation de puissance et le parametre equivalent sur l'equation d'acceleration. Des connexions peuvent etre observees entre l'equation de position et l'equation de vitesse, qui est finalement reliee a l'equation d'acceleration.



# Mesures d'efficacité à l'aide de paramètres

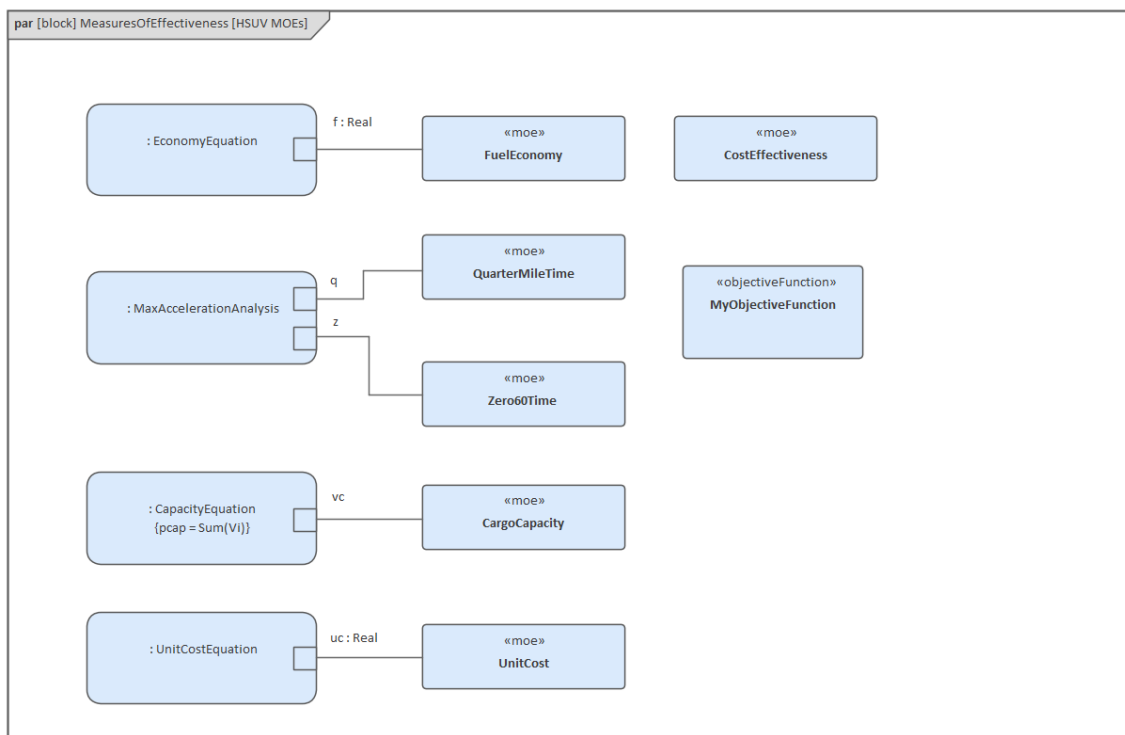
Les mesures d'efficacité (MOE) sont un mécanisme d'ingénierie important utilisé pour quantifier la réalisation des objectifs de mission ou des résultats souhaités spécifiés. Elles peuvent être modélisées à l'aide du diagramme Paramétriques, et l'élément Mesure d'efficacité est disponible à partir de la page « Paramétriques » de la boîte à outils Diagramme, à partir de laquelle il peut être glissé sur un diagramme et lié aux paramètres des équations représentées sous la forme de propriétés de contrainte.



Les mesures d'efficacité peuvent être réutilisées pour évaluer un nombre quelconque d'alternatives de conception et permettre de comparer et d'évaluer systématiquement ces conceptions. Ce diagramme montre la rentabilité globale d'un SUV hybride (HSUV) pour une solution proposée intitulée « Alternative One » (alt1) sur la base d'un certain nombre de mesures d'efficacité, notamment :

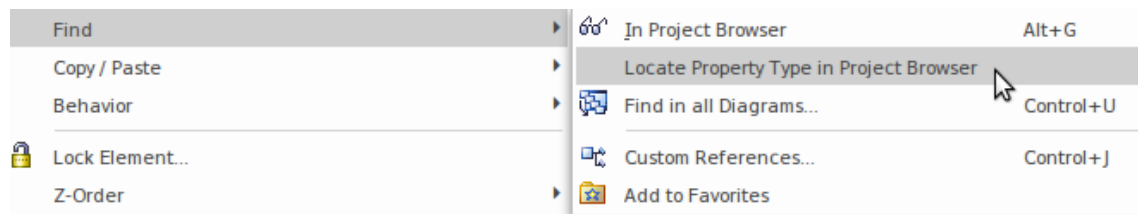
- FuelEconomy - Expression de l'économie de carburant
- QuarterMile - Temps nécessaire pour parcourir 0,25 miles
- Zero60Time - Temps nécessaire pour accélérer à 60/mph
- CargoCapacity - Le volume des espaces de chargement
- UnitCost - Coût du véhicule

Chacune de ces alternatives aurait son propre modèle Paramétriques qui serait capable de déterminer la valeur effective et de contribuer à l'équation globale, qui est une série de sommes pondérées  $\{CE = \text{Somme}(W_i * P_i)\}$ . Enterprise Architect permet de définir un nombre quelconque d'alternatives et l'ingénieur peut réutiliser les éléments MOE et les ConstraintBlocks utilisés pour définir les équations contributives.





Enterprise Architect dispose d'une fonctionnalité de recherche utile qui permet de localiser le type sur lequel une propriété est basée dans la fenêtre Navigateur . Cette fonction est particulièrement utile pour trouver le Bloc ou la contrainte propriétaire d'un Bloc ou d'une ConstraintProperty trouvé sur diagrammes Paramétriques et Bloc internes.



Il est possible de paramétrer un diaporama en utilisant l'onglet ' Modèle Vues ' de la fenêtre Focus . Pour plus d'informations, consultez la rubrique d'aide [Diagram Slide Show](#) .

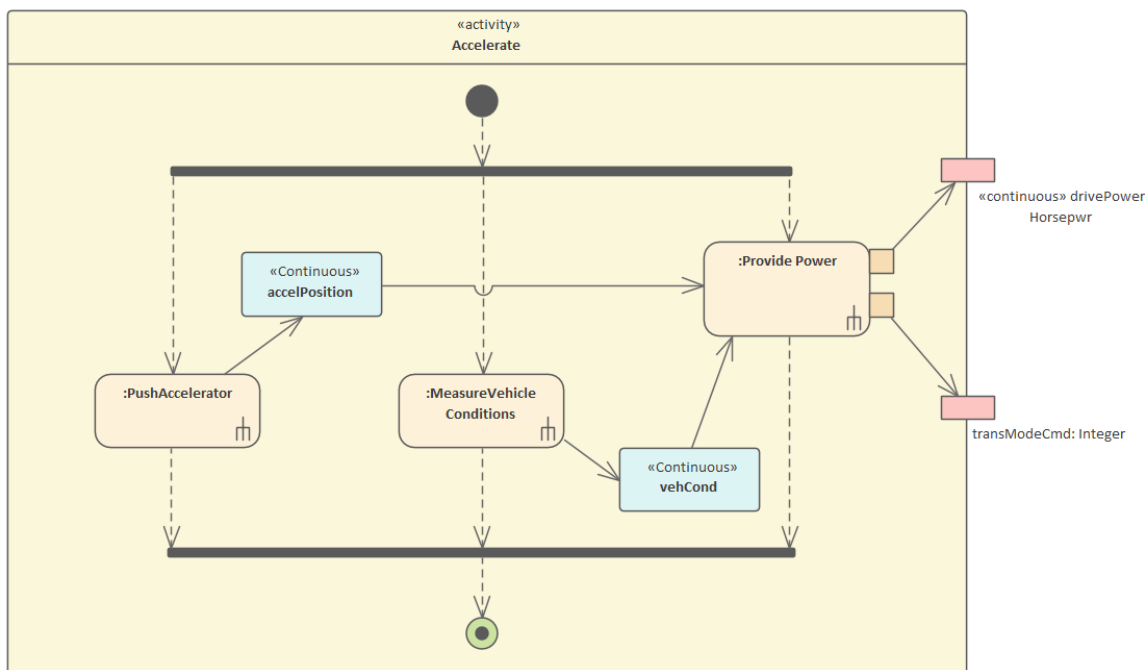
## Coordonner le comportement avec les activités

Comme nous l'avons vu dans un sujet précédent, le Systems Modeling Language (SysML) présente deux aspects fondamentaux qui sont analogues à deux catégories grammaticales importantes dans les langues naturelles que les humains utilisent pour communiquer, à savoir *les noms* et *les verbes*. Dans le SysML, il s'agit de constructions *structurelles* et *Comportementale*; les constructions structurelles étant analogues aux noms dans nos langues naturelles, et les constructions Comportementale étant analogues aux verbes.

Nous avons évoqué les aspects structurels du langage dans les chapitres précédents, lorsque nous avons discuté Paquetages et des Blocs. Nous allons maintenant nous intéresser au diagramme Comportementale principal, à savoir le diagramme d'activité. Il existe un certain nombre d'autres diagrammes de comportement, et le comportement est visible dans diagrammes structurels sous la forme d'opérations et également dans le comportement qui est directement attribué à un Bloc.

Si le nouveau venu dans SysML, en visualisant les diagrammes d'activité pour la première fois, pourrait se souvenir du diagramme de flux, il apprendra bientôt que le diagramme d'activité a une syntaxe et une sémantique qui vont bien au-delà du diagramme de flux. Le diagramme d'activité est formellement basé sur une branche des mathématiques appelée *réseaux de Petri* et il utilise un système de jetons pour indiquer à la fois la séquence d'actions et les éléments qui circulent dans le système. Les éléments qui circulent peuvent être des éléments d'information, des éléments physiques ou même des signaux de contrôle. Nous ferons référence à ce système de jetons pour éclairer le fonctionnement du diagramme d'activité.

Ce diagramme, décrivant l'accélération d'un véhicule, reprend de nombreux éléments que l'on retrouve généralement sur un diagramme d'activité. Vous verrez dans les sujets suivants qu'il s'agit d'un diagramme très expressif qui, s'il est élaboré avec soin, peut transmettre de manière rigoureuse de nombreuses informations.



En fait, la syntaxe du diagramme d'activité est l'une des plus riches de tous les diagrammes SysML, et lorsque vous ajoutez à cela les mécanismes et outils riches et efficaces qu'Enterprise Architect inclut pour travailler avec ces diagrammes, les opportunités pour un modélisateur de s'exprimer en font l'une des parties les plus polyvalentes mais aussi les plus difficiles de la représentation système.

Le diagramme d'activité SysML est basé sur le diagramme UML du même nom, mais une sémantique supplémentaire a été ajoutée dans deux domaines :

- *Flux continu*, permettant des restrictions sur la vitesse à laquelle les entités circulent le long des bords d'une activité, et des mécanismes pour garantir que les informations les plus récentes sont disponibles pour les actions
- *Probabilité*, introduite dans les activités pour inclure la probabilité qu'une valeur soit disponible pour un bord ou une sortie sur un ensemble de paramètres

Bien que l'on puisse dire que le diagramme est basé sur les mécanismes de sérialisation des verbes (chaînes de verbes connectés entre eux par des noms) dans notre langage naturel, comme mentionné précédemment, il a ses origines formelles dans une branche des mathématiques appelée *réseaux de Petri* et flux de jetons. Il est impératif qu'un modélisateur comprenne l'aspect flux de jetons du langage et puisse apprendre à visualiser ces éléments invisibles qui circulent à travers les flux. Object, sont retenus dans des tampons et sont contrôlés par d'autres mécanismes du langage qui dirigent la manière dont les éléments circulent à partir des actions. Sans cette compréhension, il est difficile d'interpréter un diagramme d'activité, y compris la manière dont la séquence d'actions est contrôlée, comment les entrées sont consommées et comment les sorties sont créées.

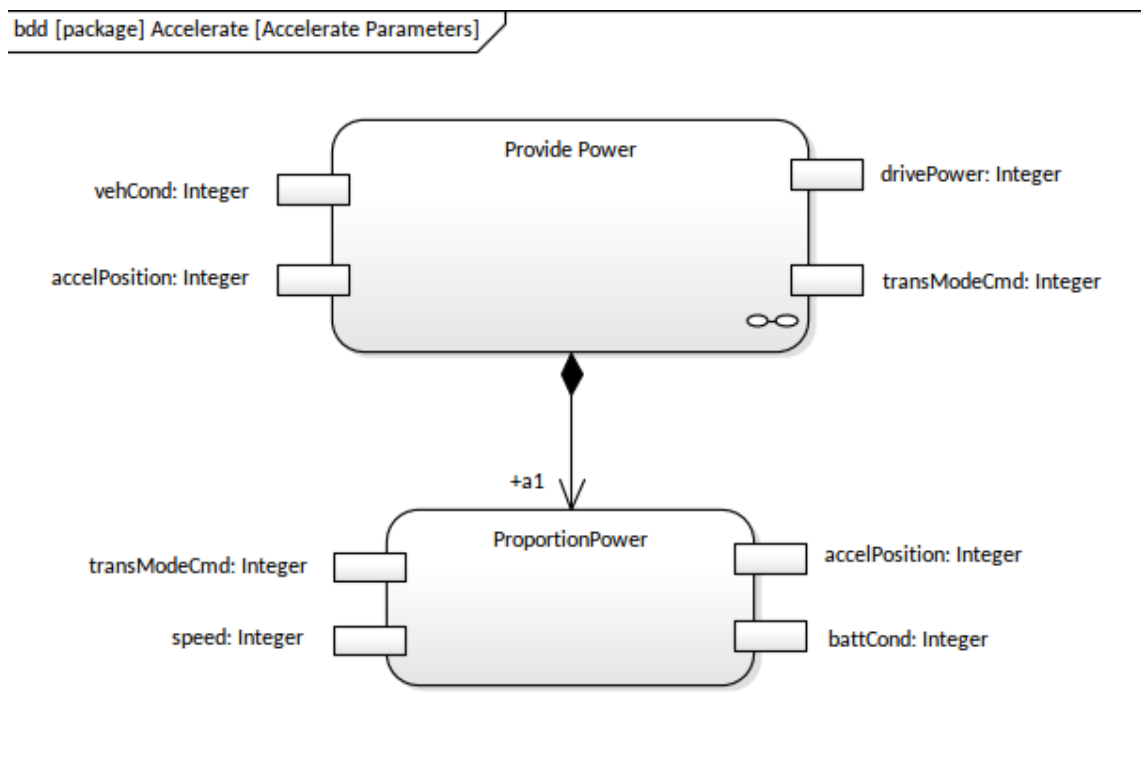
La différence significative entre diagrammes d'activité et leurs proches cousins, tels que les organigrammes ou diagrammes de processus, est la capacité à créer des relations entre ces éléments comportementaux et les éléments structurels.

Un aspect fondamental de la discipline de Ingénierie des Systèmes est la capacité à séparer la fonction de la forme, mais aussi à pouvoir créer une correspondance entre elles qui expose les coutures qui relient ces deux parties intégrantes de architecture et du design. Des preuves empiriques sur des problèmes d'ingénierie de systèmes complexes à grande échelle ont prouvé que cette approche apporte des avantages considérables.

Enterprise Architect fournit une riche boîte à outils pour travailler avec ces relations, y compris la possibilité non seulement d'allouer le comportement du système sous la forme d'activités et d'actions à des blocs, mais également de relier ces éléments à fonctionnalités comportementales appartenant à des blocs, telles que des opérations.

## Actions les éléments fondamentaux du Comportementale

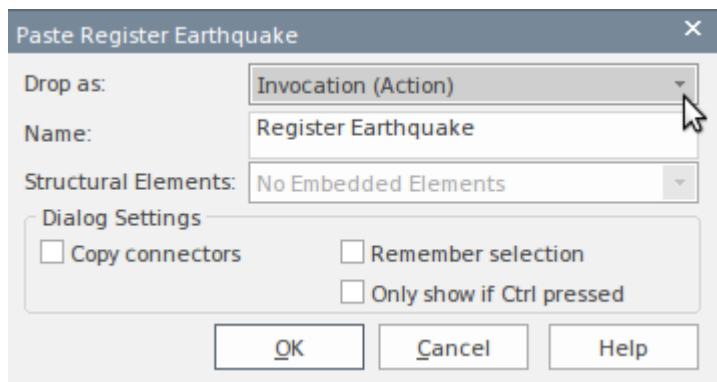
Les actions sont les atomes comportementaux qui sont connectés entre eux pour décrire le comportement d'une activité, d'un sous-système, d'un système ou de l'une de ses parties. En fait, une activité est composée d'un ensemble d'actions qui fonctionnent ensemble pour convertir les éléments (jetons) qui sont entrés dans l'activité en éléments (jetons) qui sont sortis par l'activité. La première Action d'une séquence recevra ses entrées de l'un des nœuds de paramètres d'entrée de l'activité propriétaire et la dernière Action de la séquence placera la sortie sur l'un des nœuds de paramètres de sortie de l'activité. Les actions elles-mêmes ont des périphériques d'entrée et de sortie appelés Pins - une Action recevra des jetons sur ses Pins d'entrée, effectuera son travail et placera les jetons résultants sur ses Pins de sortie.



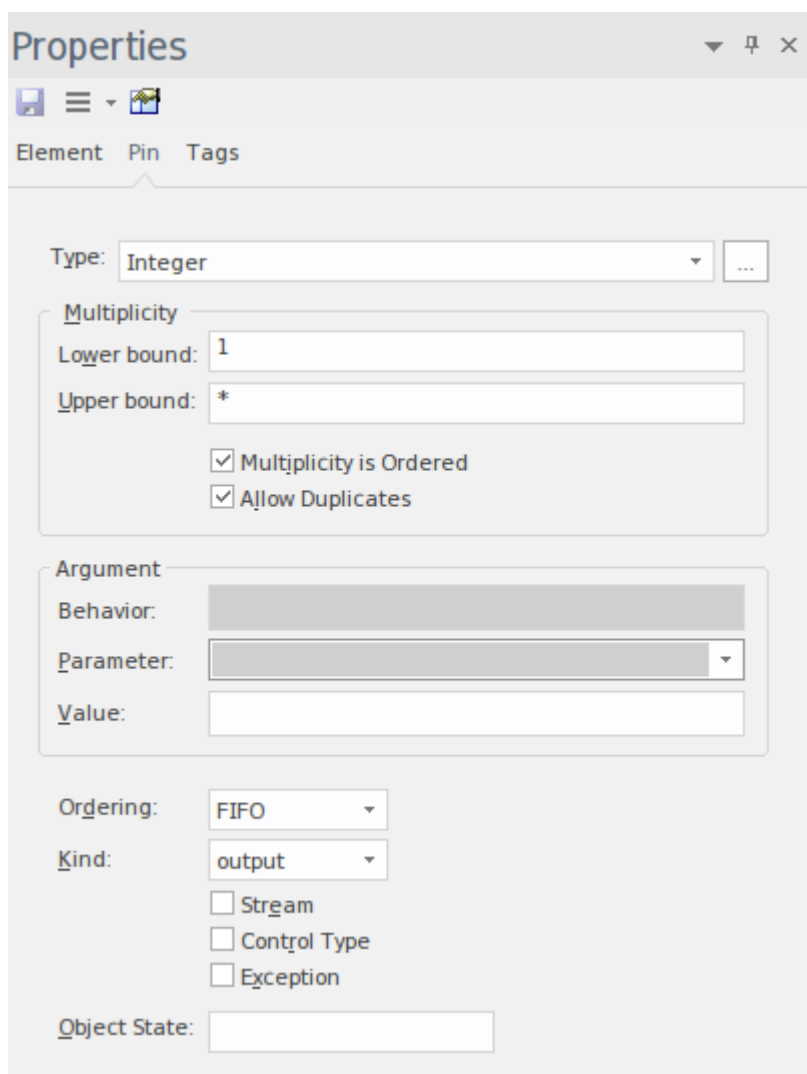
Enterprise Architect offre support complète des actions modélisation et de leurs entrées et sorties, et diverses parties de l'interface utilisateur peuvent être utilisées lors de l'utilisation de ces éléments fondamentaux du comportement. Il existe un certain nombre de types d' Action différents disponibles dans la boîte à outils.

- ▢ Action
- 🔊 Action (call behavior)
- 📧 Action (accept event)
- ⌚ Action (accept event timer)
- 📤 Action (send signal)

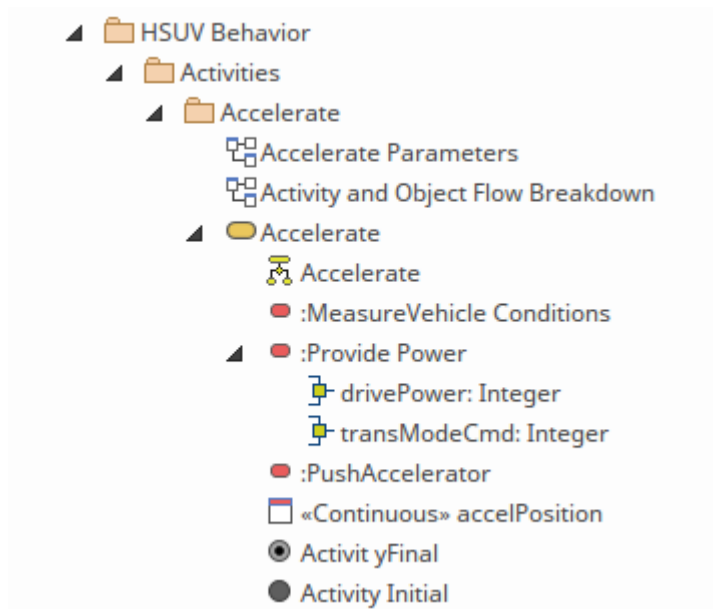
Un ingénieur peut ajouter une Action directement depuis la boîte à outils, mais il est plus courant de créer des actions à partir d'activités existantes qui ont été définies dans des hiérarchies, comme décrit dans la rubrique *Création de hiérarchies d'activités* . Pour ce faire, une activité ne doit pas être glissée depuis la boîte à outils mais depuis la fenêtre Navigateur et déposée sur un diagramme d'activité ouvert en tant qu'invocation - cela a pour effet de créer une Action basée sur l'activité et de la placer dans le diagramme .



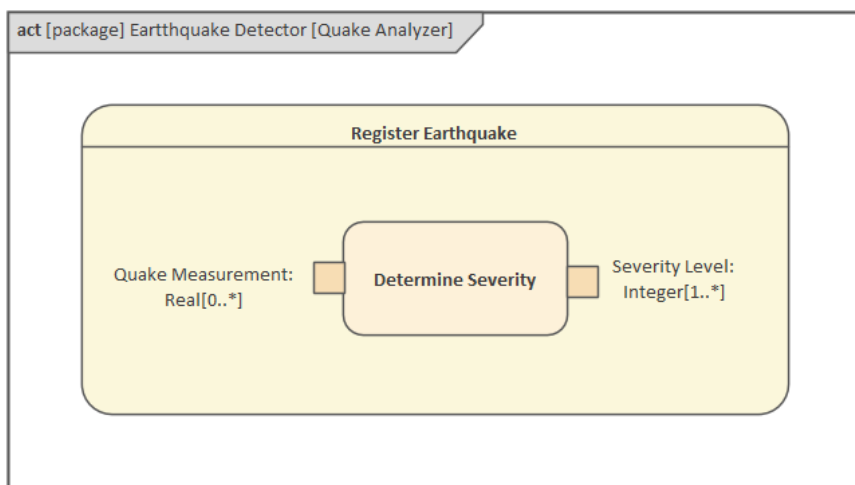
La fenêtre Propriétés intégrée facilite le travail avec les actions et leurs Pins . Un ingénieur peut spécifier une gamme de propriétés pour l' Action et ses Pins , y compris le nom de la broche, Type , la multiplicité, la direction et bien plus encore. La fenêtre Propriétés peut être ancrée ou flottante, et même déplacée vers un autre écran ; lorsque des éléments sont sélectionnés dans la fenêtre Navigateur ou dans un diagramme les propriétés peuvent être visualisées, créées ou modifiées.



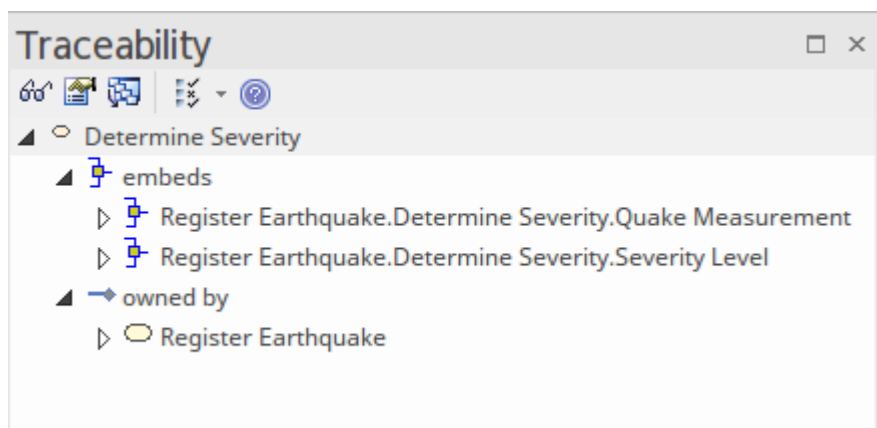
Au fur et à mesure de la création d'un diagramme , les éléments ajoutés sont automatiquement insérés dans la fenêtre Navigateur , ce qui permet à l'ingénieur d'avoir une vue structurée de l'activité, des actions, Pins et des autres éléments. Même si le diagramme n'est pas ouvert, les éléments peuvent être sélectionnés dans la fenêtre Navigateur et édités directement dans la fenêtre Propriétés .



Un certain nombre de ces propriétés seront affichées dans le diagramme de l'étiquette Pin, notamment le nom, le type et la multiplicité. La multiplicité spécifie à la fois une limite inférieure et une limite supérieure. La limite inférieure spécifie, pour une exécution donnée de l'action, le nombre autorisé de jetons que le pin peut consommer ou créer, et la limite supérieure spécifie le nombre maximal de jetons consommés ou créés sur ce pin.



Il existe également une large gamme de fenêtres qui peuvent être utiles lorsque vous travaillez avec les actions, notamment la fenêtre de traçabilité, qui montre comment les éléments sont liés quel que soit leur emplacement dans le référentiel ; elle affiche également leurs fonctionnalités structurales telles que Pins et les paramètres.



# Présentation Diagrammes d'activité

Le diagramme d'activité est un diagramme qui peut être utilisé pour montrer la séquence d'actions qui décrivent le comportement d'un Bloc ou d'un autre élément structurel. Les actions sont séquencées à l'aide de flux de contrôle et peuvent contenir Pins d'entrée et de sortie qui agissent comme des tampons pour les éléments qui circulent d'une Action à une autre (ou à partir de nœuds de contrôle ou de tampon). Le travail effectué par les actions consomme ou produit ces éléments. Les éléments peuvent être des matériaux, de l'énergie ou des informations, selon le système et l'activité décrits.

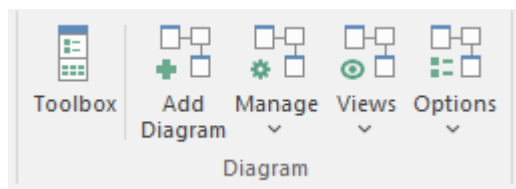
## Créer Diagrammes d'activité

Un diagramme d'activité peut être créé à partir de plusieurs emplacements dans l'Interface Utilisateur tels que :

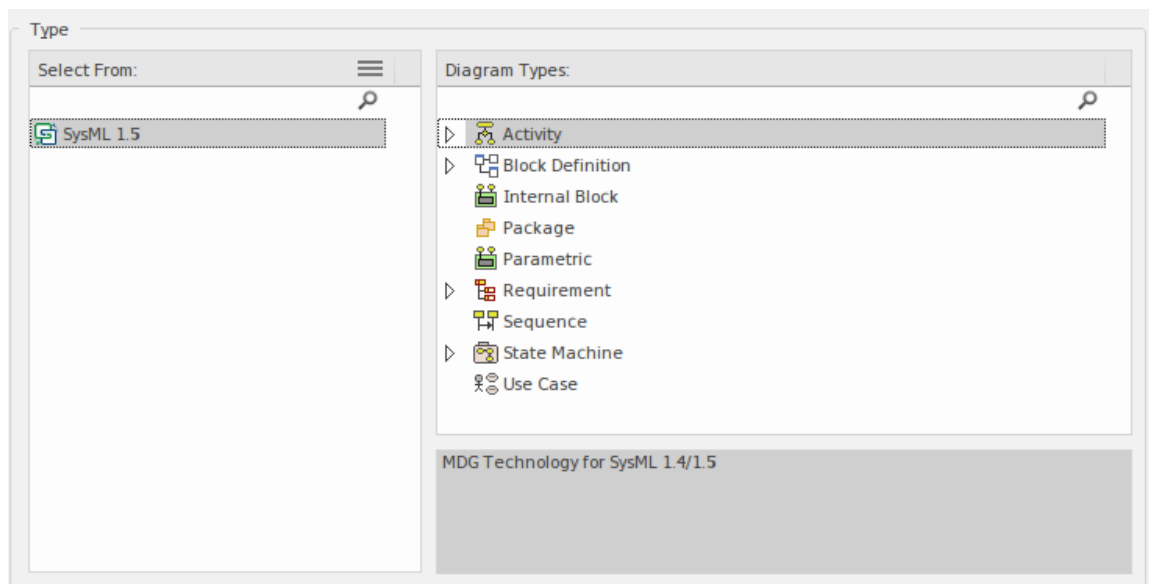
- Ruban de Conception - *Icône 'Ajouter' Diagramme sur le panneau 'Diagramme'*
- Barre d'outils Navigateur - *Nouveau Diagramme icône*
- Navigateur Menu Contexte - *Nouveau Diagramme*

Nous allons utiliser le Ruban de Conception pour créer un diagramme d'Activité. Tout d'abord, sélectionnez l'emplacement dans le Navigateur où vous souhaitez créer le diagramme d'Activité. Comme pour tous diagrammes, il peut s'agir soit d'un Paquetage, soit d'un élément, mais il est courant d'insérer diagrammes d'Activité dans un Paquetage. Une fois l'emplacement Paquetage sélectionné dans le Navigateur, sélectionnez l'option du ruban :

Conception > Diagramme > Ajouter Diagramme



La sélection de cette option ouvre la dialogue *Nouveau Diagramme*, qui vous permet de modifier le nom du diagramme (qui correspond par défaut au nom du Paquetage ou de l'élément contenant le diagramme, tel que sélectionné dans le Navigateur). Avec la perspective SysML choisie et la version de SysML sélectionnée, une liste de diagrammes s'affiche à partir de laquelle vous sélectionnez le diagramme d'activité. Lorsque vous cliquez sur le bouton OK, un nouveau diagramme d'activité est créé à l'emplacement sélectionné dans le Navigateur. Le canevas du diagramme s'ouvre, vous permettant de commencer à ajouter des éléments et des connecteurs qui décrivent la valeur que le système fournira à ses utilisateurs. Enterprise Architect affiche également les pages « Activité » de la boîte à outils, qui contiennent les éléments et les relations définis par la spécification SysML comme applicables à la construction diagrammes d'activité. N'importe quel nombre d'autres pages de la boîte à outils peut être ouvert si nécessaire, en plus des pages de la boîte à outils Éléments communs et Relations communes qui sont toujours disponibles.



Les éléments et connecteurs les plus importants utilisés avec le diagramme d'activité sont :

### Éléments

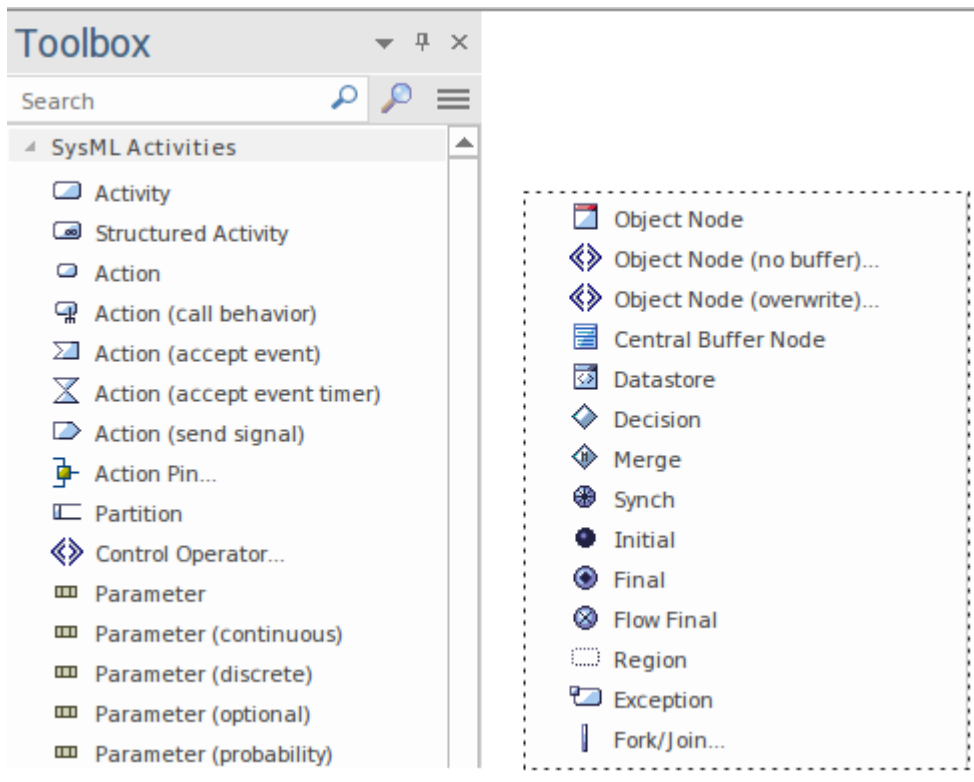
- Activité
- Action
- Action Pin
- Partition
- Paramètre
- Initial
- Final
- Décision
- Fourche et Joindre
- Magasin de données

### Connecteurs

- Contrôle du flux
- Flux Object
- Interrompre le flux

Il existe de nombreux autres éléments et connecteurs qui peuvent être utilisés sur ces diagrammes et qui sont importants pour modélisation plus avancée ; certains d'entre eux peuvent être nécessaires à mesure que les modélisateurs deviennent plus expérimentés ou que des parties plus complexes du comportement d'un système sont décrites ou conçues. Il s'agit notamment des paramètres d'activité, de fusion, du nœud tampon central, des régions, Fourche et Joindre , Décision et de Fusion.





Les éléments peuvent être ajoutés au diagramme en les faisant glisser de la boîte à outils vers le canevas diagramme . Il est recommandé de commencer par un *élément initial* et un ou plusieurs éléments *finaux* , qui sont nommés de manière appropriée pour décrire la manière dont l'activité démarre et les multiples manières possibles dont elle peut se terminer. Laisser le nom de ces éléments vide ou leur donner un nom galvaudé tel que « début » ou « fin » n'aidera pas à faire comprendre au lecteur quel système ou partie d'un système est modélisé et peut conduire à une mauvaise interprétation du diagramme . Une fois ces nœuds ajoutés et placés de manière appropriée dans le diagramme , des actions et des nœuds Object peuvent être ajoutés au diagramme . Les actions peuvent être connectées à l'aide de la relation de flux de contrôle, définissant la séquence dans laquelle les actions seront exécutées.

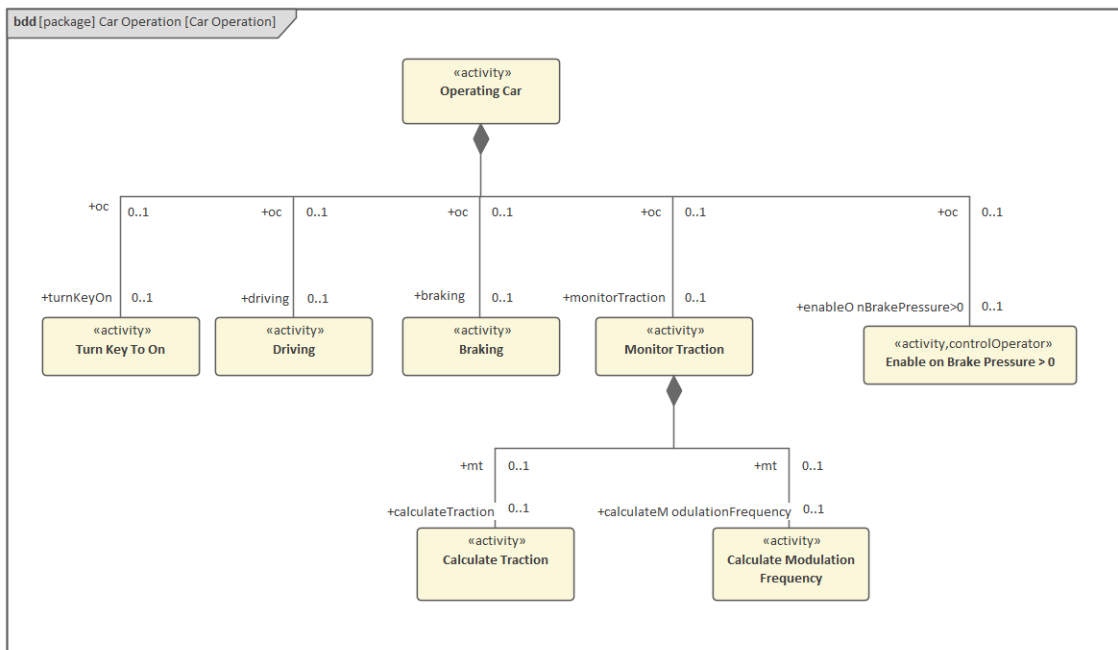
Une fois qu'un diagramme de base a été créé, et à mesure que la connaissance du domaine et des comportements du système s'accroît, il est possible de structurer ou d'embellir le diagramme en utilisant des éléments et des relations supplémentaires, notamment :

- Structures de contrôle pour les flux Object : fourches et Jointures , nœuds Décision et de fusion
- Structures d'entrée et de sortie d'activité : paramètres d'activité (streaming et non-streaming)
- Regroupement d'ensembles d'actions : régions et arêtes interruptibles
- Structures de stockage de jetons : magasins de données et tampons centraux

Comme indiqué précédemment, le diagramme d'activité dispose d'un riche ensemble de dispositifs linguistiques et l'ingénieur est encouragé à utiliser ces dispositifs pour enrichir la description du système, mais une certaine prudence doit être exercée pour garantir que ces mécanismes linguistiques peuvent être compris par le public visé.

## Créer des hiérarchies d'activités

Les nouveaux venus dans le Langage Modélisation des Systèmes et Entreprise Architect pourraient être surpris d'apprendre que ce ne sont pas des Activités mais des Actions qui sont utilisées sur les diagrammes d'Activité. Les Activités qui sont les classificateurs des Actions sont généralement visualisées sur diagrammes de Définition Bloc . Cela peut sembler un peu contre-intuitif, mais lorsque vous comprenez que l' Action est l'atome fondamental du comportement du système, cela prend plus de sens. Les Activités sont des classificateurs et en tant que telles peuvent, comme les éléments Bloc , participer à une large gamme de relations structurales, c'est pourquoi les relations telles que les Associations marquées avec Composition peuvent être utilisées entre les Activités.



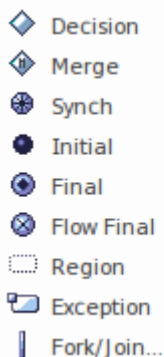
Dans ce diagramme une structure de décomposition a été utilisée où une activité est décomposée en un certain nombre d'activités plus granulaires à l'aide de la relation Composition. Un certain nombre de relations ont été regroupées pour rendre le diagramme plus attrayant, en utilisant l'un des styles de ligne flexibles disponibles dans le menu contextuel diagramme .

# Spécification Séquence Action avec des flux de contrôle

Les actions sont exécutées dans le contexte d'une activité, et l'ordre dans lequel les actions sont exécutées est en grande partie contrôlé par l'utilisation de connecteurs spéciaux appelés flux de contrôle. Ces connecteurs sont des lignes dirigées tracées entre les actions et agissent essentiellement comme un conduit pour les jetons de contrôle - permettant aux jetons de circuler d'une Action à l'autre dans le sens de la flèche. Une Action ne peut pas commencer son travail tant que tous les flux de contrôle entrants n'ont pas reçu un jeton ; une fois qu'ils l'ont reçu et que l' Action est exécutée, un jeton est dit placé sur le flux de contrôle sortant, ce qui implique qu'il se déplacera vers l' Action suivante de la séquence. Les relations de flux de contrôle sont disponibles dans les pages « Activité » de la boîte à outils Diagramme

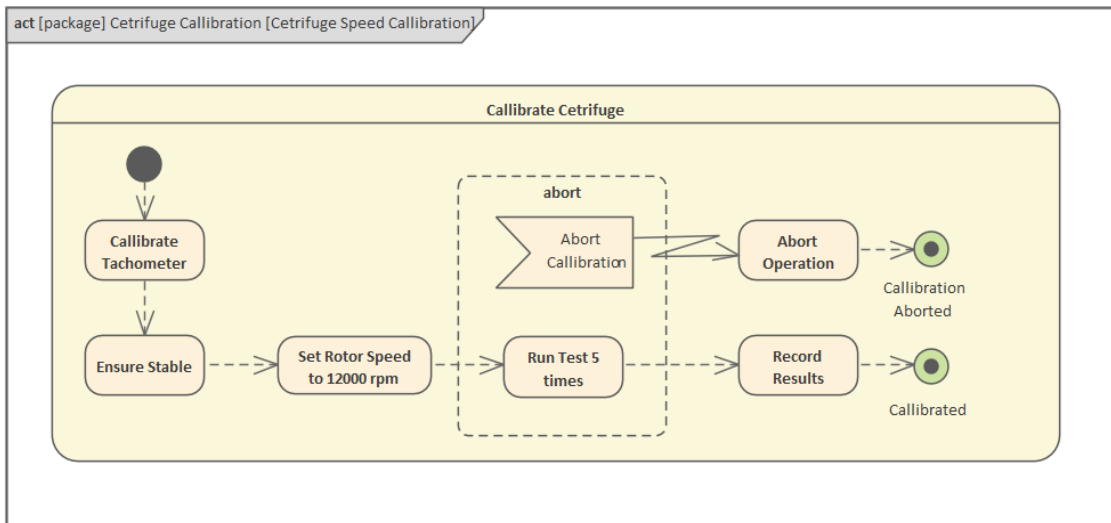


Il existe également des nœuds de contrôle qui peuvent être utilisés avec les flux de contrôle pour orchestrer le fonctionnement des flux avec l'utilisation de fourches, Jointures, de décisions et de fusions. Il existe trois nœuds spécialisés : Initial, Final et Flow Final, qui agissent respectivement comme le début et la fin du flux. Le nœud Final (anciennement Activity Final) est utilisé pour indiquer que lorsqu'un jeton arrive, l'activité entière se termine, tandis que le Flow Final consommera les jetons entrants mais n'aura aucun effet sur l'activité englobante.



## Interruption du flux normal

Il existe un certain nombre de circonstances au cours de l'exécution d'une activité dans lesquelles un modélisateur peut vouloir spécifier un moyen d'arrêter le comportement dans une partie d'une activité. Par exemple, dans un scénario réel, un utilisateur peut utiliser une fonction de machine telle que l'étalonnage d'une centrifugeuse, puis décider de mettre fin à une partie particulière du processus d'étalonnage. Ce scénario peut être fourni par un bouton Annuler sur l'interface. Le SysML permet de modéliser cette situation à l'aide d'une région interruptible et d'un bord d'interruption. La notation permet de dessiner la région interruptible pour inclure un certain nombre d'éléments tels que des actions et d'autres nœuds. En règle générale, lorsqu'un événement inhabituel se produit, un événement est déclenché dans l'activité et reçu par une Action de signal d'acceptation ; cet élément n'a aucun flux de contrôle entrant et un seul bord d'interruption sortant, qui cible une Action qui réside en dehors de la région.











Dans ce diagramme , un ingénieur a modélisé le processus utilisé pour effectuer un étalonnage de vitesse pour une centrifugeuse. Le processus d'étalonnage de la centrifugeuse peut être interrompu pour diverses raisons ; par exemple, si la centrifugeuse est devenue instable ou si l'opérateur est appelé pour effectuer d'autres tâches. Une Action d'événement d'acceptation est utilisée pour montrer que l'activité dispose d'un mécanisme pour écouter une interruption requise dans une région spécifiée de l'activité. Le connecteur spécial de flux d'interruption cible ensuite une Action en dehors de la région, qui est utilisée pour arrêter la centrifugeuse ; enfin, cela se poursuit jusqu'à ce que l'activité d'étalonnage soit terminée.





## Spécification du flux Item avec des flux Object

Les activités et les actions qui les composent fonctionnent généralement en traitant les éléments qui arrivent sur les nœuds d'entrée et, une fois le travail terminé, en plaçant les éléments résultants sur les nœuds de sortie. Comme nous l'avons vu précédemment, modélisation des activités dans SysML est basée sur une branche des mathématiques appelée réseaux de Petri, qui s'intéresse aux systèmes d'événements State discrets. Les éléments qui arrivent aux structures d'entrée doivent traverser le graphique des activités et leurs actions contenues de manière ordonnée et systématique. Le passage est créé par des flux Object qui agissent comme des conduits pour transporter les jetons d'un nœud à un autre. Les jetons représentent un certain nombre de types différents de « choses », notamment des informations, des structures ou des éléments physiques tels que des solides, des liquides et des gaz. Il y a donc deux parties importantes dans la façon dont les éléments traversent l'activité : les nœuds qui agissent comme origines et destinations des jetons, et les connecteurs (conduits) qui transmettent les éléments.

Enterprise Architect offre support complet pour modélisation de ces flux, et lorsqu'un diagramme est créé ou ouvert pour modification, la boîte à outils contient les nœuds Object comme indiqué :

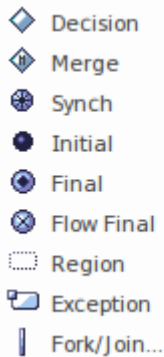
-  Parameter
-  Parameter (continuous)
-  Parameter (discrete)
-  Parameter (optional)
-  Parameter (probability)
-  Object Node
-  Object Node (no buffer)...
-  Object Node (overwrite)...

Il contient également une section qui répertorie les relations de flux Object qui peuvent être utilisées pour connecter les nœuds, créant ainsi le conduit permettant aux jetons de circuler d'un nœud à un autre.

-  Object Flow
-  Object Flow (Continuous)
-  Object Flow (Discrete)
-  Object Flow (Probability)

## Orchestrer le flux de jetons

Lors de modélisation de systèmes complexes, il est souvent nécessaire de créer des chemins plus élaborés (conduits) pour le flux de jetons, tels que des chemins de bifurcation et de jonction pour permettre l'envoi de jetons à un certain nombre de nœuds object afin que le travail puisse être effectué simultanément, ou pour permettre l'acheminement des jetons vers un chemin particulier en fonction de certaines conditions. Ces nœuds de contrôle contrôlent le flux et sont regroupés sur une page de la boîte à outils Diagramme .



Enterprise Architect permet de manipuler les connecteurs pour créer n'importe quel chemin requis. Cela peut être réalisé en utilisant les styles de ligne du menu contextuel d'un connecteur ; le plus flexible d'entre eux est le style de ligne personnalisé, mais il existe plusieurs autres styles très utiles. Un modélisateur peut également fixer les extrémités du connecteur à une partie spécifique de l'élément source ou cible.

## Stockage des jetons en transit

Lors de l'exécution d'une activité, il est parfois nécessaire de stocker des jetons pendant une période plus longue que ce qui est possible avec les paramètres d'activité et Pins Action, qui agissent simplement comme des périphériques de stockage temporaires. Une circonstance courante est celle où un certain nombre d'actions nécessitent l'accès à un flux de jetons. Les jetons peuvent être stockés dans un tampon central et mis à la disposition des nœuds qui en ont besoin. Le tampon central accepte tous les jetons sur ses flux entrants, puis les met à la disposition des nœuds en aval ; une fois acceptés, les jetons sont ensuite supprimés du tampon.

Le tampon central peut être créé en faisant glisser l'icône « Tampon central » de la boîte à outils sur un diagramme d'activité ouvert ; il peut ensuite être connecté à d'autres nœuds object à l'aide de flux Object .

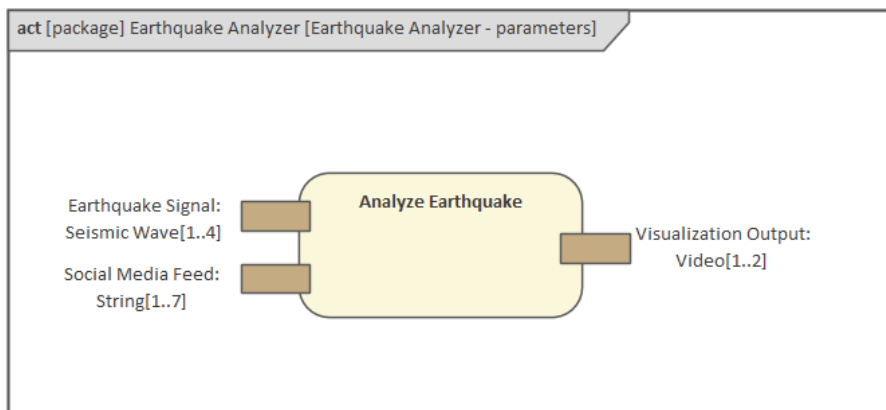


Ainsi, le tampon central peut, pendant l'exécution de l'activité, être rempli de jetons ou vide en fonction de la consommation de jetons. Un autre type de nœud est le magasin de données, une spécialisation du tampon central où, lorsque les jetons sont consommés par les actions en aval, une copie est effectuée et stockée dans le tampon. Cela a pour effet de donner au magasin de données l'apparence d'un magasin permanent - mais uniquement pendant la durée de vie de l'exécution de l'activité.

Le magasin de données peut être créé en faisant glisser l'icône « Magasin de données » de la boîte à outils sur un diagramme d'activité ouvert ; il peut ensuite être connecté à d'autres nœuds object à l'aide de flux Object .

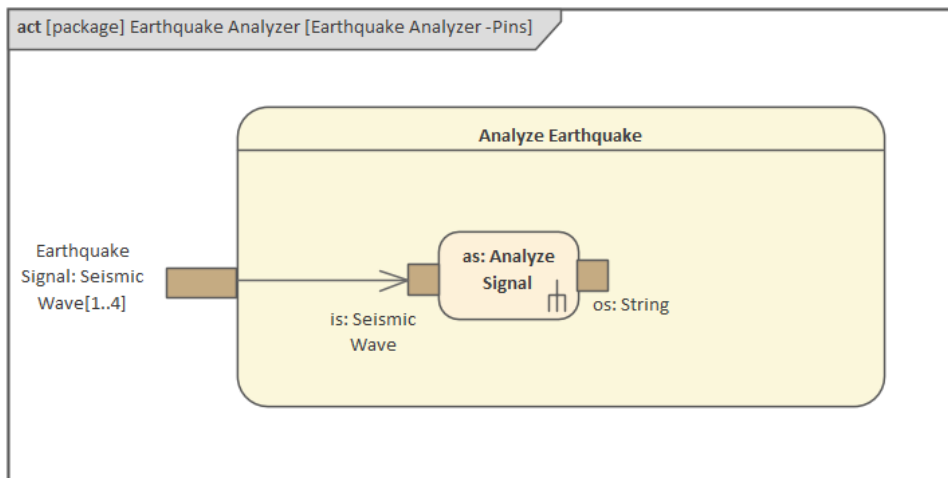
## Entrées et sorties Modélisation avec paramètres et Pins

Les activités et leurs actions constitutives constituent la force de travail des systèmes. Tandis que les éléments structurels tels que les blocs et les parties définissent la structure ou l'anatomie d'un système, les activités définissent la physiologie. Lorsqu'une activité est en cours d'exécution, nous voyons les éléments structurels être appelés à agir pour accomplir un certain type de comportement du système. Une grande partie du travail qu'un système effectue et les comportements qui définissent ce travail dépendent des entrées du système que l'activité en cours d'exécution consomme afin de produire des sorties.



Les entrées et les sorties varient considérablement d'un système à l'autre et peuvent inclure des éléments tels que des signaux de contrôle, des matériaux, de la lumière, des fluides, de l'énergie, des nombres et des informations. Les entrées et les sorties sont appelées paramètres, qui peuvent être typés et peuvent avoir des multiplicités. Le typage garantit que l'activité spécifie le type de « chose » qu'elle attend. Ainsi, si un distillateur avait un paramètre d'entrée avec un type de liquide défini ou, plus précisément encore, un contaminant liquide, alors l'activité serait mal formée si elle recevait un gaz ou une valeur Integer comme entrée sur ce paramètre. Les types peuvent être n'importe lequel d'un ensemble défini allant d'un simple Integer à une structure composée. Les entrées et les sorties peuvent être typées par un Bloc, de sorte que vous ayez un élément structurel bien défini - par exemple, un article d'épicerie qui passe par un système d'auto-scanner à la caisse d'un supermarché. Il existe toute une gamme d'autres propriétés qui peuvent être définies pour un paramètre, notamment Streaming ou Non Streaming, Multiplicités et Direction. Le streaming est utilisé lorsqu'il y a un flux continu dans le paramètre, comme avec un fluide, ou un signal de communication ou d'information tel qu'un flux audio ou visuel. Les multiplicités définissent les limites supérieure et inférieure du nombre de jetons consommés par un paramètre d'entrée ou produits par un paramètre de sortie. Tandis que la direction définit si le paramètre reçoit une entrée (in) ou produit une sortie (out) ou une combinaison des deux (inout).

Lorsque des activités sont placées sur un diagramme d'activité en tant qu'invocations, elles sont représentées par des actions et tous les paramètres appartenant à une activité seront modélisés sous forme de Pins sur ces actions. Les Pins reçoivent des jetons sur les flux Object entrants et l'Action propriétaire exécute son travail et place un nombre spécifié de jetons sur les Pins de sortie. Les Pins peuvent avoir un type simple tel qu'un Integer, une structure complexe telle qu'une matrice ou même un Bloc tel qu'un flux vidéo. Les multiplicités spécifient une limite inférieure et supérieure qui définissent le nombre minimum et maximum de jetons autorisés à arriver et à partir d'une broche donnée. Ce diagramme inachevé montre une Action avec une broche d'entrée et de sortie et la transmission des jetons à partir du paramètre d'entrée de l'activité propriétaire le long du flux Object.



Les paramètres et Pins sont collectivement connus sous le nom de points d'interaction, ce qui signifie qu'ils sont des emplacements où un élément interagit avec son environnement ; ils peuvent être sélectionnés pour être inclus dans un diagramme en utilisant la fenêtre Fonctionnalités polyvalente.

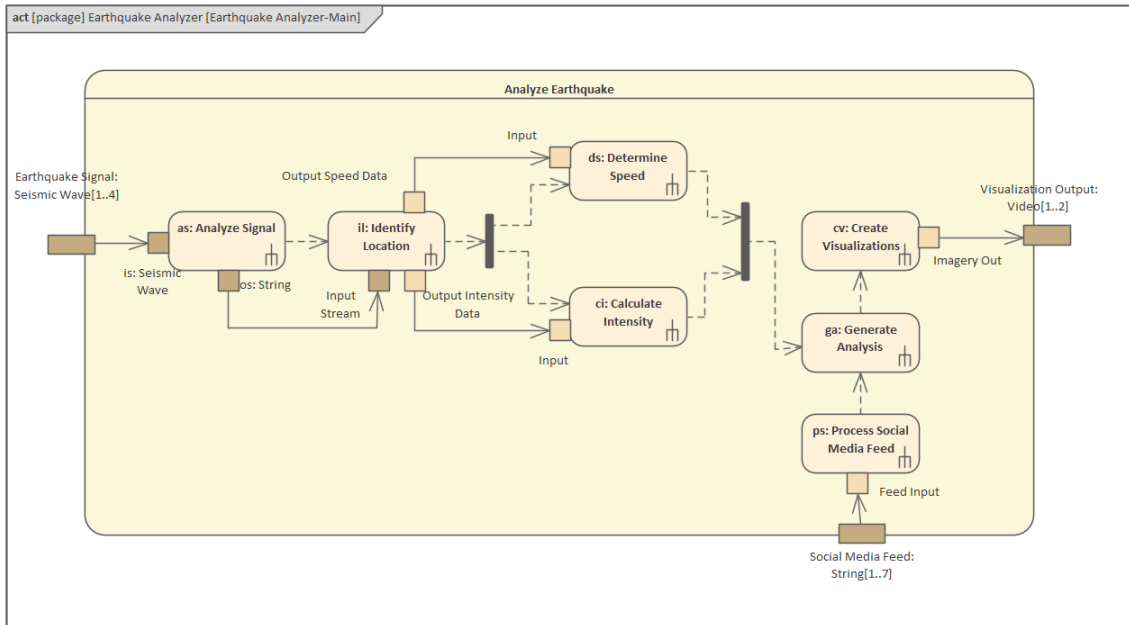
**Features**

Attributes Operations Receptions Parts / Properties Interaction Points

Name	Element	Type	Stereotype	Visible	Owner
<input checked="" type="checkbox"/> Social Media Feed	ActivityParameter			True	Analyze Earthquake
<input checked="" type="checkbox"/> Earthquake Signal	ActivityParameter			True	Analyze Earthquake
<input checked="" type="checkbox"/> Visualization Output	ActivityParameter			True	Analyze Earthquake

Enterprise Architect vous permet de créer un diagramme qui montre l'activité propriétaire comme un conteneur pour les autres activités incluses dans le diagramme en tant qu'actions. Dans ce diagramme , les paramètres d'activité définis sur les activités propriétaires sont exprimés sous forme Pins sur les limites des actions qui ont été incluses en tant qu'invocations des activités. Le diagramme montre une activité avec deux paramètres d'entrée et un seul paramètre de sortie. Les entrées sous forme de jetons peuvent être tracées dans le diagramme lorsqu'elles arrivent aux Pins . Une fois que l' Action a terminé son travail, les jetons sont placés sur les Pins de sortie. Les flux de contrôle montrent le séquençage des actions incluses. Notez qu'une Fourche et Joindre sont utilisées pour montrer que deux actions peuvent être exécutées en parallèle. Notez également qu'un certain nombre de Pins ont été définies comme un flux, ce qui est indiqué sur le diagramme par la couleur unie de la broche.





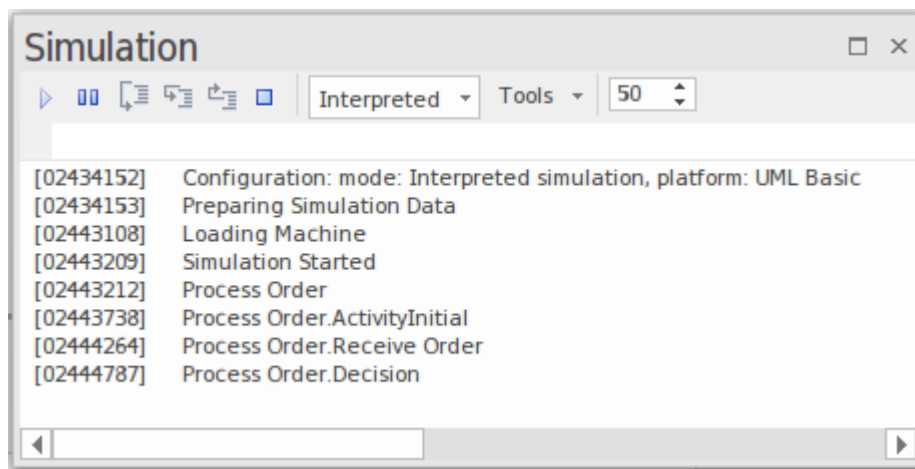
## Visualisation des activités avec des simulations

N'importe lequel des diagrammes d'activité SysML de vos modèles peut être simulé à l'aide du simulateur de modèle dynamique intégré. Cela permet de visualiser les diagrammes de manière convaincante et est utile pour effectuer des démonstrations ou des visites guidées avec l'utilisateur et d'autres membres de la communauté d'ingénieurs.

En utilisant le simulateur Modèle , vous pouvez simuler l'exécution de modèles conceptuels contenant un comportement. Lorsque vous démarrez une Simulation , le modèle actuel Paquetage est analysé et un processus Simulation dynamique est généré pour exécuter le modèle. Comme le simulateur analyse et travaille directement avec les constructions UML , il n'est pas nécessaire de générer du code intermédiaire ou de compiler des « exécutables » de simulation. Il en résulte un environnement de simulation très rapide et dynamique dans lequel des modifications peuvent être apportées et testées très rapidement.

### Fenêtre Simulation

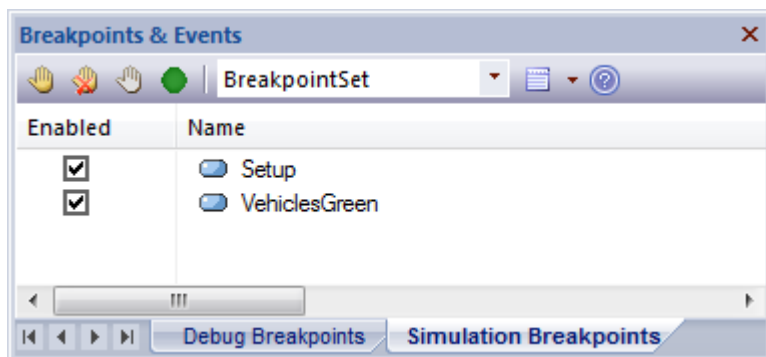
La fenêtre Simulation fournit l'interface principale pour démarrer, arrêter et parcourir votre Simulation . Pendant l'exécution, elle affiche les résultats relatifs à l'étape en cours d'exécution et d'autres informations importantes. Consultez la rubrique d'aide [Run Model Simulation](#) pour plus d'informations sur les commandes de la barre d'outils.



Note la zone de saisie de texte juste en dessous de la barre d'outils. Il s'agit de la zone de saisie de la console. Vous pouvez y saisir des commandes JavaScript simples telles que : `this.count = 4;` pour modifier dynamiquement une variable Simulation nommée « count » à 4. De cette façon, vous pouvez influencer dynamiquement la simulation au moment de l'exécution.

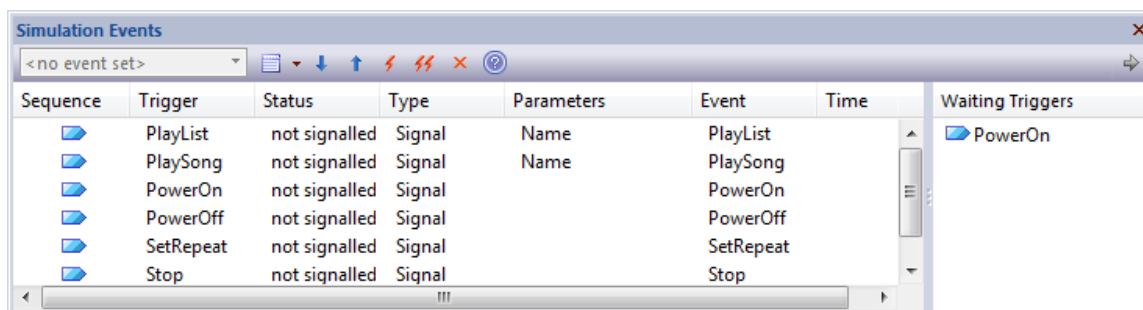
### Fenêtre Points d'Arrêt et Événements

Le processus Simulation utilise également l'onglet « Points d'Arrêt Simulation » de la fenêtre Points d'Arrêt et Marqueurs ('Simuler > Simulation Dynamique > Points d'Arrêt '). Ici, vous définissez des points d'arrêt d'exécution sur des éléments et des messages spécifiques dans une Simulation . Voir les [Simulation Breakpoints](#) Aide sujet pour plus de détails.



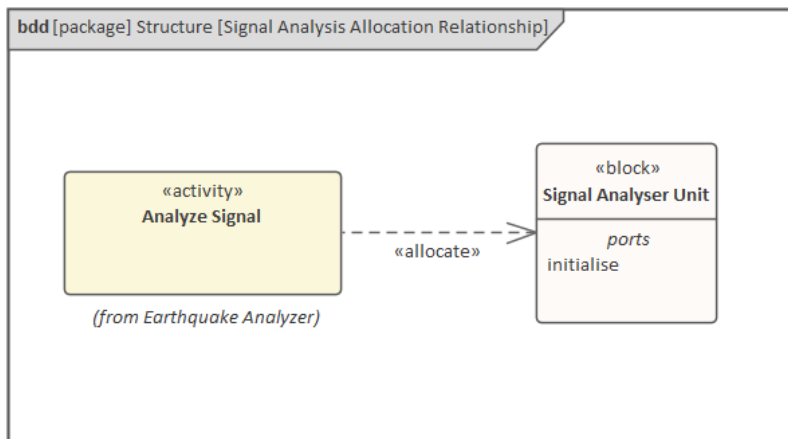
## Fenêtre Simulation Événements

La fenêtre Simulation Événements ('Simulate > Simulation Dynamique > Événements ') fournit des outils pour gérer et exécuter déclencheurs . Déclencheurs sont utilisés pour contrôler l'exécution des transitions Statemachine .

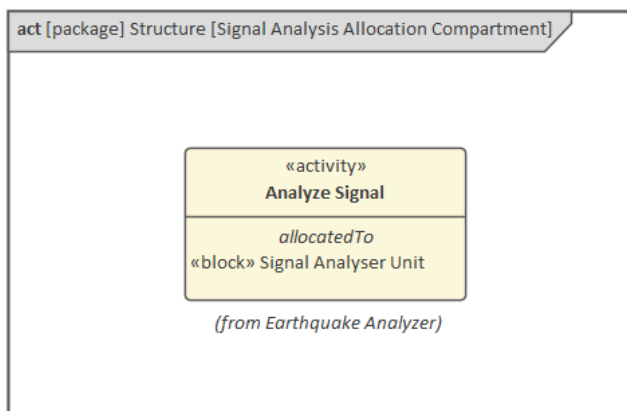


## Affectations et autres Relations

Un aspect fondamental ou une discipline de Ingénierie des Systèmes consiste à relier les éléments structurels aux éléments comportementaux. Ces deux aspects importants d'un modèle s'entremêlent au fur et à mesure du développement du modèle, mais lorsqu'un certain nombre d'ingénieurs commencent à travailler à la définition du système, il est souvent difficile de dire exactement comment le comportement et la structure seront liés. La relation d'allocation est particulièrement utile dans ces situations. Elle peut être utilisée comme un moyen de montrer la relation entre les éléments comportementaux et les éléments structurels qui éclaireront la modélisation plus rigoureuse qui sera utilisée à mesure que les notions décrites dans le modèle deviennent plus précises.



Enterprise Architect prend également supporte un certain nombre d'autres manières de représenter la relation d'allocation, notamment en tant que compartiment dans l'élément de comportement ou de structure.



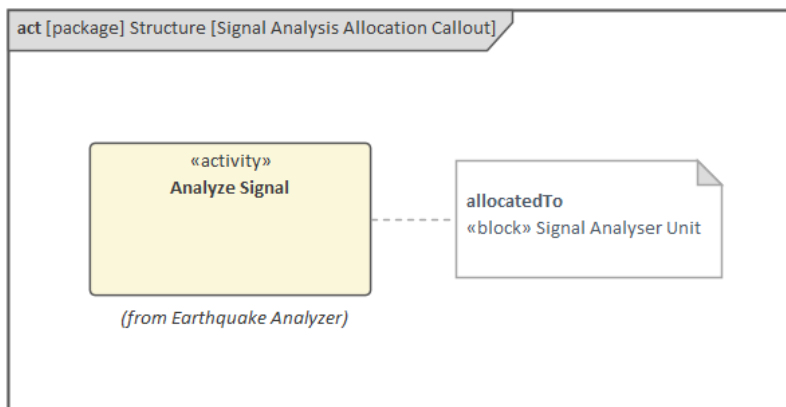
Tout élément qui autorise les compartiments peut être configuré pour masquer ou afficher n'importe quel nombre de compartiments disponibles. La liste des compartiments est spécifique à un élément et est dynamique, ce qui signifie qu'un compartiment ne sera visible que si l'élément participe à une ou plusieurs relations du type spécifié et que la relation n'est pas visible sur le diagramme qui le contient. Les mêmes options peuvent être utilisées pour afficher une gamme d'autres aspects structurels ou sémantiques des éléments, comme indiqué ici.



Il est également possible d'afficher la relation dans une notation de légende, où une note est connectée à l'élément et affiche le nom de la relation et les détails de l'élément associé. Ce diagramme montre la notation d'une activité, montrant le Bloc auquel elle a été alloué . Pour y parvenir, un ingénieur doit :

1. Assurez-vous que la relation est affichée dans le diagramme .
2. Sélectionnez la relation et affichez le menu contextuel.
3. Choisissez l'option « Créer Note liée ».

Cette notation de légende peut être utilisée avec n'importe quel type d'élément ou de relation SysML et constitue un moyen utile d'afficher la relation pour certains types d'audiences.

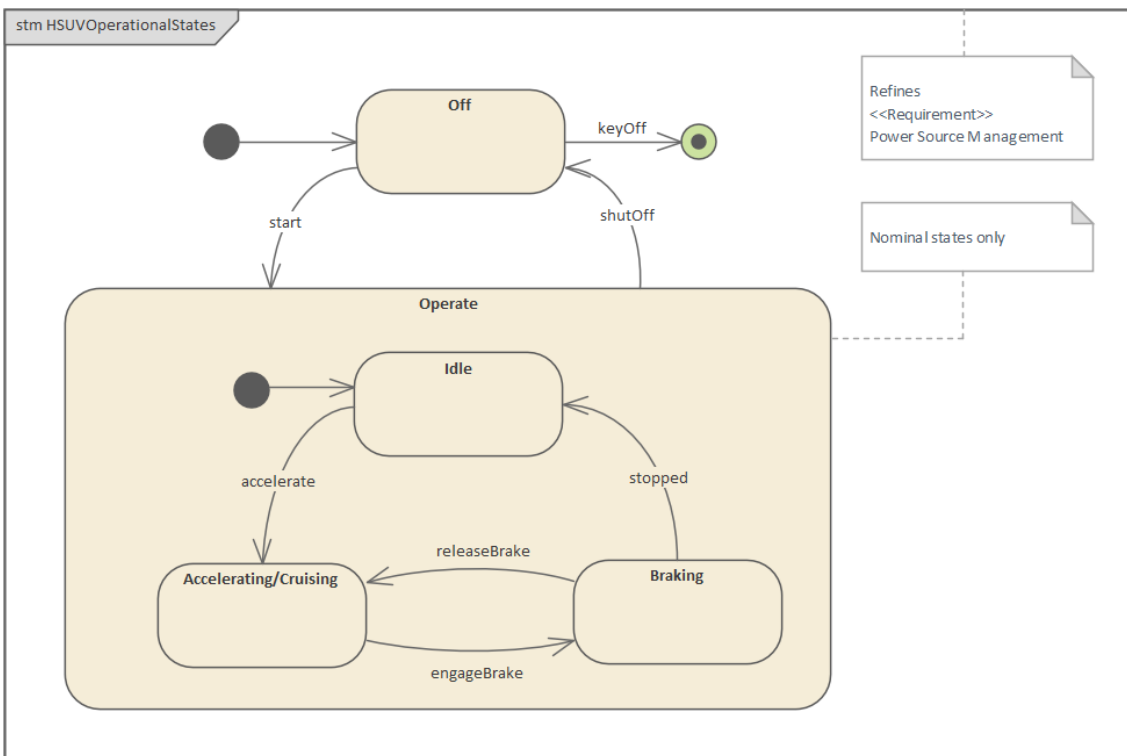


# Modélisation du changement avec Statemachines

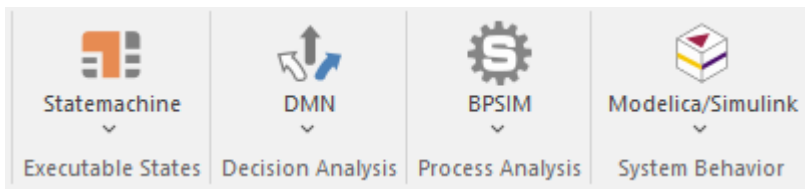
Notre monde est en constante évolution, car les « choses » changent ou évoluent, passant d'un état à un autre. L'eau gèle, les glaciers se déforment et coulent, la glace fond, les feux de circulation passent du vert à l'orange et au rouge, les avions décollent, montent, volent, descendent et atterrissent. La Statemachine SysML est utilisée pour décrire comment une structure, sous forme de blocs, change d'état dans un cycle de vie limité dans le temps. Notre préoccupation ne concerne pas la structure de l'instance Bloc mais son comportement, qui peut à son tour avoir un impact sur sa structure. Nous ne nous intéressons pas à chaque état dans lequel une « chose » peut se trouver, mais plutôt aux états significatifs. Ainsi, les états importants pour les molécules d'eau peuvent être un *solide*, un *liquide* ou un *gaz*, mais nous ne nous intéressons généralement pas à l'eau liquide à une température de 67 degrés centigrades. Si nous regardions une bobine de film sur la durée de vie d'un objet, une Statemachine sélectionnerait les images significatives où des changements importants et pertinents se sont produits.

Déterminer ce qui est pertinent est la prérogative et le privilège de l'ingénieur modélisation, et le même Bloc pourrait avoir n'importe quel nombre de Statemachines définies par le même ingénieur ou par des ingénieurs différents. L'état d'un avion pourrait être modélisé du point de vue de l'embarquement et du débarquement des passagers, du point de vue de son calendrier de maintenance, du point de vue de la portance ou de tout autre perspectives.

Ce diagramme Statemachine décrit les états de fonctionnement d'un véhicule à moteur SUV. Le diagramme utilise States composites, qui imbriquent des States dans d'autres States. Il existe trois States de haut niveau : Arrêt, Fonctionnement et l' State final sans nom. L' State Fonctionnement comporte un certain nombre de sous-états, à savoir Ralenti, Accélération/Croisière et Freinage. Avec les transitions, cela décrit les états du véhicule lorsqu'il Démarre, Accélère, Freine, S'arrête et enfin lorsque le contact est coupé.

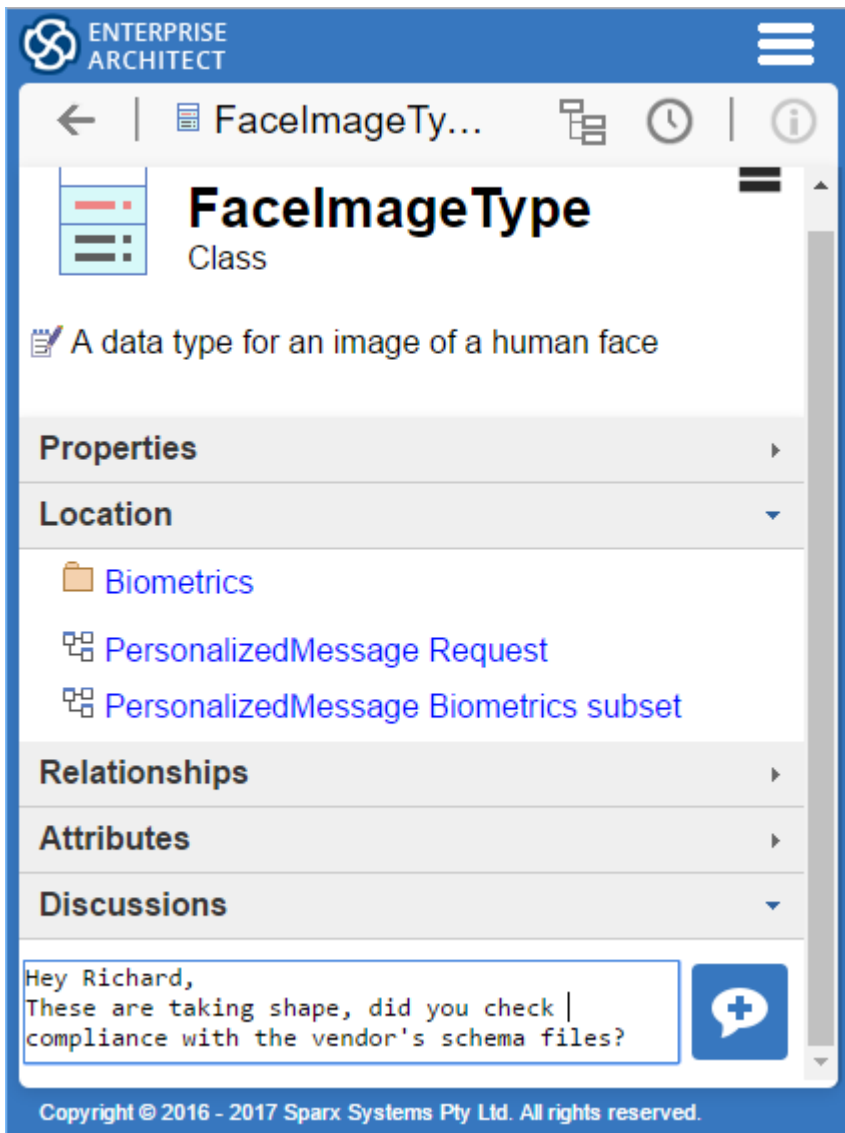


En utilisant Enterprise Architect un ingénieur peut créer Statemachines et définir les transitions d'un état à un autre, y compris Événements qui déclencheur un changement d'état et les actions qui sont déclenchées. En plus de ces représentations modélisation standard, l'outil dispose d'une gamme de fonctionnalités qui peuvent aider à visualiser et à raisonner sur cet important mécanisme linguistique qui lie la structure et le comportement. L'une de ces facilités - que nous examinerons dans cette rubrique - est Statemachines Exécutables, disponible à partir du ruban Simuler.



Statemachines peuvent être définies à n'importe quel niveau de granularité car elles sont l'expression du comportement d'un Bloc . De nombreux nouveaux venus dans SysML sont confus sur ce point. Comme un Bloc peut représenter quelque chose de très simple - comme un interrupteur sur un panneau de commande de sous-marin - ou quelque chose de complexe comme le sous-marin lui-même, une Statemachine peut également représenter les états du commutateur et du sous-marin. Les deux modèles Statemachine peuvent avoir la même complexité, même si les éléments modélisés se situent clairement à l'une ou l'autre extrémité du spectre en matière de complexité.

diagrammes Statemachine peuvent paraître assez simplistes pour le modélisateur inexpérimenté, mais ils sont des outils très efficaces pour la description et l'analyse de problèmes complexes qui ne peuvent être résolus par d'autres moyens. Il faut une approche et un état d'esprit différents, et souvent le cœur du problème est centré sur la sélection du niveau de Bloc , son contexte et la perspective de la Statemachine , plutôt que sur les détails du diagramme . Souvent, les meilleurs résultats sont obtenus de manière heuristique par un certain nombre d'ingénieurs travaillant ensemble. Cela peut être accompli en utilisant fonctionnalités de collaboration d' Enterprise Architect , permettant aux ingénieurs dispersés géographiquement de communiquer au sein du modèle, soit par courrier électronique, discussions, chats et révisions formelles via le client de bureau, soit dans un Navigateur sur un smartphone, une tablette ou un ordinateur portable.



La Statemachine trouve son origine dans les comportements discrets pilotés par événements, utilisant une Statemachine

finie basée sur une variante orientée objet du formalisme StateCharts de David Harel.



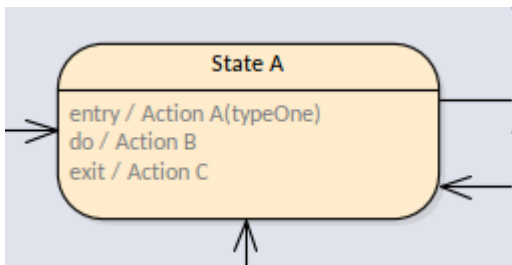
## States et comportements

Un State est créé dans le contexte d'une Statemachine et est utilisé pour modéliser la condition significative définie par l'ingénieur du Bloc propriétaire. Il est important de se rappeler que la Statemachine décrit la durée de vie du Bloc à partir d'une perspective particulière, et que les States doivent être définis à partir de ce point de vue - pas tous States, mais les états significatifs tels que Marche/Arrêt, Ouvert/Fermé, Vert/Orange/Rouge ou Glace/Eau/Vapeur. Formellement, un State modélise une situation dans l'exécution d'un comportement Statemachine où une condition invariante est valable pendant une durée particulière.

Un Bloc passe généralement un certain temps dans un State donné, qui peut durer des nanosecondes ou des jours selon le contexte ; cet aspect temporel n'est généralement pas codifié dans les modèles mais peut être défini dans une simulation. Il existe trois comportements (appelés Actions) qui peuvent être définis par rapport à un State donné :

- *Entrée* - Déclenché lorsqu'un State est entré
- *Do* - Déclenché après le comportement d'entrée et avant le comportement de sortie
- *Sortie* - Licencié avant la sortie de l' State

Ce diagramme montre comment ces States sont représentés dans un diagramme Statemachine. Enterprise Architect peut afficher ces compartiments et d'autres de manière conditionnelle au niveau d'un élément individuel ou collectivement pour tous les éléments du diagramme.



Il est également important de noter que le nœud final est formellement aussi un State, mais il n'a pas la même sémantique de comportement que les States représentés sur diagrammes sous forme de rectangles aux coins arrondis.

Il existe trois types fondamentaux d' State, chacun étant important pour modélisation d'une classe différente de problèmes :

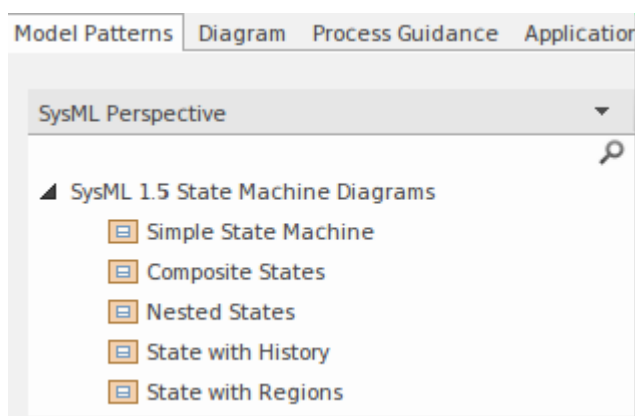
- *State simple* - ne contient pas States internes
- *State composite* - contient au moins une région qui possède States
- *State de sous-machine* - représente une Statemachine entière imbriquée dans l' State propriétaire

Enterprise Architect vous permet de modéliser chacun de ces types State, et un modélisateur peut les utiliser de manière productive dans diagrammes Statemachine pour exprimer des problèmes et des solutions d'ingénierie du monde réel.

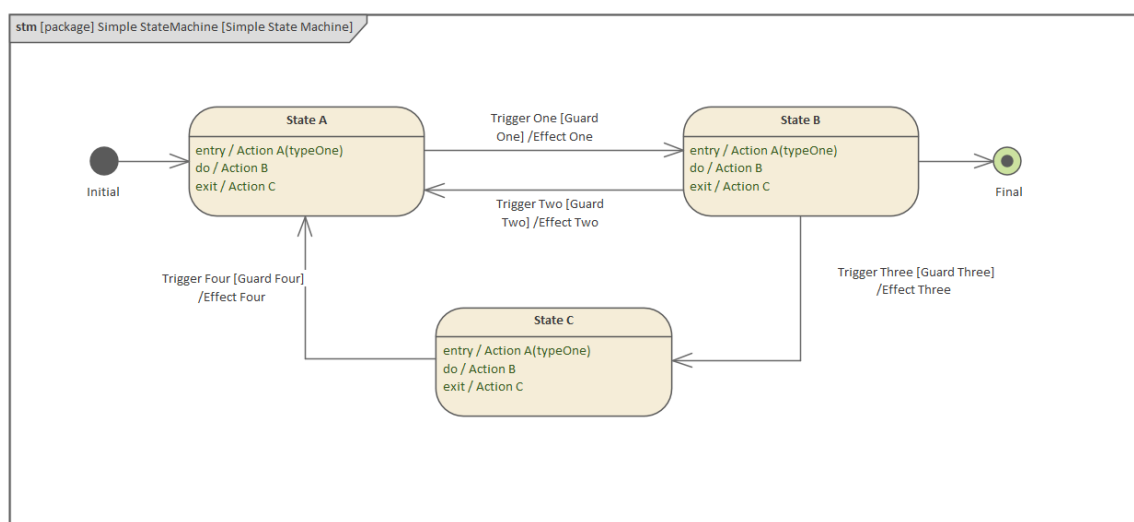
## Présentation Diagrammes Statemachine

Le diagramme Statemachine est l'un des neuf types diagramme SysML de base. Il est utilisé pour créer et visualiser le comportement des blocs lorsqu'ils changent d'état. Les éléments clés du diagramme sont States , les Transitions et les Pseudo-états. Les States représentent le moment significatif de la vie du Bloc d'un point de vue particulier, les Transitions représentent le mouvement d'un State à un autre et les Pseudo-états, comme nous le verrons, agissent comme des contrôleurs de trafic qui influencent la façon dont fonctionnent les transitions.

Enterprise Architect permet à un modélisateur de créer un nombre quelconque de diagrammes Statemachine , et chaque diagramme peut avoir un nombre quelconque d' States , de transitions et de pseudo-états. Chacun de ces éléments et connecteurs diagramme peut à son tour être complété par d'autres informations qui embelliront les diagrammes avec plus de détails. L'application dispose d'une bibliothèque de motif un outil de productivité très utile pour les nouveaux venus et également apprécié des modélisateurs expérimentés. Cette capture d'écran montre la liste des motifs de modèle qui peuvent être utilisés pour créer diagrammes Statemachine .



Le motif peut être utilisé pour créer un certain nombre de Statemachines différentes ; dans cet exemple, nous créons un diagramme simple (région unique) qui contient tous les détails appropriés ajoutés aux States et aux transitions. Un modélisateur peut créer ce diagramme à l'emplacement approprié dans le référentiel, puis modifier les States et les transitions et diagramme pour l'adapter à son propre contexte modélisation . Le diagramme Statemachine initial créé à partir du motif ressemblera à ceci :



Cette fonctionnalité pratique prompts les ingénieurs à compléter des détails tels que les conditions Déclencheur et de Garde sur une Transition, ou les actions d'Entrée et de Sortie sur un State , dont ils n'auraient peut-être pas eu connaissance autrement - le résultat étant diagrammes alignés sur les meilleures pratiques d'ingénierie, produisant de meilleurs résultats pour les clients.

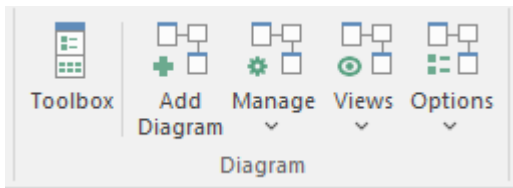
### Création Diagrammes Statemachine

Un diagramme Statemachine peut être créé à partir de plusieurs emplacements dans l' Interface Utilisateur en utilisant :

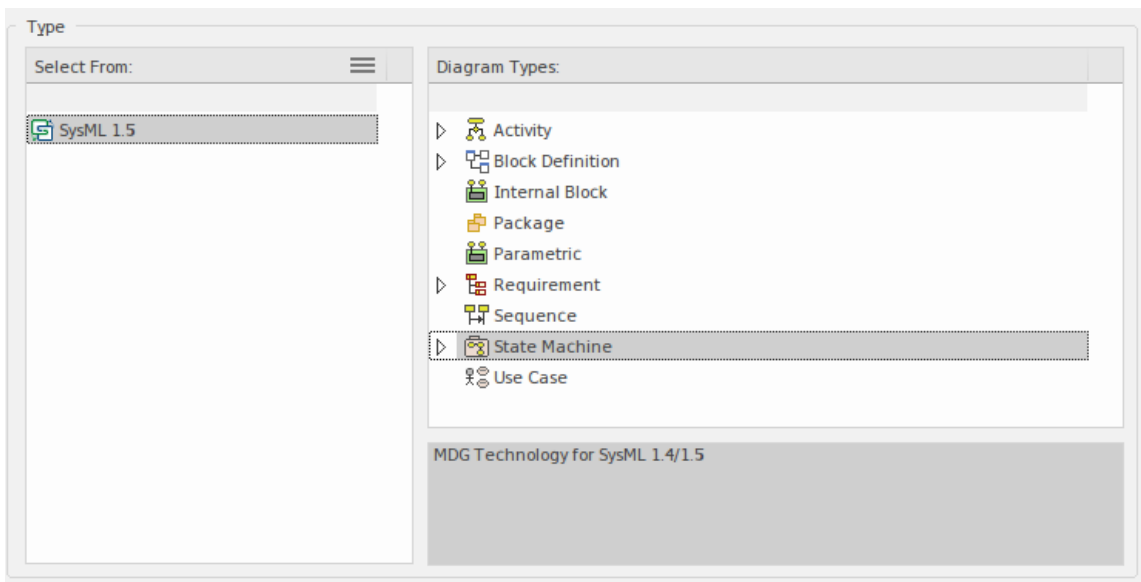
- Ruban de conception - Icône 'Ajouter Diagramme ' sur le panneau ' Diagramme '
- Barre d'outils de la fenêtre Navigateur - Icône ' Nouveau Diagramme '
- Menu contextuel de la fenêtre Navigateur - *option* « Ajouter Diagramme »

Nous allons utiliser le ruban Conception pour créer un diagramme Statemachine . Tout d'abord, vous sélectionnez l'emplacement dans la fenêtre Navigateur où vous souhaitez placer le diagramme Statemachine . Comme pour tous diagrammes , il peut s'agir soit d'un Paquetage , soit d'un élément, mais il est courant d'insérer diagrammes Statemachine dans un élément tel qu'un Bloc pour décrire les phases importantes de la vie d'un Bloc . Une fois l'emplacement sélectionné dans la fenêtre Navigateur , sélectionnez :

Ruban : Conception > Diagramme > Ajouter Diagramme



La sélection de cette option ouvrira la dialogue *Nouveau Diagramme* , qui vous permettra de nommer le diagramme ; le nom sera par défaut le nom du Paquetage ou de l'élément qui contient le diagramme , mais vous pouvez le modifier. Avec la perspective SysML choisie et la version de SysML sélectionnée, une liste de diagrammes s'affichera à partir de laquelle vous pourrez sélectionner le diagramme Statemachine . Une fois le bouton OK sélectionné, un nouveau diagramme Statemachine sera créé à l'emplacement sélectionné dans le Navigateur . Le canevas diagramme s'ouvrira, vous permettant de commencer à ajouter des éléments et des connecteurs qui décrivent les phases importantes de la vie du sujet. Enterprise Architect affichera également les pages Statemachine de la boîte à outils Diagramme , qui contiennent les éléments et les relations définis par la spécification SysML comme applicables à la construction diagrammes Statemachine . N'importe quel nombre d'autres pages de la boîte à outils peut être ouvert si nécessaire, en plus des pages Commun (Éléments) et Relations communes qui sont toujours disponibles.



Les éléments et connecteurs les plus importants utilisés avec le diagramme Statemachine sont :

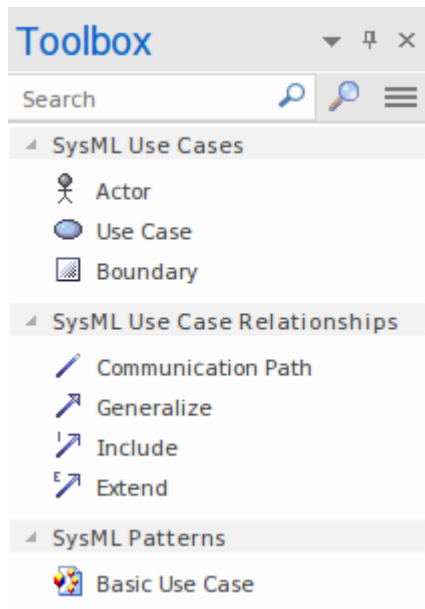
### Éléments

- State - définit une phase significative dans la vie d'une entité
- Statemachine - définit un ensemble d' States

- Initial - définit le point d'entrée d'une région
- Final - définit le dernier State qu'une entité aura
- Histoire - agit comme un memento ou un signet lorsqu'un State est quitté et réintégré

### Connecteurs

- Transition - représente le mouvement d'un State à un autre

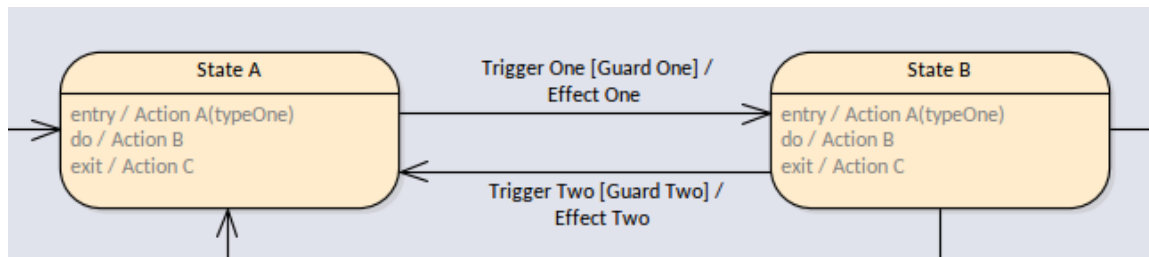


Les éléments peuvent être ajoutés au diagramme en les faisant glisser depuis les pages de la boîte à outils vers le canevas diagramme . Il est considéré comme une bonne pratique de commencer par un élément State final, qui doivent être nommés de manière appropriée pour décrire la manière dont la Statemachine démarre et les multiples manières potentielles dont elle peut se terminer. Laisser le nom vide ou lui donner un nom galvaudé tel que « début » ou « fin » n'aidera pas à faire comprendre au lecteur quel système ou partie d'un système est modélisé et peut conduire à une mauvaise interprétation du diagramme . Avec ces nœuds ajoutés et placés de manière appropriée dans le diagramme , States et des transitions peuvent être ajoutés, définissant ainsi les phases importantes de la durée de vie de l'entité modélisée.

Une fois qu'un diagramme de base a été créé, et à mesure que la connaissance du domaine et des comportements du système est plus approfondie, il est possible d'ajouter Déclencheurs , Événements et les Gardes aux Transitions, et les comportements Entrée, Do et Sortie aux States . Le nouveau venu peut souvent percevoir ces diagrammes comme triviaux, mais ils peuvent révéler des informations profondes qu'il ne serait pas possible de voir autrement.

## Déclencheurs et transitions

La majorité des connecteurs que vous voyez sur un Diagramme Statemachine sont des transitions ; ce sont les lignes qui relient un State à un autre, indiquant les façons autorisées dont le Bloc propriétaire (instance) peut changer. L'ordre dans lequel ils changent et les comportements exécutés dépendent des conditions et du contexte réel du Bloc . Par exemple, un feu de circulation peut clignoter en orange jusqu'à ce que l'ingénieur de maintenance ait corrigé une panne, ou un avion peut maintenir un motif d'attente jusqu'à ce que la tour de contrôle de l'aéroport de destination donne l'autorisation d'atterrir. Ce diagramme montre deux transitions qui sont dirigées dans des directions différentes, créant ainsi la possibilité d'un cycle entre les deux States .



Les transitions proviennent toujours d'un State et en visent un autre, y compris le cas particulier d'une auto-transition où l'origine et la cible sont identiques. Les lignes du diagramme ont une étiquette qui peut afficher un certain nombre d'options différentes : *Déclencheur* , *Garde* et *Effet* . Nous allons discuter de ces options en détail car elles expriment une sémantique importante sur les transitions, notamment si la transition sera exécutée ou non. Une transition peut être dans trois conditions :

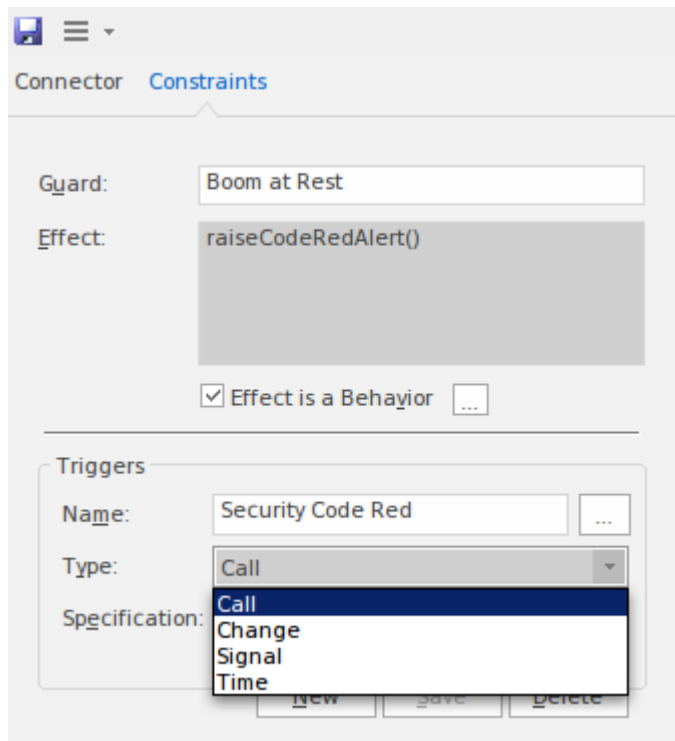
- *Atteint* - l' State d'origine (ou sommet) est actif et prêt à terminer ses comportements
- *Traversé* - la transition est en cours d'exécution (y compris tous les comportements d'effet définis)
- *Terminé* - l'état cible a été atteint et est prêt à exécuter les comportements d'entrée

Ces termes seront utiles aux ingénieurs système et autres lorsqu'ils travailleront en collaboration, discuteront de l'exécution d'une Statemachine et de sa description du comportement du Bloc propriétaire.

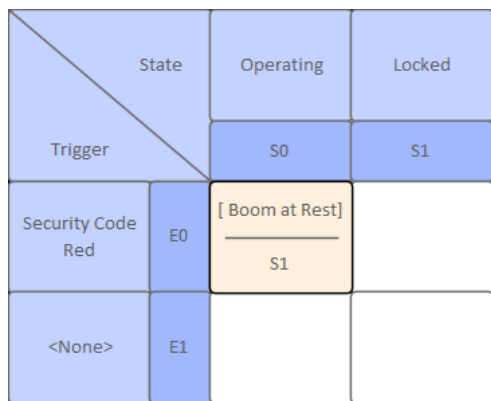
### Déclencheurs

Déclencheurs sont les initiateurs d'une transition et sont associés à des événements que l'on appelle des occurrences d'événements déclencheur . Ce sont ces déclencheurs et leurs événements associés qui entraînent l'exécution d'une transition (déclenchement) et le déplacement Bloc propriétaire d'un état à un autre. Lorsqu'un état est actif, il attend effectivement d'être déclenché par un événement, et tant que son action d'entrée est terminée - indépendamment de tout autre facteur - il est prêt à recevoir et à répondre aux événements.

Cette capture d'écran montre comment le Déclencheur , la Garde et l'Effet peuvent être saisis et visualisés dans Enterprise Architect .



Ce diagramme illustre la manière dont le Déclencheur , la Garde et l'Effet sont représentés sur un diagramme . L'Effet dans ce cas a été défini comme un comportement et a été lié à une Opération définie sur le Bloc .



## Gardes

Les gardes sont les « gardiens » d'une transition et ce n'est que lorsque l'expression du garde est évaluée à True que la transition se déclenche. Si l'expression est évaluée à False, l'événement sera consommé et il n'y aura aucun changement observable dans l'état du Bloc résultant du déclencheur .

L'expression d'une garde peut être définie en anglais simple, mais elle est généralement écrite sous la forme d'une contrainte en utilisant un langage de contrainte formel tel que l' Object Constraint Language. Lorsque vous travaillez avec des simulations ou Statemachines Exécutables la condition est exprimée dans la syntaxe du langage de code dans lequel elle doit être générée, par exemple JavaScript ou C++. Cela s'applique également aux effets. Dans ce diagramme nous pouvons voir une expression mathématique qui peut être évaluée par un humain ou une machine.

Guard:

Effect:

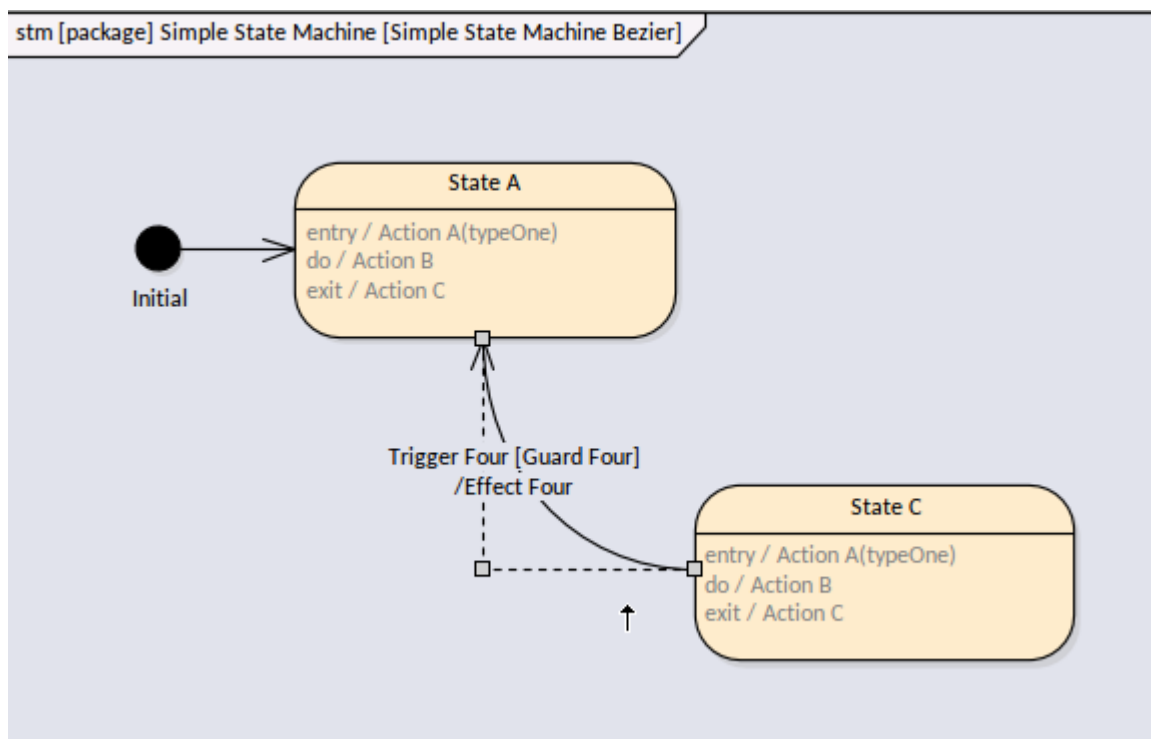
Effect is a Behavior

## Effets

Outre le déplacement du Bloc propriétaire d'un état à un autre, l'importance de la relation Transition est qu'elle peut exécuter un comportement qui peut être une Activité ou une Opération sur le Bloc propriétaire ou sur tout autre Bloc . Ce comportement s'ajoute au comportement de Sortie qui aurait pu être défini sur l' State source et au comportement d'Entrée sur l'État cible. Cela garantit qu'il existe un mécanisme pour modifier le comportement pendant l'exécution de la Transition.

## Courbes de Bézier

Enterprise Architect dispose d'une large gamme d'outils et facilités pour travailler avec diagrammes , notamment diagrammes Statemachine , qui peuvent être utilisés pour aider à créer et visualiser les informations codifiées dans les Statemachines . La possibilité d'acheminer des connecteurs qui contribuent à rendre le diagramme plus attrayant est particulièrement utile lorsque vous travaillez avec des transitions. Le style de ligne du connecteur dans ce diagramme a été défini sur une *courbe de Bézier*, ce qui donne à la transition une apparence moins rigide.



La forme de la courbe peut être modifiée en faisant glisser le point de construction vers un nouvel emplacement. N'importe lequel des nombreux styles de ligne peut être utilisé, offrant au modélisateur une boîte à outils d'options pour la présentation diagramme . Ce menu contextuel peut être sélectionné et le style de ligne défini pour chaque connecteur individuellement. La couleur et l'épaisseur de la ligne peuvent également être définies à partir du ruban Disposition .

Direct	Control+Shift+D
Auto Routing	Control+Shift+A
Custom Line	Control+Shift+C
✓ Bezier	Control+Shift+Z
Tree Style - Vertical	
Tree Style - Horizontal	
Lateral - Vertical	
Lateral - Horizontal	
Orthogonal - Square	
Orthogonal - Rounded	



## States et régions composites


La modélisation des états doit être hiérarchique pour faire face à la complexité des systèmes d'ingénierie ; le SysML fournit deux mécanismes pour modéliser cette hiérarchie dans une StateMachine :

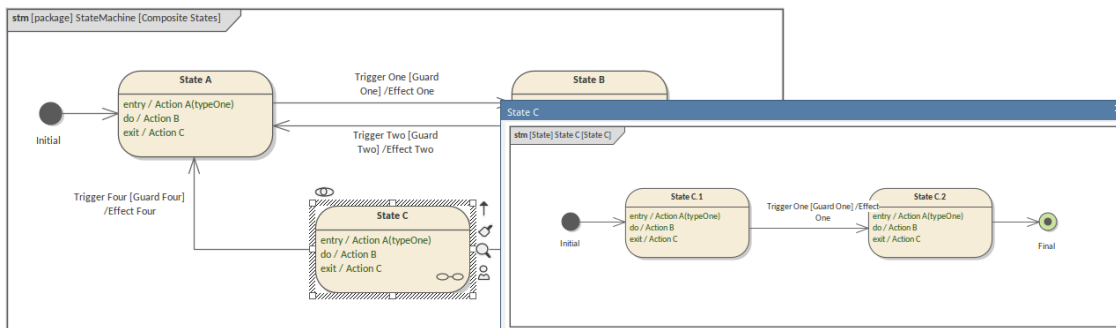
- **Régions** - qui sont des parties distinctes d'une StateMachine
- **States composites** – qui sont States qui contiennent d'autres States

Les systèmes et les objets qu'ils contiennent présentent souvent un comportement concurrent dans lequel deux événements peuvent se produire en même temps ; souvent, ces comportements distincts interagissent les uns avec les autres pour créer un comportement complexe basé sur l'état. Ces comportements sont généralement représentés dans le Bloc que la StateMachine décrit et peuvent impliquer des parties ayant des durées de vie différentes. Ces situations peuvent être modélisées à l'aide de régions ; une StateMachine peut contenir n'importe quel nombre de régions, chacune avec son propre ensemble d' States , de pseudo-états et de transitions qui les relient.

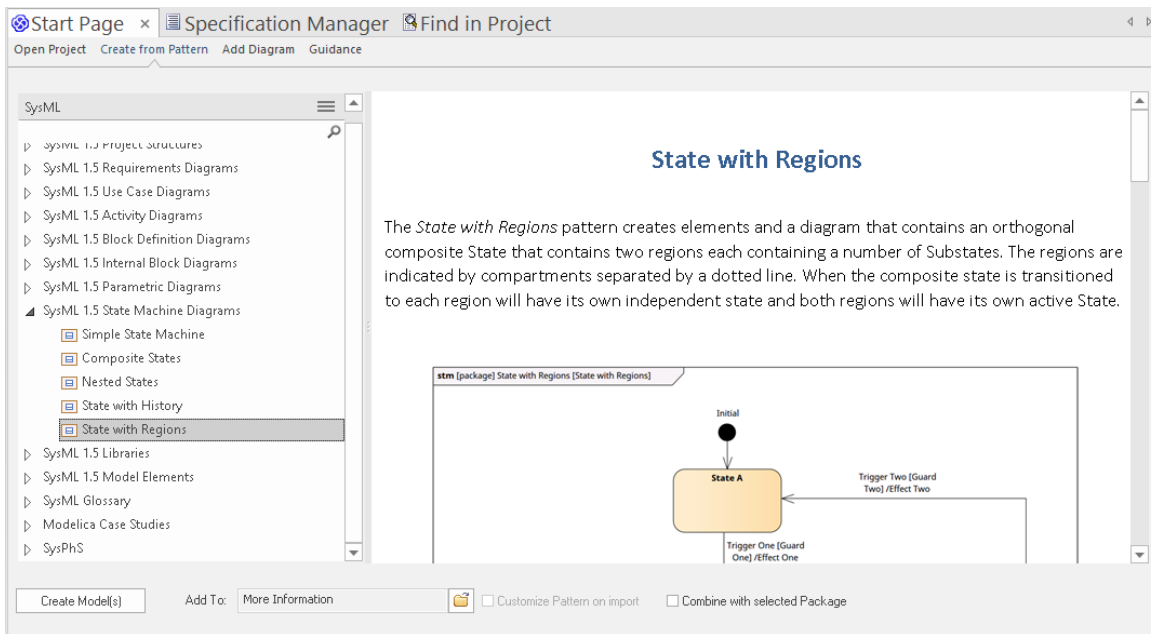
Un State lui-même peut avoir besoin d'être hiérarchique, un State unique pouvant être décomposé en un certain nombre de sous-états représentant les états qui font partie de l' State composite. Par exemple, un robot peut avoir States de fonctionnement et de maintenance ; l' State de maintenance peut être décomposé en un certain nombre de sous-états tels que la recharge de la batterie, la mise à jour de l'environnement et la mise à jour des modules logiciels. Chacun de ces States peut à son tour être décomposé en un certain nombre d'autres States .

Enterprise Architect fournit un point de départ utile pour modéliser du comportement d'un état complexe, en fournissant une série de motifs de modèle qui peuvent être utilisés pour modéliser tous les aspects des StateMachines , y compris modélisation States composites et des régions.

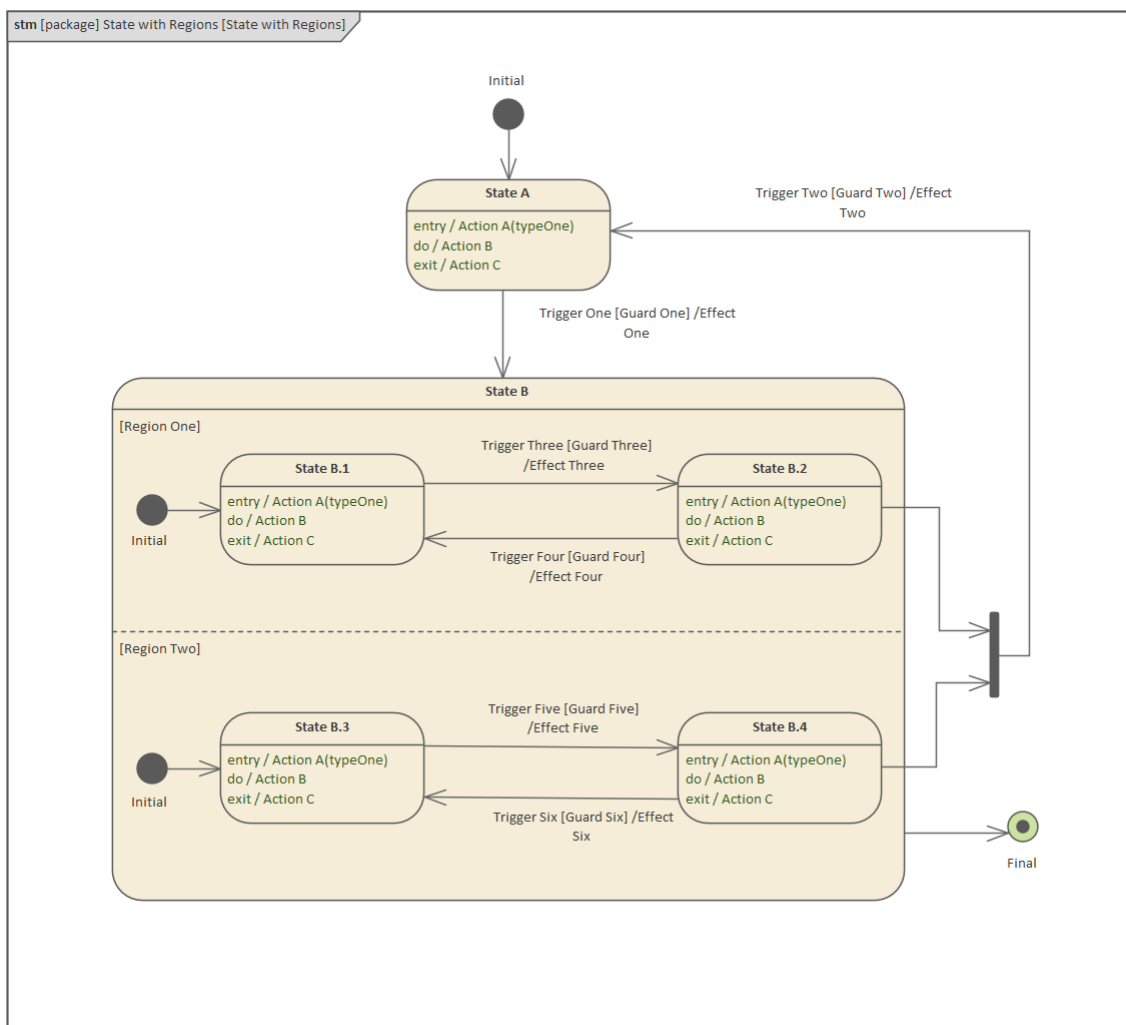
States composites sont indiqués par le symbole oo (« infini ») et font référence à un diagramme enfant. Le diagramme enfant peut être prévisualisé en cliquant sur l'icône  . Un double-clic sur l'élément ou sur l'aperçu diagramme ouvrira le diagramme pour l'édition.



Ces motifs sont accessibles via le Constructeur de Modèle .



Cette image montre un diagramme créé à partir du motif *State avec régions*, qui a été simplement injecté dans le modèle et sert de point de départ au modélisateur. Le modélisateur peut personnaliser le diagramme en remplaçant les noms des States et des Transitions et en ajoutant ou en supprimant des régions selon les besoins.



Les régions de State B fournissent un flux parallèle de processus dans la région 1 et la région 2.



## Pseudo-États – La police de la circulation

Les relations de transition qui relient States dans un diagramme Statemachine doivent être orchestrées pour garantir que la Statemachine est représentative des libertés dont dispose un Bloc pour changer son état dans le monde physique. Les pseudo-états sont des nœuds utilisés pour diriger le flux le long des transitions. Les nœuds peuvent apparaître au début, le long du chemin d'une transition ou à la fin. Les types de pseudo-états incluent :

- *Initial* - utilisé pour initier une Statemachine
- *Fourche et Joindre* - utilisé pour diviser et réunir une transition
- *Terminer* - utilisé pour terminer une Statemachine
- *Joindre* - utilisé pour réunir un certain nombre de transitions
- *Jonction* - utilisée pour diviser une transition
- *Point d'entrée et de sortie* - utilisé sur la bordure d'un State de sous-machine
- *Histoire profonde et superficielle* - utilisées comme souvenirs lorsqu'un State composite est quitté

Nous examinerons chacun d'eux un peu plus en détail et montrerons comment Enterprise Architect peut être utilisé pour créer et gérer ces nœuds importants. Il est important de comprendre que l' State final - qui a une icône analogue au pseudo-état initial - est en fait un State à part entière.

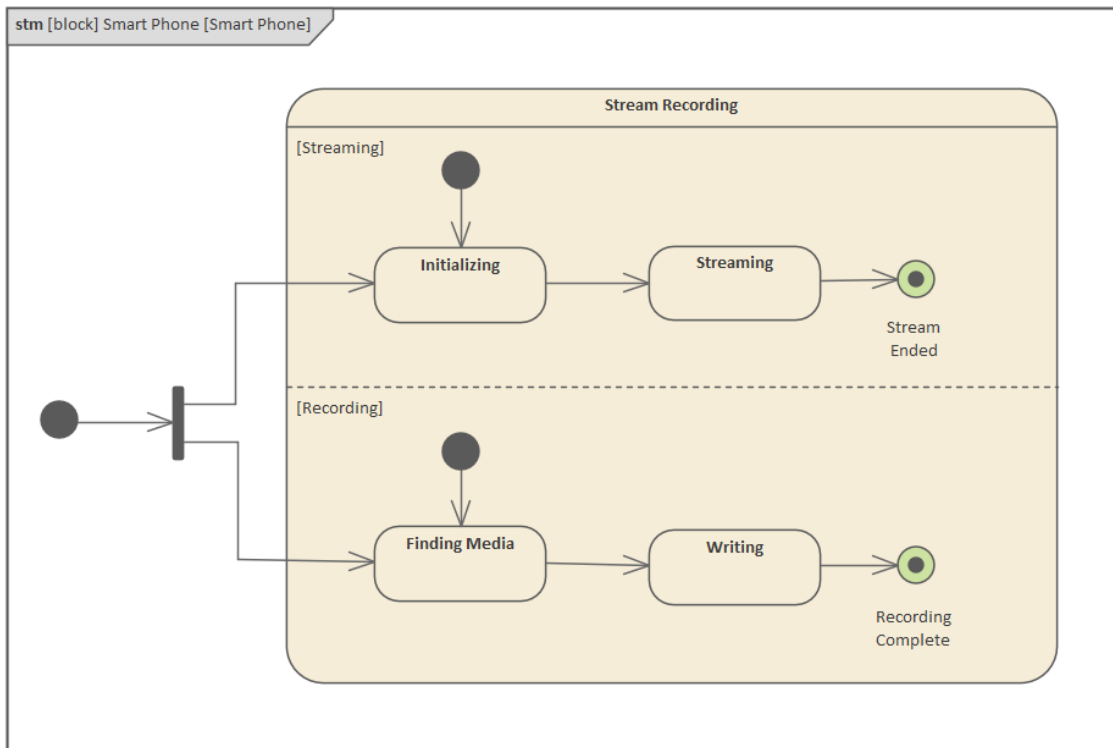
### Initial

Le pseudo-état initial est le plus utilisé de tous les nœuds et représente le point de départ d'une région. Il ne peut y avoir qu'un seul *état initial* dans une région et une seule transition est autorisée à émerger du pseudo-état initial. Comme il s'agit du point de départ, il n'aurait aucun sens d'avoir un déclencheur ou une garde - le pseudo-état devient simplement actif lorsque la région est pénétrée, un modélisateur peut cependant définir un effet. Certains ingénieurs système laisseront ce pseudo-état très important hors diagrammes , en pensant que sa position est implicite, mais il est considéré comme une bonne pratique de les inclure car formellement le point de départ *n'est pas défini* sans eux. Il est cependant courant de laisser le pseudo-état initial sans nom.

Enterprise Architect s'appuiera également sur le fait que les Statemachines soient bien formées et que chaque région ait un pseudo-état initial défini, lorsqu'un modélisateur travaille avec Statemachines Exécutables ou exécute des simulations pour visualiser les States une instance Bloc passera au cours de sa durée de vie.

### Fourche et Joindre

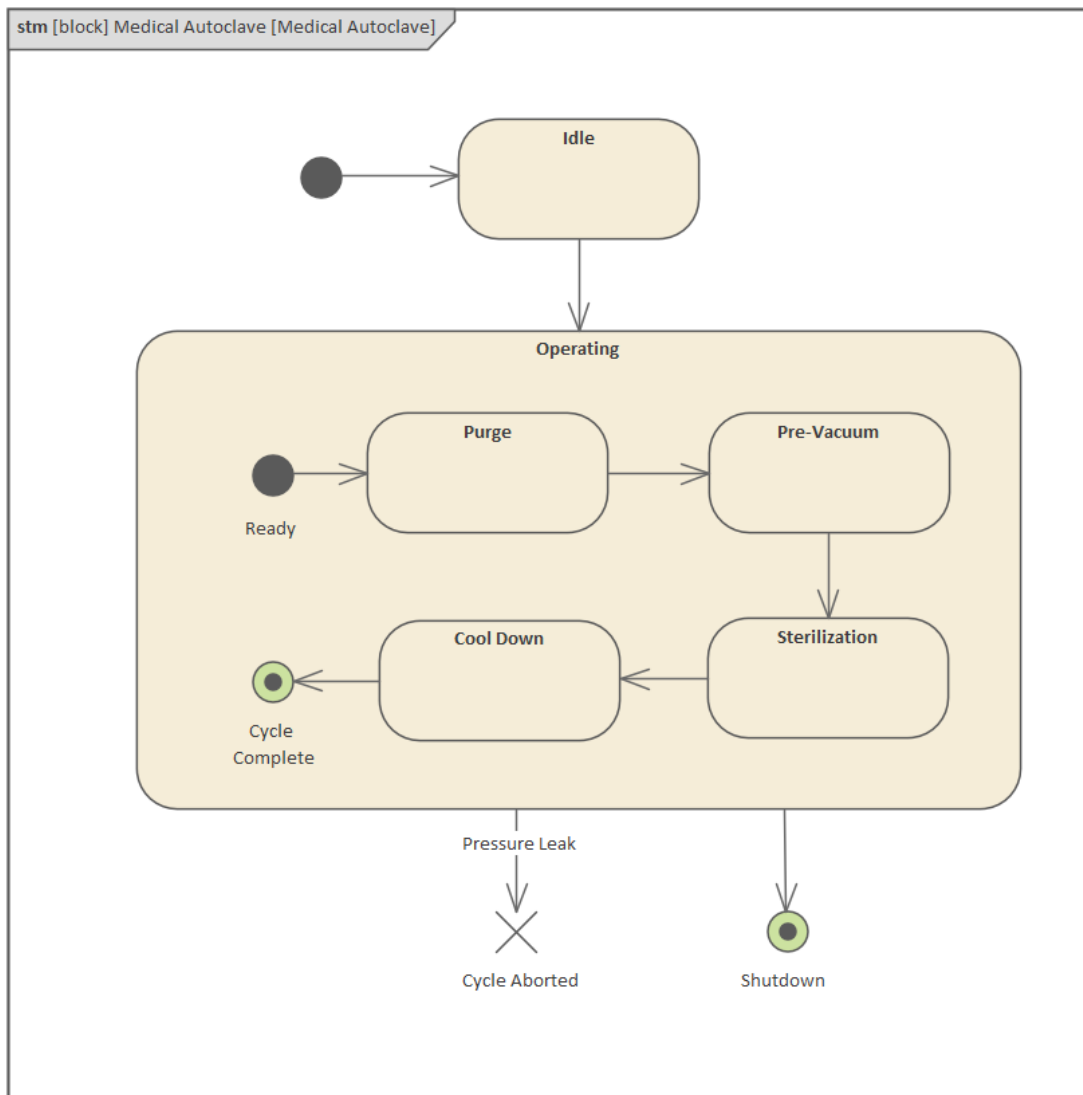
Lorsque des régions sont utilisées dans diagrammes Statemachine , il est souvent nécessaire de diviser une transition qui cible l' State en plusieurs transitions de telle sorte que chaque transition sortante cible un State donné dans chaque région orthogonale. Les transitions *sortantes* d'un pseudo-état Fork sont restreintes et ne peuvent pas avoir de garde ou de déclencheur défini, mais un effet peut être défini. Cela garantit que plusieurs régions peuvent avoir simultanément des états actifs.



Jointures fonctionnent de manière analogue mais opposée pour unir les transitions entrantes provenant de plusieurs régions. La restriction inverse s'applique de telle sorte que les transitions *entrantes* ne peuvent pas avoir de garde ou de déclencheur défini, mais un effet peut être défini. Les effets de toutes les transitions entrantes doivent être terminés avant que la transition sortante puisse se déclencher.

## Mettre fin

Le pseudo-état *Terminate* est un nœud utile pour garantir l'arrêt complet d'une Statemachine . Quel que soit le niveau de la hiérarchie State auquel se trouve le nœud, toutes les régions et tous les niveaux de la hiérarchie s'arrêteront. Il s'agit d'un nœud final dans le sens où la Statemachine propriétaire cessera immédiatement de s'exécuter. La terminaison n'est pas « gracieuse » et tous les comportements en cours d'exécution s'arrêteront simplement ; aucun comportement de sortie ne sera exécuté. Ce diagramme montre un State composite avec une seule région ; s'il y a une *fuite de pression*, le fonctionnement de l'autoclave doit être immédiatement arrêté, il y a donc une transition de l'état de fonctionnement vers le pseudo-état *Terminate Cycle Aborted*.



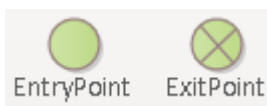
Enterprise Architect permet à ce nœud d'être placé à n'importe quel niveau et dans n'importe quelle région et honorera sa sémantique dans les simulations Statemachine Exécutable .

## Jonction

Le pseudo-état de jonction est utilisé pour modéliser les transitions avec des chemins composés. Il existe des transitions entrantes et sortantes connectées à une jonction, mais pendant l'exécution de la Statemachine une seule des transitions entrantes et une seule des transitions sortantes se déclenchent. Les transitions sortantes sont protégées par des gardes et seule la transition avec une garde dont l'expression est d'abord évaluée à la valeur booléenne true se déclenche et transporte le jeton sortant.

## Point d'entrée et de sortie

Les pseudo-états Point d'entrée et Point de sortie permettent de réutiliser une Statemachine State sous-machine dans de multiples contextes ; ils peuvent également être utilisés sur un State composite. Ces pseudo-états apparaissent comme de petits cercles qui chevauchent la bordure d'un State composite ou d'un State sous-machine ; le point d'entrée est vide tandis que le point de sortie comporte un petit x à l'intérieur du cercle.



Leur position sur la bordure est significative car, d'un point de vue syntaxique visuel, elles permettent de transmettre des messages entre l'intérieur et l'extérieur de l'élément.

## Histoire profonde et superficielle

Le pseudo-état History est comme un signet ou un memento pour un State composite, et stocke simplement le nom du sous-état qui était actif lorsque la région a été quittée. Lorsque la région est ré-entrée ultérieurement, la Statemachine reprend ses transitions à partir du sous-état spécifié par l'historique. Il est possible que pour une raison quelconque une région soit ré-entrée et que l'historique ne soit pas en mesure de fournir le dernier State ; cette situation peut être gérée par un modélisateur en créant de manière préventive une transition du pseudo-état History vers un sous-état par défaut cible ; la transition ne serait utilisée que dans le cas où l'historique ne serait pas en mesure de fournir le memento.

Dans cet exemple de centrifugeuse tubulaire, un ingénieur système a placé un State d'historique peu profond dans l' State composite en fonctionnement, indiquant que si ce démarrage est quitté alors qu'un State donné est actif, puis que l' State est réactivé par la suite, l'exécution reprendra au sous-état actif. La transition de sortie de l'historique indique que, dans le cas où l' State propriétaire est réactivé et que la machine ne sait pas quel State activer, *le rinçage* sera sélectionné.

Next State		Operating											Shutdown
		Initial	Idle	Initial	Loading	Separating	Draining Liquid	Concentrating Discharge	Cleaning	Rinsing	History	History	
State		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												

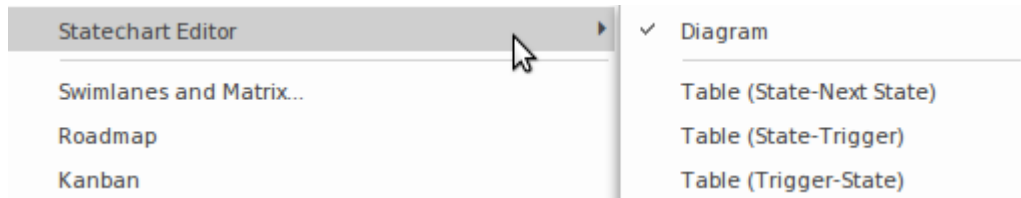
States d'historique superficiel et profond fonctionnent de la même manière, à l'exception du fait qu'un pseudo-état d'historique superficiel ne mémorise que les sous-états actifs de la région propriétaire, tandis qu'un historique profond peut mémoriser jusqu'à n'importe quel niveau dans une hiérarchie de sous-états. L'historique profond est indiqué visuellement par un astérisque placé après le H\*.





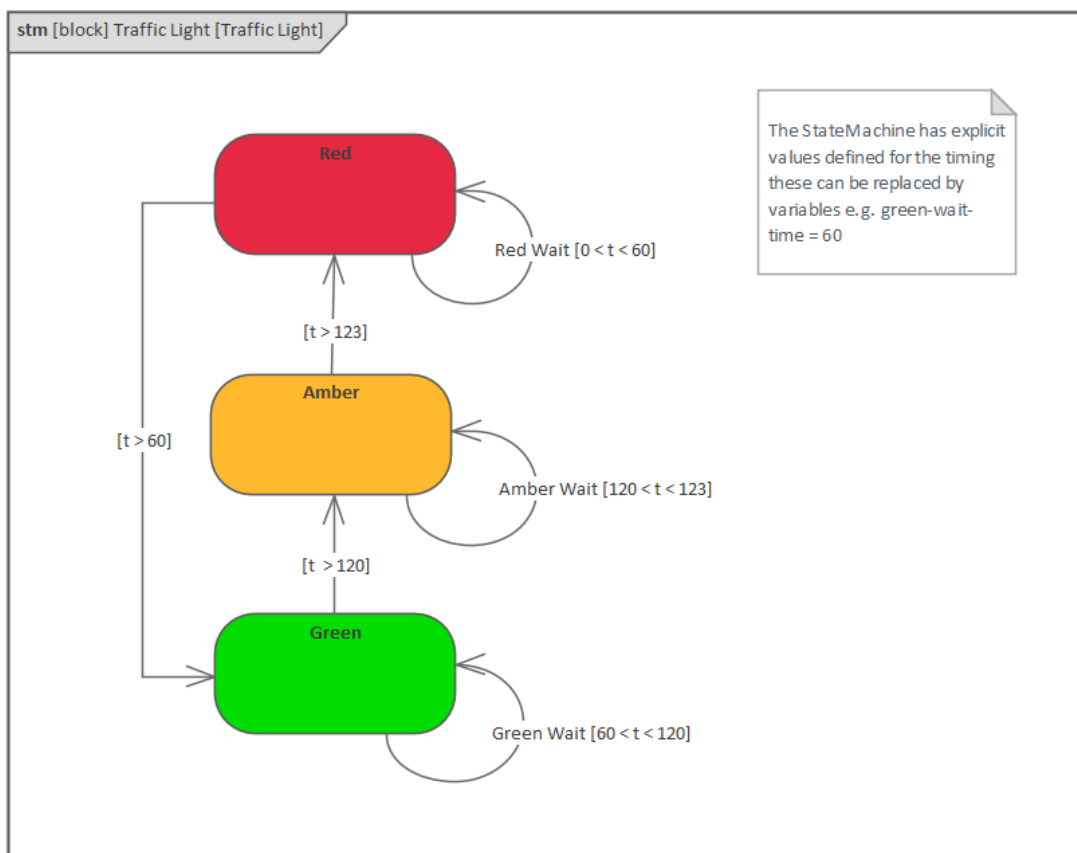
## Tableaux State - Une autre Vue

Une StateMachine peut être visualisée de différentes manières. Nous avons déjà examiné une vue diagramme de la StateMachine, mais Enterprise Architect dispose d'un outil utile qui permet de visualiser la StateMachine dans un tableau. Il existe trois variantes de la visualisation tableau :



- State State Vue - où les States et les sous-états sont organisés en lignes et en colonnes et les cellules représentent les transitions
- State Déclencheur Vue - où les déclencheurs sont organisés en colonnes et les états en lignes et les cellules représentent les transitions
- Déclencheur State Vue - où les déclencheurs sont organisés en lignes et les déclencheurs en colonnes et les cellules représentent les transitions.

Ce diagramme des changements d'un feu de circulation, comme tout diagramme StateMachine, peut être converti en un Tableau State.



Cette vue sera attrayante et plus naturelle pour certains publics, et l'ingénieur pourra simplement basculer d'une vue à l'autre. Les States et leurs sous-États sont représentés à la fois sur les lignes et les colonnes du tableau, et les transitions (représentant les chemins entre States) sont représentées dans les cellules.

Next 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 ... _____





Les deux autres vues tableau , comme indiqué précédemment, permettent de visualiser les States par rapport aux Déclencheurs qui les initient. Ces vues sont utiles lorsque l'ingénieur est plus intéressé par l'analyse causale et souhaite visualiser ou analyser comment les événements et déclencheurs entraînent le comportement State du Bloc propriétaire. Ce diagramme montre la même machine de feux de circulation représentée sous forme de tableau de déclencheurs et States .

State		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 permet également d'exporter ces tableaux afin de les analyser à l'aide d'une feuille de calcul. Il s'agit d'un mécanisme utile, en particulier lorsque la Statemachine est complexe et qu'il existe un grand nombre de transitions.

Trigger \ State		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

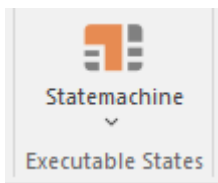
State Table Options...

-  Properties...
-  Lock Diagram
-  Save Current Changes Control+S
- Statechart Editor ▶
- Export Statechart to CSV file... ▶
- Execute Simulation ▶
-  Help...

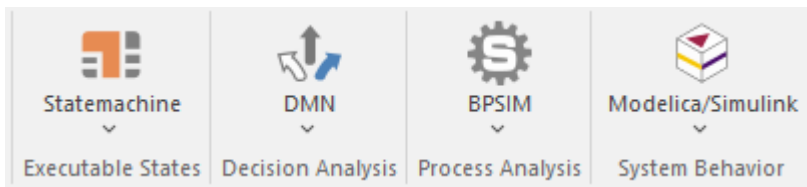
## Visualisation et mise en œuvre avec des simulations

Enterprise Architect n'est pas seulement une plate-forme de création et de gestion diagrammes Statemachine , il fournit également facilités Simulation sophistiqués pour les ingénieurs et autres parties prenantes afin de visualiser les Statemachines . Cela donne vie aux modèles et fournit un outil de visualisation, non seulement pour l'ingénieur qui développe les modèles, mais aussi pour les autres publics, techniques et non techniques, qui ont besoin de comprendre ce que dit le modèle. C'est un peu comme un auteur lisant à haute voix un passage de texte nouvellement écrit, et cela peut aider l'ingénieur à trouver des erreurs dans les modèles ou des aspects des modèles qui devraient être corrigés ou retravaillés. C'est particulièrement utile lorsque les modèles deviennent plus complexes, avec des sous-états imbriqués, Déclencheurs et des gardes complexes, et des pseudo-états tels que des fourches et Jointures qui divisent et réunissent les transitions.

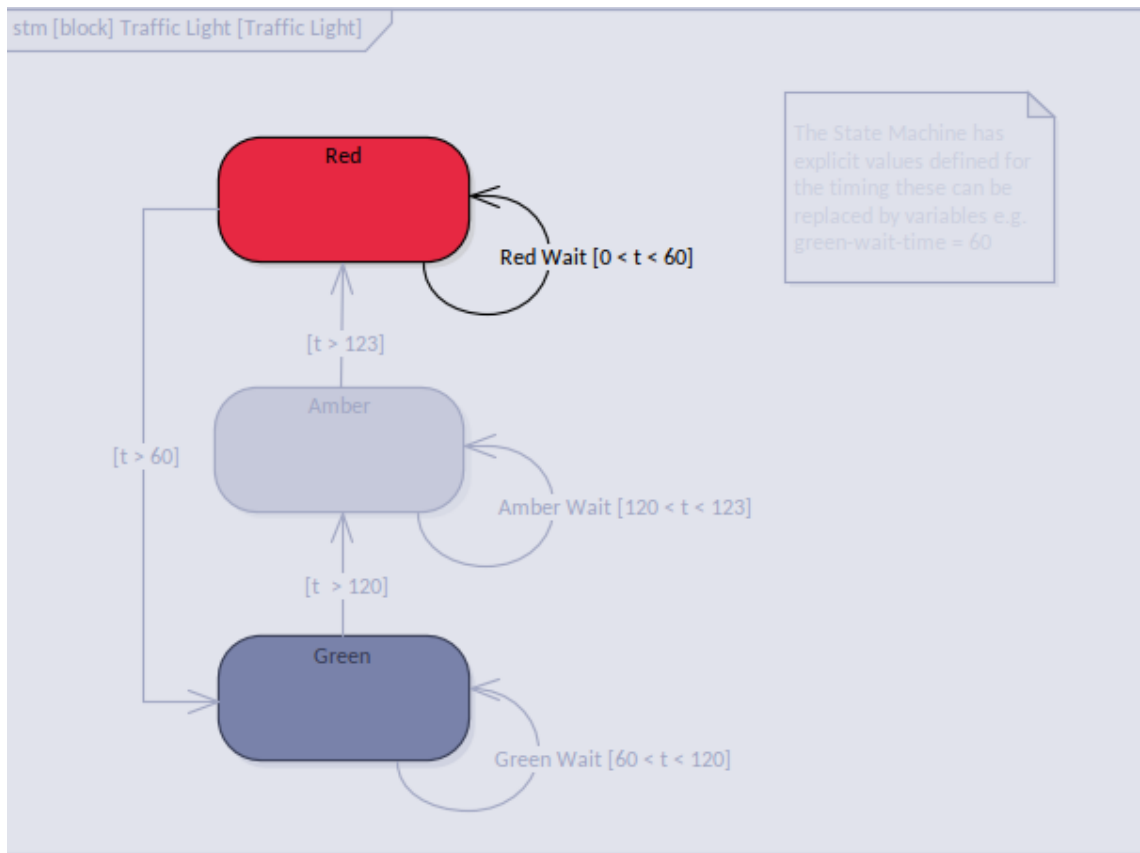
Avec support étendue des Déclencheurs , Ensembles Déclencheur , States imbriqués, States concurrents, des effets dynamiques et d'autres fonctionnalités de simulation avancées, la fonctionnalité fournit un environnement sophistiqué dans lequel créer des modèles interactifs et fonctionnels qui aident à explorer, tester et tracer visuellement le comportement complexe des entreprises, des logiciels et des systèmes. Il existe un ruban dédié à la simulation, qui fournit une gamme d'éléments pouvant être utilisés pour les simulations dynamiques et exécutables de Statemachines . Cette image montre les principaux outils permettant de travailler avec des simulations dynamiques.



La deuxième image montre les autres outils avancés, notamment l' Statemachine Exécutable , qui peuvent être utilisés pour créer des exécutions de la Statemachine afin de produire un code de programmation entièrement implémentable et compilable directement à partir des Statemachines simulées. L'image montre également un certain nombre d'autres facilités , notamment Modelica et Simulink, qui sont des outils sophistiqués pour exécuter des simulations paramétriques complexes.

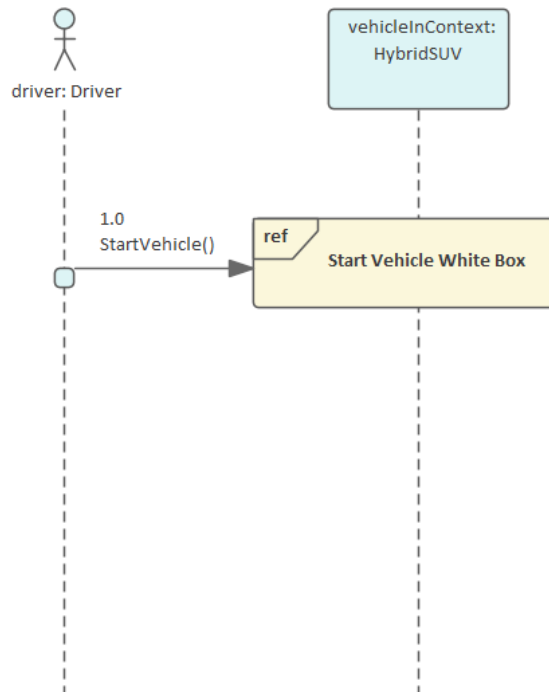


Ce diagramme montre une simulation dynamique simple du système de feux de circulation que nous avons étudié dans l'exercice précédent. Il montre la simulation dynamique en action.



## Les interactions comme Séquence de messages

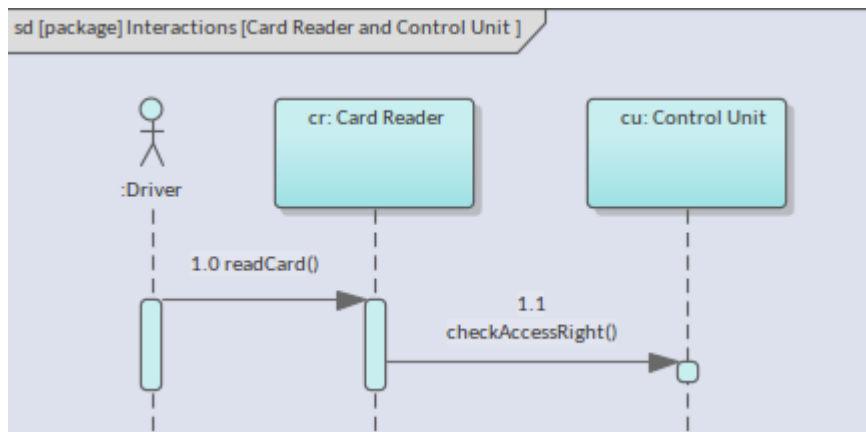
Les systèmes sont constitués de parties et le comportement global d'un système est assuré par ces parties travaillant ensemble de manière orchestrée. Communication entre les parties et la synchronisation de leurs comportements sont essentielles, tant du point de vue de la conception que de la visualisation. Les unités structurelles d'un système, représentées principalement par les blocs, échangent des messages et des signaux qui déclenchent des comportements, ce qui donne lieu à un comportement coordonné du système qui représente les fonctions du système.



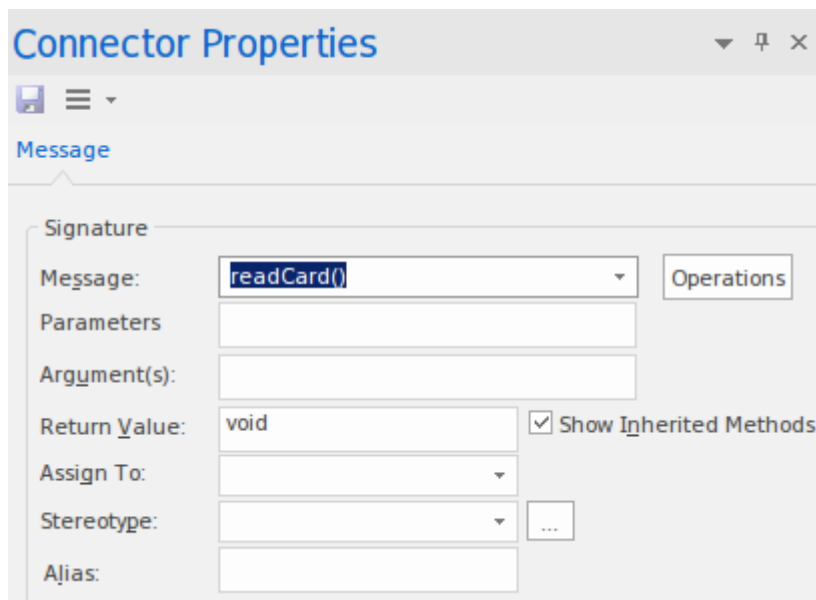
Cet échange de messages et de signaux, ainsi que les comportements qui en découlent, peuvent être représentés sur un diagramme Séquence qui montre les messages et les signaux séquencés dans le temps entre les instances Bloc qui participent à une interaction spécifique.

## Lignes de vie, messages et activations

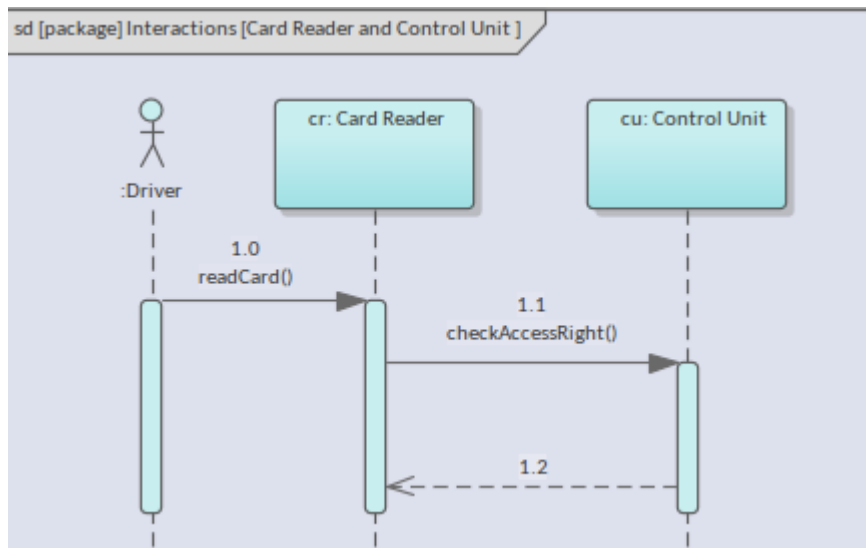
Dans un diagramme Séquence, les blocs qui participent à l'interaction ont une durée de vie représentée par une ligne en pointillés, partant de la base de l'élément et se poursuivant verticalement pendant toute la durée de vie de l'élément. Les éléments peuvent être créés ou détruits à tout moment pendant la période représentée par le diagramme Séquence, et la ligne de vie représente donc leur existence. Les éléments présents en haut du diagramme sont créés au début de l'interaction. Un échange de messages entre un expéditeur et un destinataire débutera par une ligne de vie (l'expéditeur) et se terminera par une autre (le destinataire).



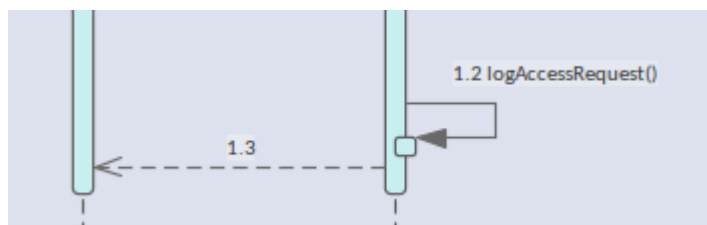
L'expéditeur appelle ou active effectivement un comportement fourni par la ligne de vie réceptrice. Enterprise Architect fournit des mécanismes utiles pour utiliser les comportements déjà définis sous forme d'opérations. Cette illustration montre les propriétés d'un message existant, que le modélisateur peut sélectionner dans une liste déroulante d'opérations précédemment définies. Le modélisateur peut également définir une nouvelle opération en sélectionnant le bouton Opérations.



Formellement, lorsqu'un message cible une ligne de vie, une exécution se produit, ce qui signifie qu'un comportement est initié ou augmenté. Cette exécution est représentée visuellement par une activation, qui est dessinée sur le diagramme sous la forme d'une fine superposition rectangulaire sur la ligne de vie, dont la longueur représente la durée relative du comportement. L'étendue de l'activation du rectangle prend fin lorsqu'un message de réponse est renvoyé à l'appelant.

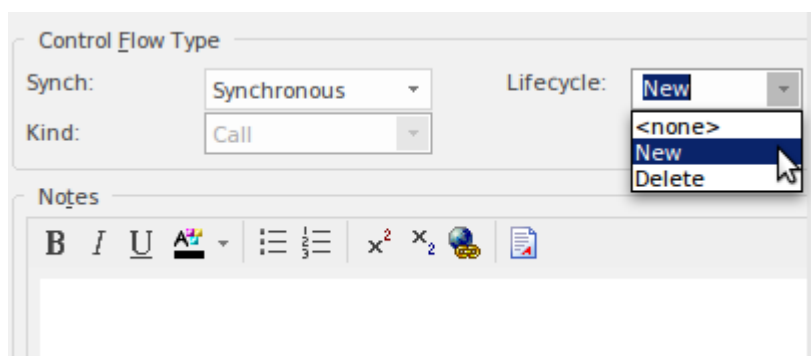


Un Bloc (Instance) peut être à la fois l'expéditeur et le destinataire d'un message, qui peut être qualifié de message réflexif car il commence et se termine sur la même ligne de vie. Dans ce cas, un deuxième rectangle d'activation plus court est superposé sur le premier mais décalé vers la droite.



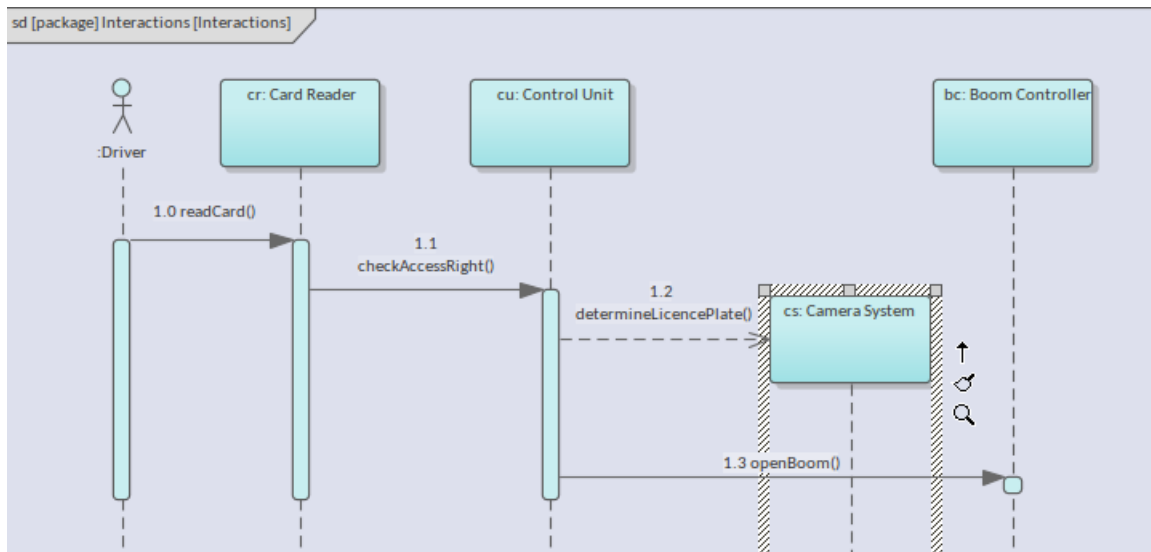
## Messages qui créent et détruisent des blocs

Un nombre quelconque d'instances Bloc peut faire partie d'une interaction, et souvent un ensemble de ces instances sera présent pendant la durée représentée par le diagramme Séquence. Ces blocs seront positionnés sur une ligne en haut du diagramme. Il est cependant possible que des blocs soient créés et détruits pendant la séquence du diagramme. Par exemple, un Bloc particulier peut n'être nécessaire que pendant une courte période et peut donc être instancié, exécuter sa fonction puis être détruit. Enterprise Architect permet à un ingénieur de spécifier qu'un message est un message *de création*, ce qui signifie que le Bloc récepteur sera instancié à ce moment dans la séquence temporelle du diagramme. Pour cela, définissez l'action du message *Cycle de vie* sur « Nouveau » comme indiqué dans cette illustration.

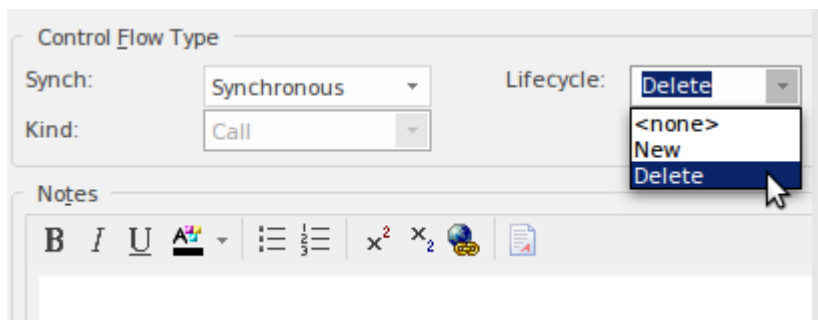


La définition de cette propriété a pour effet de déplacer le Bloc ciblé vers une position plus basse dans le diagramme, ce qui signifie que l'instance Bloc ne sera pas créée avant ce point dans la séquence temporelle globale du diagramme. Le style de la ligne de message est également modifié en une ligne en pointillés avec une pointe de flèche ouverte pour indiquer qu'il s'agit d'un message *de création*. Les autres aspects de la sémantique et de l'effet de ce message restent inchangés.

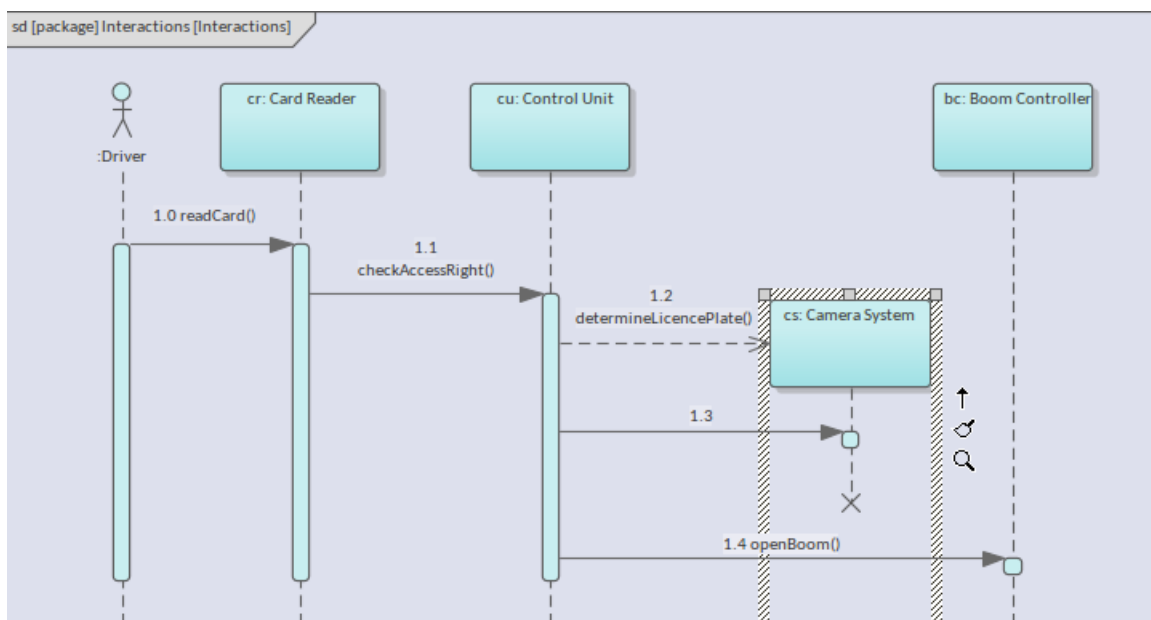




De la même manière, la durée de vie d'une instance peut être terminée en envoyant un message *Destroy*. En pratique, cela signifie que l'instance a rempli son rôle et n'est plus nécessaire. Cela peut être réalisé facilement en définissant à nouveau une propriété de message, mais cette fois-ci, nous définissons l'action du message *Life Cycle* sur « Delete ».



La définition de cette propriété a pour effet de mettre fin immédiatement à la durée de vie de l'instance ; ceci est représenté visuellement par la ligne de vie en pointillés arrêtée par une petite croix sur la ligne de vie au moment où le message arrive.

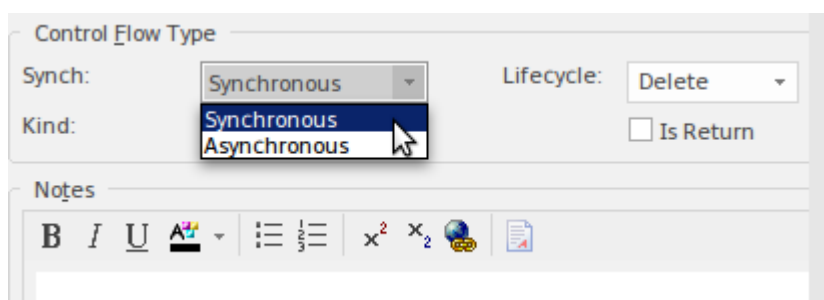


Lorsqu'un diagramme Séquence représente un système logiciel qui dispose d'une mémoire limitée disponible, la destruction de l'instance ciblée entraînera le retour de la mémoire alloué au pool de mémoire. En ingénierie système, la gestion de la durée de vie des objets électromécaniques peut présenter de nombreux autres avantages, tels que la consommation d'énergie, la surchauffe, la disponibilité ou même des risques tels que la sécurité.

## Messages synchrones et asynchrones

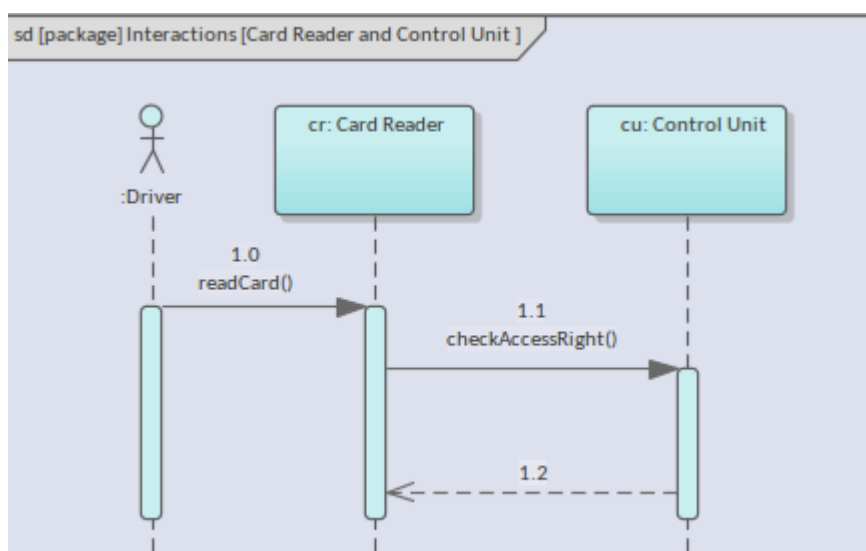
Les messages représentent essentiellement l'envoi d'un type de requête d'un expéditeur à un destinataire. L'expéditeur d'un message peut interagir avec le destinataire de deux manières fondamentales. Le premier type de message est appelé message *synchrone* car les messages se produisent à peu près au même moment. Avec ce type de message, l'expéditeur attend que le destinataire réponde avant d'envoyer des messages supplémentaires. Le deuxième type de message est appelé *asynchrone* car l'expéditeur n'attend pas la réponse du destinataire avant de poursuivre l'exécution, y compris l'envoi de messages supplémentaires à ce destinataire ou à d'autres.

Par défaut, Enterprise Architect crée *des messages synchrones*, mais le type de message peut être modifié en définissant la propriété « Type de synchronisation » dans la dialogue « Propriétés ». Lorsque le type de message est défini sur Synchronique (par défaut), le style de ligne est défini sur solide avec une pointe de flèche fermée pointant vers l'instance du destinataire. La ligne peut être annotée avec le nom et les paramètres du message.



Les messages *asynchrones* représentent des appels à des opérations ou *des signaux* envoyés aux destinataires. Dans les deux cas, l'expéditeur n'attend pas de réponse et ne suspend pas son exécution en attendant une réponse. Dans le cas de l'appel à une opération, l'opération elle-même serait définie comme asynchrone et le système ou la machine représentés par le diagramme saurait qu'il ne faut pas attendre un retour de message (réponse). Enterprise Architect permet de définir ce type de message via la propriété « Synch » comme expliqué précédemment. Lorsque le type de message est défini sur Asynchrone, cela définit le style de ligne en pointillé avec une pointe de flèche ouverte pointant vers l'instance du destinataire. La ligne peut être annotée avec le nom et les paramètres du message.

Il existe un troisième type de message qui peut éventuellement être utilisé avec un message synchrone, il s'agit du *message de réponse*. Ce message signifie qu'une opération a été invoquée lorsque le destinataire a renvoyé un accusé de réception indiquant que le comportement a été exécuté et est terminé. L'inclusion de messages de réponse dans un diagramme est une décision stylistique. Certains ingénieurs et modélisateurs préfèrent les omettre diagrammes pour réduire l'encombrement visuel. Si un type de retour et valeur ont été définis, ils seront renvoyés dans le cadre du message de réponse.



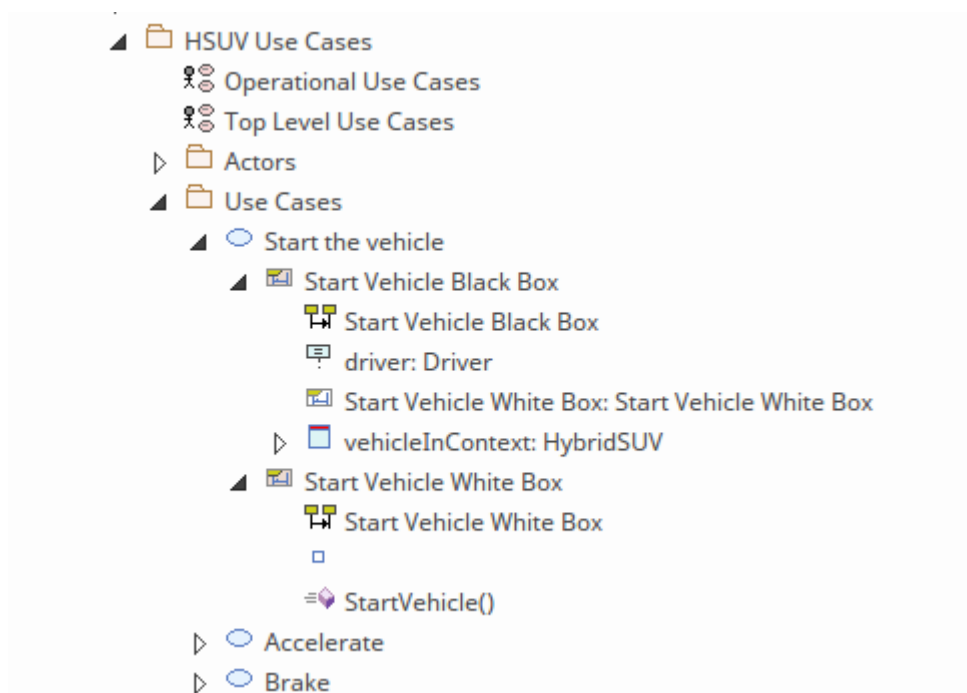
Les blocs peuvent avoir à la fois *des opérations* et *des réceptions* spécifiées dans le cadre de leur définition ; ce paradigme décrit la relation des opérations et des réceptions avec les messages et les signaux.

- Appel synchrone d'opération - Message synchrone
- Appel asynchrone d'opération - Message asynchrone
- Réception Réception du signal - Message asynchrone

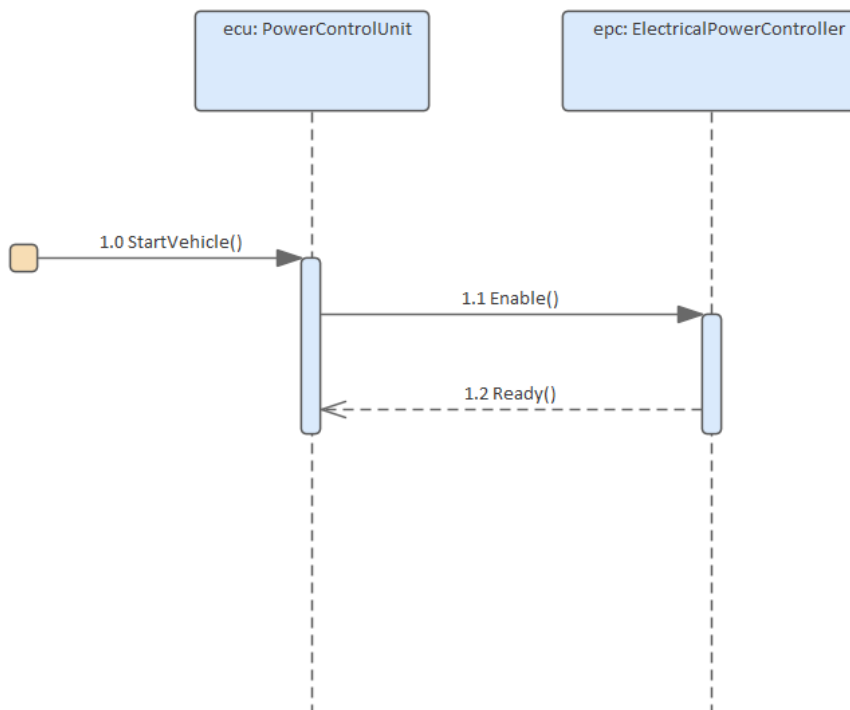
## Présentation du Diagramme de Séquence

Le diagramme Séquence trouve son origine dans le Unified Modeling Language ) et a été principalement utilisé dans ce langage pour représenter les interactions entre composants dans des systèmes centrés sur le logiciel. Son utilisation a été élargie dans le contexte de Ingénierie des Systèmes , où il est utilisé de manière plus générique pour représenter l'échange séquentiel dans le temps de messages et de signaux entre les unités structurelles d'un système ou d'une partie d'un système.

Le diagramme Séquence comporte deux axes ; par convention, l'axe horizontal (x) représente les Bloc (instances) qui participent à l'interaction et l'axe vertical (y) représente le temps. Les blocs ne doivent pas nécessairement être ordonnés de manière prescrite, mais un modélisateur les placera généralement dans l'ordre le plus illustratif et cet ordre est souvent basé sur le moment où ils sont utilisés dans l'interaction. Le temps ne s'exécute pas sur une échelle linéaire et l'échelle de temps entre deux diagrammes peut être très différente. Par exemple, l'échelle de temps sur diagramme Séquence représentant un système photographique à grande vitesse serait très différente de l'échelle sur un diagramme représentant une machine de caisse d'épicerie. Ce diagramme montre l'emplacement de deux diagrammes Séquence (« Démarrer Vehicle Black Box » et « Démarrer Vehicle White Box ») qui sont des nœuds enfants d'un cas d'utilisation nommé « Démarrer the Vehicle ».



L'arborescence sert d'aide à la navigation et en double-cliquant sur l'élément dans la fenêtre Navigateur , vous ouvrirez le diagramme à partir de cette vue.



Dans le deuxième diagramme nous voyons un diagramme Séquence simple qui représente la séquence de messages impliqués dans le démarrage d'un véhicule. On peut voir à partir du diagramme qu'il y a deux blocs (instances) qui font partie de l'interaction, et des messages sont échangés entre les deux blocs et l'initiateur de l'interaction, et finalement le cas d'utilisation.

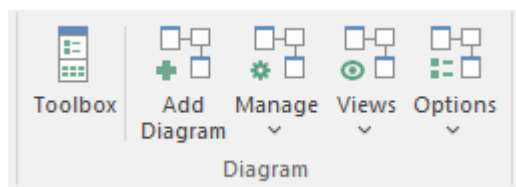
## Création de Diagrammes Séquence

Un diagramme Séquence peut être créé à partir de plusieurs emplacements dans l' Interface Utilisateur en utilisant :

- Ruban de conception - *Ajouter Icône Diagramme sur le Panneau de Diagramme*
- Barre d'outils de la fenêtre Navigateur - *Nouveau Diagramme Icône*
- Fenêtre Navigateur *Menu Contexte - Ajouter Diagramme*

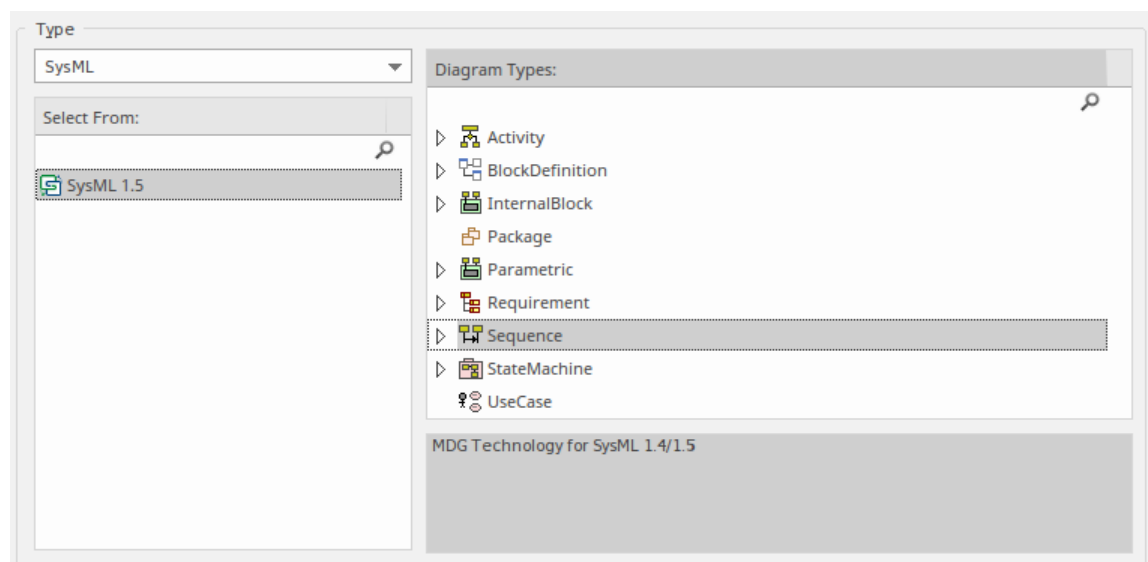
Nous allons utiliser le ruban Conception pour créer un diagramme Séquence . Vous devez d'abord sélectionner l'emplacement dans la fenêtre Navigateur où vous souhaitez créer le diagramme Séquence . Comme pour tous diagrammes , il peut s'agir soit d'un Paquetage , soit d'un élément, mais il est courant d'ajouter diagrammes Séquence à un Paquetage car il implique généralement un certain nombre d'objets dans le Paquetage . Une fois l'emplacement Paquetage sélectionné dans la fenêtre Navigateur , sélectionnez :

Ruban : Conception > Diagramme > Ajouter



La sélection de cette option ouvrira la dialogue « Nouveau Diagramme » qui vous permettra de nommer le diagramme ; initialement, le nom sera par défaut le nom du Paquetage ou de l'élément qui contient le diagramme . Avec la perspective SysML choisie et la version de SysML sélectionnée, une liste de diagrammes s'affichera, permettant la sélection du diagramme Séquence . Une fois le bouton OK sélectionné, un nouveau diagramme Séquence sera créé à l'emplacement sélectionné dans la fenêtre Navigateur . Le canevas diagramme s'ouvrira, vous permettant de commencer à ajouter des éléments et des connecteurs qui décrivent les interactions importantes entre les objets. Enterprise Architect affichera

également les pages de la boîte à outils diagramme Séquence qui contiennent les éléments et les relations définis par la spécification SysML comme applicables à la construction diagrammes Séquence . N'importe quel nombre d'autres pages de la boîte à outils peut être ouvert si nécessaire, en plus des pages Éléments communs et Relations communes qui seront toujours disponibles.



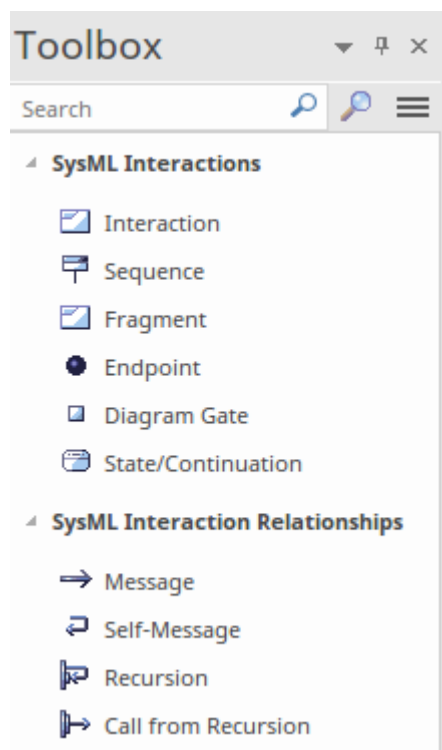
Les éléments et connecteurs les plus importants utilisés avec le diagramme Séquence sont :

#### Éléments

- Interaction
- Séquence
- Fragment
- Point final
- Porte Diagramme
- State / Suite

#### Connecteurs

- Message
- Message personnel
- Récursivité
- Appel depuis la récursivité

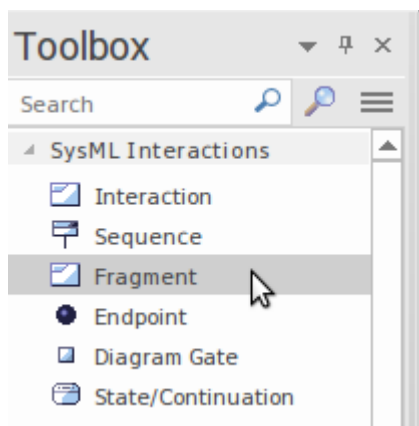


Les éléments peuvent être ajoutés au diagramme en les faisant glisser depuis le Navigateur ou depuis la Boîte à outils vers le canevas diagramme . Le processus typique consiste à réutiliser des éléments existants tels que les Blocs, qui ont des comportements sous forme d'opérations pouvant être sélectionnées comme base pour les messages échangés entre les lignes de vie. Les éléments peuvent être ajoutés au diagramme sous forme de lien, mais plus généralement, ils sont ajoutés sous forme de ligne de vie.

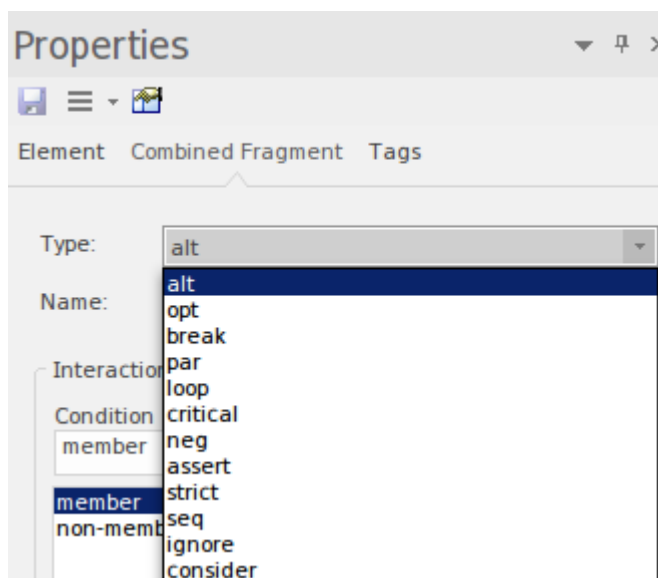
Une fois qu'un diagramme de base a été créé, et à mesure que la connaissance du domaine et des interactions du système est révélée, il est possible d'ajouter des fragments, des points de terminaison, des portes Diagramme et des éléments State /de continuation.

## Orchestration de messages avec des fragments

De nombreux systèmes sont intrinsèquement complexes et, bien que diagrammes Séquence simples soient utiles pour transmettre une image globale d'un élément logiciel ou d'un dispositif électromécanique, ils doivent être augmentés pour permettre la création de modèles sophistiqués de ces systèmes plus complexes. L'une des options de modélisation de la complexité dans les flux de messages est le fragment combiné. Ceux-ci peuvent être utilisés pour séquencer les messages différemment, notamment en permettant de sélectionner des messages particuliers dans certaines circonstances ou d'exécuter un message plusieurs fois. Il existe un ensemble de fragments combinés qui peuvent être utilisés et leur opérateur détermine le type de fragment. Enterprise Architect supporte tous les opérateurs, ce qui permet aux ingénieurs de créer diagrammes capables de décrire de manière adéquate les systèmes d'ingénierie complexes modélisés. Les fragments peuvent être ajoutés à un diagramme directement à partir de la page de la boîte à outils Interactions et peuvent être positionnés pour superposer le groupe de messages approprié.

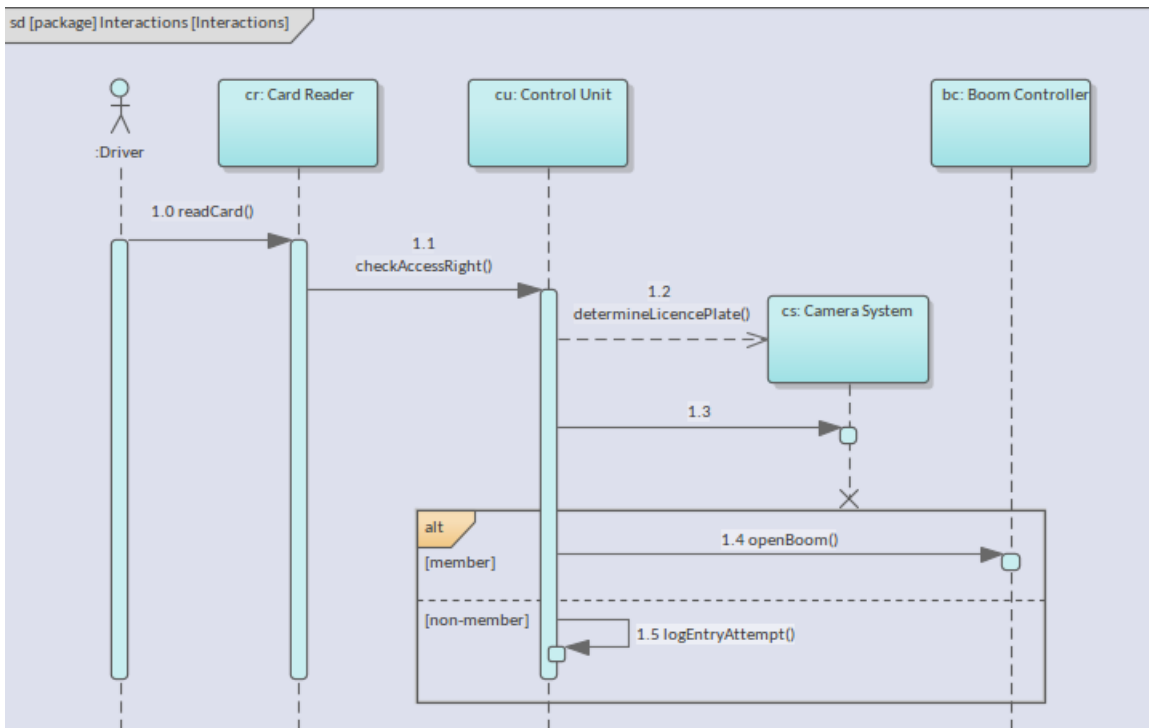


L'élément placé sur le diagramme est un fragment générique et son opérateur doit être défini. Pour cela, sélectionnez une valeur pour le type dans la feuille de propriétés du fragment combiné.



Cela modifiera le fragment vers le type approprié et permettra, par exemple avec le type *alt*, de définir un nombre quelconque de conditions alternatives qui détermineront quel message sera déclenché.





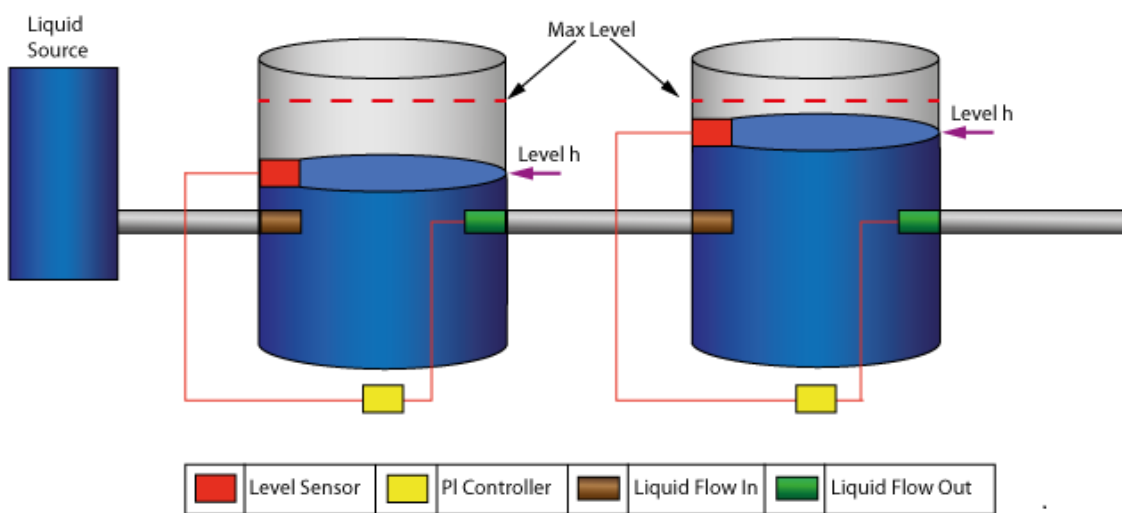
## Visualisation avec des simulations

Au fil des siècles, les hommes ont élargi leur connaissance du monde grâce à l'étude des mathématiques et de la physique et à l'application de systèmes de pensée et d'équations à des problèmes réels. Grâce à ces connaissances, nous avons construit des automobiles et des avions, envoyé des hommes sur la Lune, divisé l'atome et résolu d'innombrables autres problèmes complexes. Ces connaissances, dans l'Antiquité, se transmettaient de bouche à oreille, par des tablettes d'argile et des rouleaux de papyrus, puis par des livres laborieusement rédigés par des scribes. Puis, avec l'avènement de l'imprimerie, ces connaissances ont été consignées sous forme d'articles, de revues et de livres et largement diffusées. Quatre cents ans se sont écoulés avant l'arrivée d'Internet et une grande partie des connaissances existantes a été transférée vers des supports en ligne sous forme de documents, de pages et de sites consacrés à ces disciplines, les rendant accessibles à un grand nombre de personnes aux quatre coins du monde. Aujourd'hui, une nouvelle ère est arrivée où ces connaissances peuvent être utilisées pour construire des modèles qui nous permettent de visualiser ces équations en mouvement et dans leur contexte, avec des paramètres spécifiques à notre domaine et aux problèmes que nous essayons de comprendre et de résoudre.

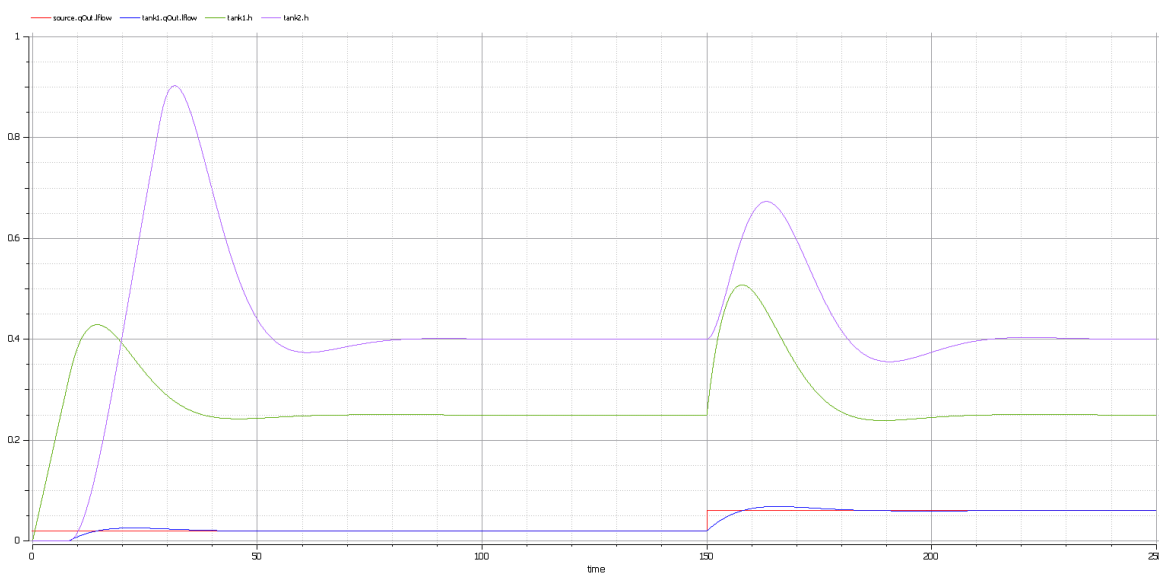
## Simulation SysML dans Modelica et Simulink

Enterprise Architect, outil leader Ingénierie des Systèmes, permet de construire des modèles à l'aide de techniques et de langages modélisation conformes aux normes de l'industrie pour la représentation de systèmes cybermécaniques. Ces modèles servent de dispositifs de communication entre les ingénieurs collaborateurs, les équipes de consultants et autres, mais peuvent également être utilisés pour générer des visualisations avancées à l'aide de langages modélisation standard de l'industrie utilisés par *OpenModelica* et *Simulink de MATLAB*.

Cet exemple montre la puissance d'Enterprise Architect dans l'exploitation des normes ouvertes existantes pour visualiser des solutions. Le diagramme représente deux réservoirs connectés ensemble et une source d'eau qui remplit le premier réservoir. Deux contrôleurs continus sont utilisés pour réguler le débit d'eau du premier réservoir vers le second, ainsi que le débit du second réservoir.

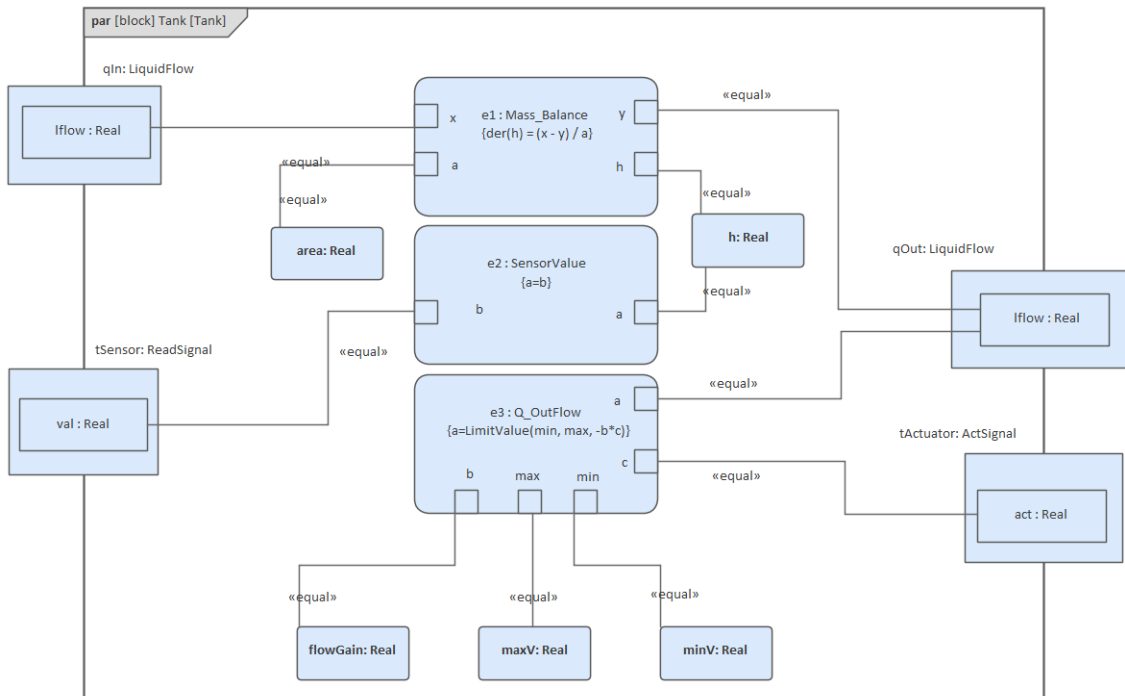


Un ensemble de diagrammes est créé dans Enterprise Architect qui modélise les aspects physiques des réservoirs et les équations (écrites dans le langage Modelica/MATLAB) qui caractérisent les flux entre les réservoirs. Une fois la simulation configurée et le bouton *Résoudre* sélectionné, le résultat ressemble à ce diagramme :



Nous aborderons les détails de la modélisation paramétrique et des contraintes dans une section ultérieure de ce sujet, et verrons comment les modèles que nous créons sont simplement des diagrammes de définition Bloc et diagrammes

paramétriques que nous avons étudiés dans une section précédente du guide. Ce diagramme Paramétriques montre un exemple de modélisation pour le problème des deux réservoirs, en utilisant des propriétés de contrainte et des connecteurs qui lient les paramètres dans un système d'équations. D'autres diagrammes sont nécessaires pour produire le résultat, mais c'est le diagramme principal qui montre la masse, les flux et le capteur déterminant le niveau dans le réservoir.



## Comment fonctionne Simulation SysML

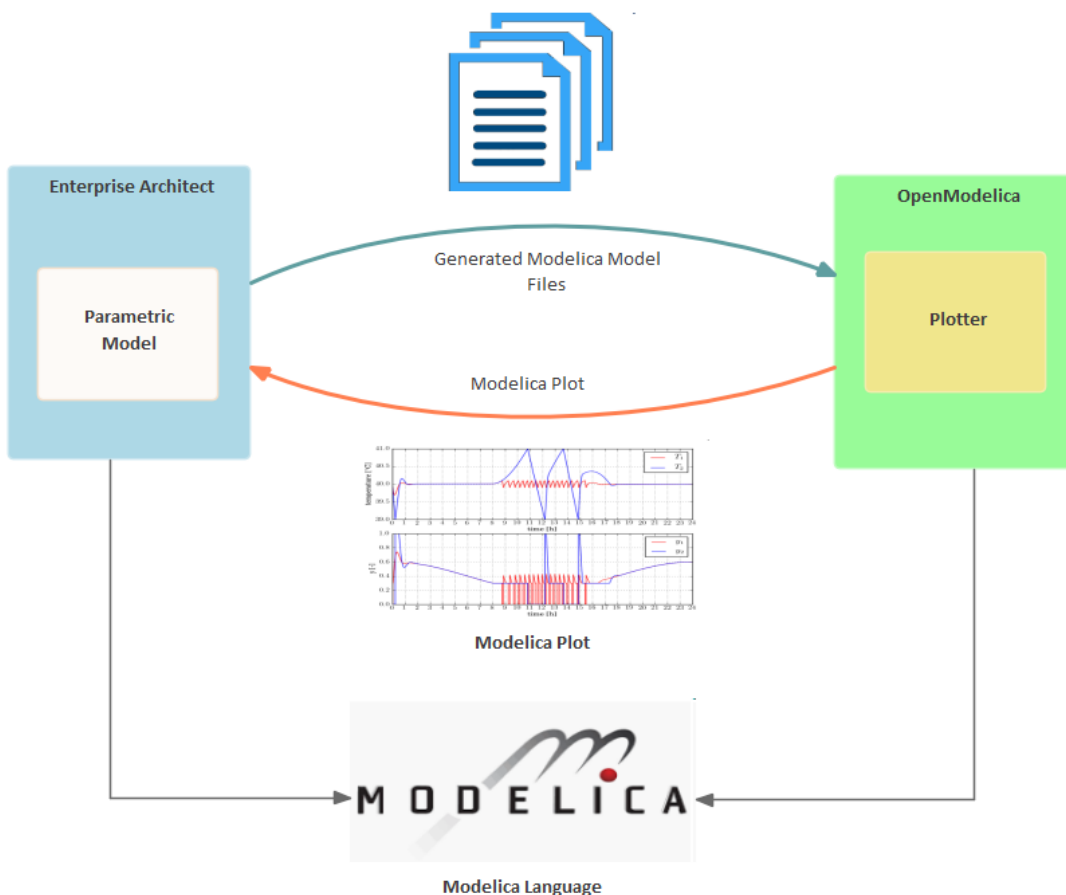
Enterprise Architect a été conçu sur la base de normes ouvertes et les équipes produit sont toujours conscientes de la puissance et de l'efficacité de l'utilisation des solutions existantes sans avoir à réinventer la roue. L'outil utilise ces normes pour permettre la visualisation :

- Systems Modeling Language (SysML) - géré par l' Object Management Group (OMG)
- OpenModelica - géré par le consortium Open Source Modelica (OSMC)
- Modelica - géré par l'Association Modelica

Nous avons consacré beaucoup de temps à formation des sujets précédents sur le SysML ; en fait, nous avons déjà appris la plupart de ce dont nous avons besoin pour créer les exemples de définition Bloc et Paramétriques pour ces visualisations. De plus, nous apprendrons comment ajouter des informations supplémentaires dont OpenModelica a besoin ; cela sera abordé dans la section suivante.

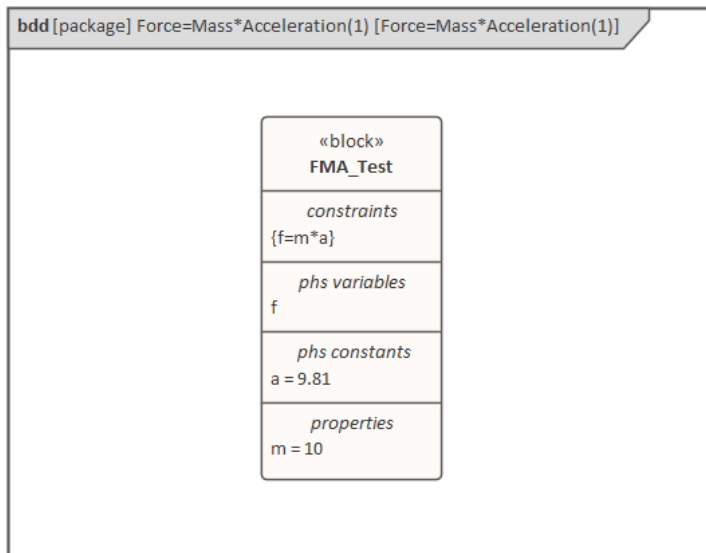
Modelica est un langage ouvert et orienté objet basé sur des équations, permettant la modélisation de systèmes cybermécaniques utilisant des sous-composants. Comme ses cousins mathématiques, Modelica est un langage inter-domaines qui a une grande variété d'applications, notamment dans les domaines mécanique, électrique, électronique, hydraulique, thermique, de contrôle, d'énergie électrique et orienté processus, pour ne citer que les sous-composants possibles d'un modèle Modelica et les types de systèmes qui peuvent être modélisés à l'aide de l'outil.

Enterprise Architect est capable d'effectuer des simulations de base pour diagrammes de processus et Statemachines , mais pour modélisation de systèmes cybermécaniques complexes, il utilise la puissance d'OpenModelica, soutenue par le langage Modelica lui-même pour faire le gros du travail. Enterprise Architect permet de relier ces modèles cybermécaniques à un large éventail d'autres systèmes et artefacts d'ingénierie logicielle, notamment les missions, les exigences des parties prenantes, Statemachines , le code de programmation, Tableaux Décision , les architectures, l'analyse des compromis et bien plus encore.

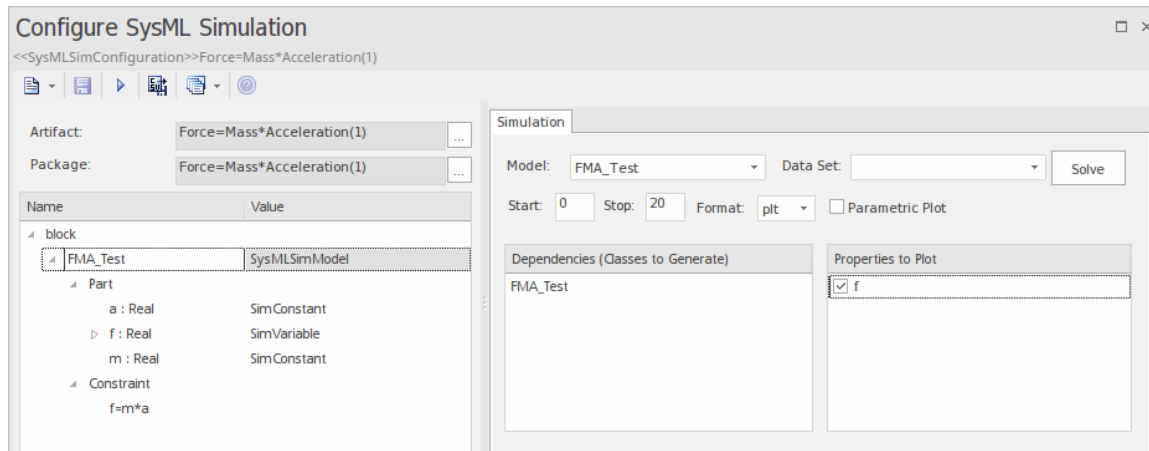


Le meilleur moyen de donner un aperçu de son fonctionnement est de prendre un exemple simplifié. Pour créer une visualisation simple de la deuxième loi de Newton - « Le taux de variation de l'impulsion est proportionnel à la force agissante et se produit dans la direction de cette force. » ( $F = m \cdot a$ ), un ingénieur utilisant Enterprise Architect devra :

1. Créez un diagramme Bloc décrivant les équations à l'aide de contraintes et de valeurs.



2. Configurez l'artefact de configuration SysMLSim (utilisé pour définir les informations nécessaires à OpenModelica).



3. Exécutez la Simulation en sélectionnant le bouton Résoudre dans la fenêtre.

Un graphique sera tracé avec  $f = 98,1$  (qui est le produit de la masse (10) et de l'accélération (9,81) exprimée dans l'équation  $[f = m \cdot a]$  avec la valeur à la place  $98,1 = 10 \cdot 9,81$ ). Il s'agit d'un exemple simpliste visant à montrer les ingrédients fondamentaux pour créer la visualisation ; nous examinerons des exemples plus avancés dans les sections suivantes montrant l'utilisation des Propriétés de contrainte et Ensembles de données définis par l'utilisateur.

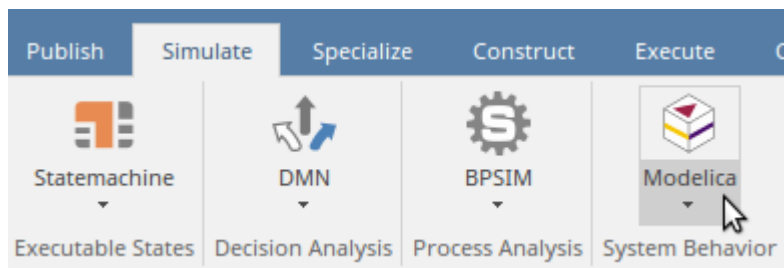
## Démarrage avec OpenModelica

Une façon simple de commencer à utiliser OpenModelica et les outils d' Enterprise Architect pour produire des simulations Paramétriques est de visualiser quelques exemples existants. C'est un outil formation utile pour toute fonctionnalité d' Enterprise Architect , mais il est particulièrement pertinent pour formation à la puissance d'OpenModelica car il y a un certain nombre de nouvelles choses à apprendre et cela se fait mieux avec quelques aides formation . Nous commencerons par examiner un exemple entièrement fonctionnel tiré du modèle Example, qui est distribué avec chaque installation d' Enterprise Architect et est disponible à partir de l' Item Aide du ruban Démarrer .

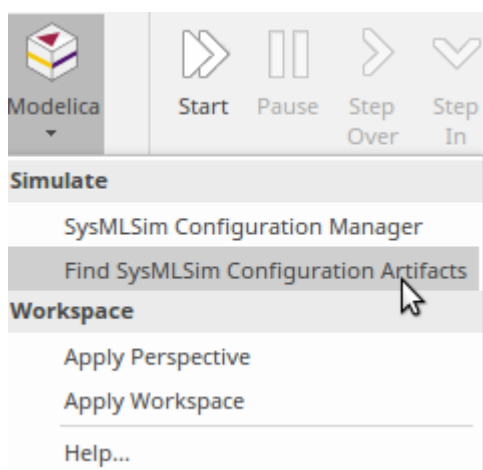
L'exemple que nous allons explorer est l'exemple du pendule, mais nous devons d'abord ouvrir le modèle d'exemple, ce que nous faisons en sélectionnant l'option du ruban :

Démarrer > Aide > Aide > Ressources > Ouvrir un exemple de Modèle

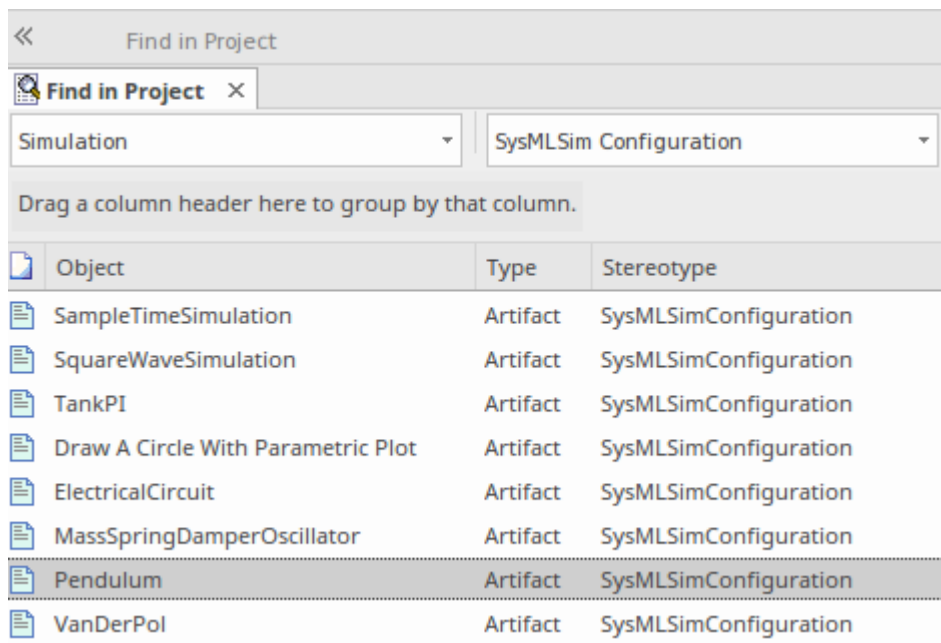
Les fonctionnalités d'OpenModelica sont regroupées de manière pratique dans un seul endroit avec d'autres fonctionnalités de simulation, et peuvent être lancées à l'aide du menu Modelica/Simulink disponible dans le panneau ' Comportement du Système ' du ruban ' Simuler ' . La facilité d'OpenModelica n'a rien à envier aux autres outils de simulation tels que Statemachine Exécutable , Décision Modélisation Notation et BPSim.



Avec le modèle chargé, nous pouvons utiliser les fonctionnalités OpenModelica dans Enterprise Architect pour localiser le pendule exemple dans le modèle. Il est courant qu'un modèle d'un système complexe soit très volumineux et qu'un nombre quelconque de simulations existantes soient configurées. Enterprise Architect fournit donc un mécanisme permettant de rechercher ces simulations.

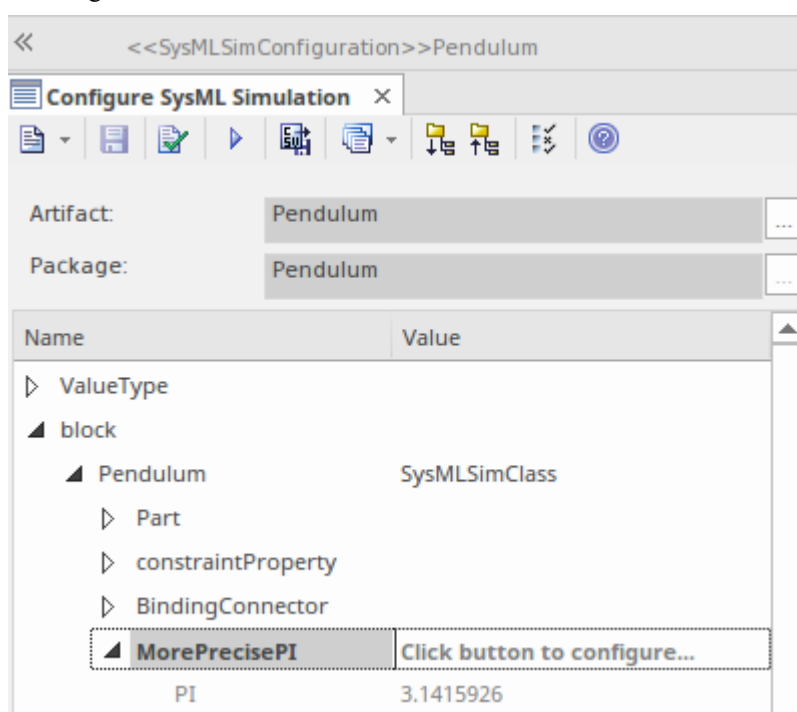


La sélection de cette option renvoie une liste d'artefacts de configuration de simulation SysML, qui sont les éléments dans lesquels les détails OpenModelica sont spécifiés. Dans cette liste, nous pouvons sélectionner l'exemple Pendulum, qui lancera une fenêtre qui peut être utilisée pour configurer et résoudre le problème codifié dans l'exemple. Pour pouvoir exécuter la simulation, une version d'OpenModelica doit être installée sur la machine qui exécute Enterprise Architect . Les détails sur la façon d'installer OpenModelica sont contenus dans la section suivante.



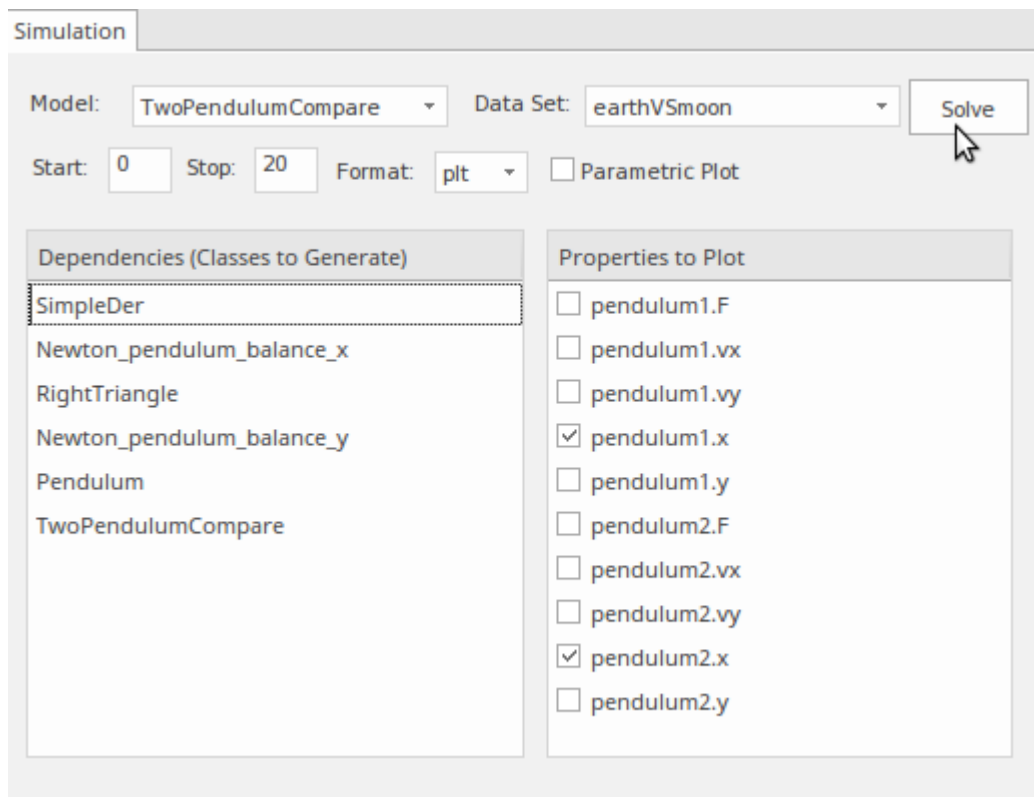
La fenêtre est divisée en deux sections :

- Le panneau de configuration (à gauche) définit les types de valeur, les blocs et Propriétés de contrainte, qui sont tous définis dans le référentiel et - comme nous le verrons plus tard - peuvent être visualisés dans la fenêtre Navigateur et diagrammes

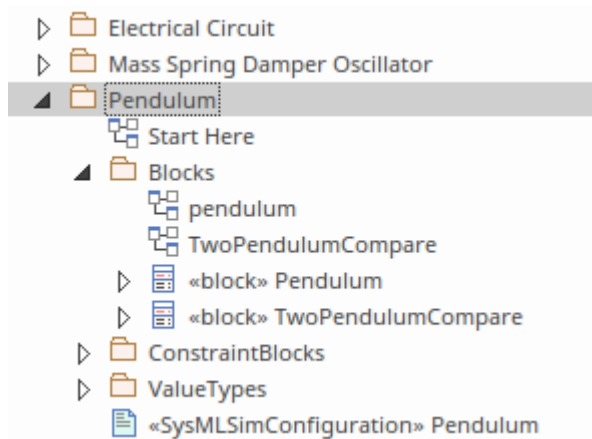


- Le panneau de simulation (à droite) permet de sélectionner et de spécifier les options qui affecteront la simulation, telles que l'ensemble de données, le temps d'attente et la durée de la simulation, les types de fichiers de sortie, etc.





Les informations de la fenêtre sont automatiquement renseignées à partir du modèle visible dans la fenêtre Navigateur , et l'emplacement des éléments - y compris les types de valeur, les blocs et Propriétés de contrainte - peut être trouvé à l'aide de l'option « Rechercher dans Projet Navigateur » du menu contextuel. Il existe également une option analogue pour rechercher l'élément sélectionné dans tous diagrammes dans lesquels il apparaît.



## Installation d'OpenModelica

Enterprise Architect utilise la puissance de la plateforme OpenModelica. Ainsi, lorsque vous exécutez une simulation à partir de la fenêtre Simulation , elle fait appel à OpenModelica (installé sur la même machine) pour effectuer le gros du travail et renvoyer les résultats de la simulation. Cela permet à Enterprise Architect de tirer parti de la puissance de cet outil ouvert et de tous les esprits brillants qui ont contribué à son excellence. Il existe des versions Windows et Linux d'OpenModelica et vous devrez installer celle qui convient à votre environnement. Les étapes sont résumées ici.

1. Téléchargez le logiciel OpenModelica ( Windows ou Linux).
2. Installez le logiciel.
3. Vérifiez l'installation.
4. Configurez le Solveur en spécifiant le chemin dans Enterprise Architect .

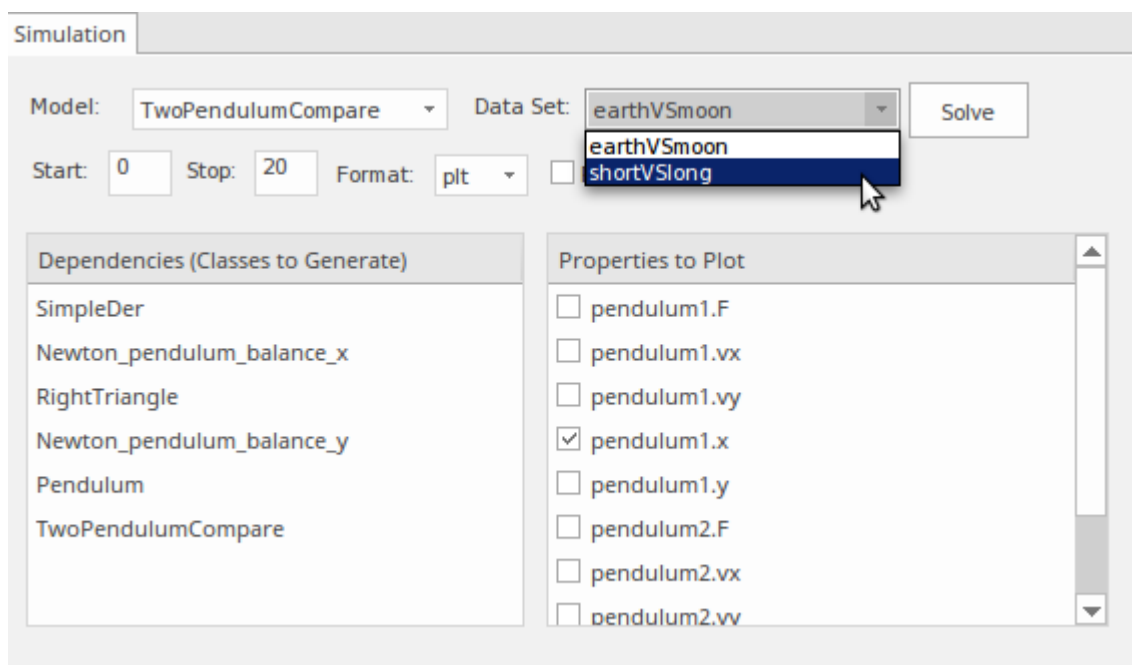
Tous les détails de l'installation et de la configuration peuvent être trouvés dans la rubrique d'aide [OpenModelica Integration](#).

## Simulation avec Ensembles de données

Simulation est un aspect important de l'analyse technique et constitue un moyen utile et rentable d'analyser le comportement d'un système. Le système peut avoir déjà été construit ou être en cours de développement. Dans les deux cas, la simulation peut être utilisée pour visualiser les considérations de conception des composants ou du système, l'optimisation des performances, l'ingénierie de sécurité et bien plus encore. Pour fournir cette capacité, il est important de pouvoir exécuter une simulation en utilisant différentes valeurs pour les variables et les constantes. Par exemple, dans notre exemple de pendule, nous pourrions vouloir analyser les performances du système sur différentes planètes, pour examiner, par exemple :

- L'effet de la force gravitationnelle terrestre par rapport à la force gravitationnelle lunaire (ou à celle d'une autre planète)
- L'effet des différentes longueurs string
- L'effet des différentes masses
- L'effet des différents points de départ
- Toute combinaison des éléments ci-dessus

Enterprise Architect fournit un ensemble de données facilité pour les simulations qui peuvent être appliquées au niveau Bloc . En utilisant l'onglet « Simulation » (panneau de droite) de la fenêtre de configuration SysMLSim, nous pouvons sélectionner des ensembles de données prédéfinis.



Vous pouvez définir un nombre illimité d'ensembles de données et les ajouter au Bloc approprié dans le panneau « Configuration » (à gauche) de la fenêtre Simulation . Ceux-ci sont disponibles sous forme de nœuds sous chaque Bloc et il existe une option permettant d'afficher et de modifier les valeurs de données dans une fenêtre en utilisant le bouton Parcourir [...] sur la ligne des ensembles de données.

Name	Value
Value Type	
block	
Pendulum	SysMLSimClass
TwoPendulumCompare	SysMLSimModel
Part	
earthVSmoon	Click button to configure...
pendulum2.g	1.6
shortVSLong	Click button to configure...
pendulum2.L	0.8
pendulum2.x.start	0.8
constraintBlock	

Une fois sélectionnée, la fenêtre Configurer les données Simulation s'ouvre et permet de visualiser et de modifier les valeurs, de les importer ou de les exporter. Ce mécanisme signifie que les machines de simulation peuvent être réutilisées dans de nombreux contextes différents et que les organisations d'ingénierie qui se concentrent sur des types particuliers de problèmes peuvent créer des bibliothèques de simulations qui pourraient être réutilisées dans une multitude de contextes et de types de problèmes d'ingénierie. La fenêtre illustrée ici montre un ensemble de données qui contient des valeurs relatives aux deux problèmes de pendule que nous avons examinés et nous pouvons voir à titre d'exemple que l'accélération due à la gravité lunaire a été définie comme une approximation de 1,6 m/s<sup>2</sup>, soit environ 16,6 % de la valeur à la surface de la Terre. Cette simulation pourrait être réutilisée avec des ensembles de données différents applicables à Mars ou à Jupiter ou dans un exemple plus terrestre avec une masse ou une longueur de string différente.

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		

# Créer des modèles pour Simulation

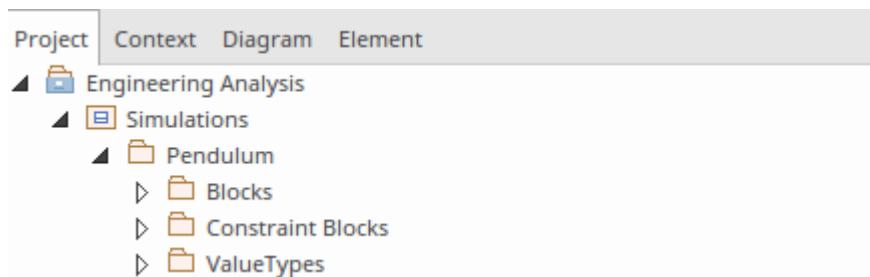
L'exemple que nous avons étudié dans la section *Fonctionnement* de ce guide était volontairement trivial. Dans cette section, nous examinerons ces étapes plus en détail, y compris les options de configuration des modèles. Enterprise Architect offre une grande flexibilité et permet de définir des modèles simples à usage unique ou de créer des modèles polyvalents sophistiqués basés sur une bibliothèque qui peuvent être réutilisés dans plusieurs projets et domaines à l'aide de différents ensembles de données et contextes. Dans cette rubrique, nous explorerons certaines de ces options de configuration des modèles afin qu'ils soient adaptés à l'objectif et qu'ils génèrent les résultats d'ingénierie requis.

## Création d'un Paquetage Modèle de Simulation

Un Paquetage Modèle peut être configuré à n'importe quel endroit de la fenêtre Navigateur, mais il est généralement considéré comme une bonne pratique de configurer un Paquetage *Simulations* sous chaque projet ou initiative. Celui-ci peut alors contenir des sous-paquets pour chaque simulation. Il est prévu que pour un projet donné, un certain nombre de simulations différentes puissent être nécessaires. Il est également probable, comme indiqué dans la section Ensembles de données, qu'une organisation souhaite réutiliser certaines des simulations dans plusieurs projets. Ces simulations peuvent être configurées dans le Navigateur de projet et définies à un niveau supra-projet, par exemple au niveau d'une entreprise, d'une organisation ou d'un service d'ingénierie. Elles peuvent ensuite être incluses dans un diagramme au niveau du projet, indiquant qu'elles sont applicables à un projet ou à un contexte de problème donné. Il est également probable qu'une équipe d'ingénierie souhaite réutiliser les Types de valeur et leurs bibliothèques de quantités et d'unités concomitantes entre les projets, et ceux-ci, comme indiqué dans une rubrique précédente du guide, sont mieux définis et modélisés à un niveau supra-projet.

Enterprise Architect utilise le mécanisme d'importation Paquetage SysML pour garantir que les types de valeur définis au niveau de l'entreprise peuvent être inclus et réutilisés à chaque niveau Paquetage Simulation. La structure contient généralement les Paquetages suivants :

1. Types de valeur (spécifiques à ce projet)
2. Blocs
3. Propriétés



Dans la section suivante, nous apprendrons à créer et configurer l'artefact Simulation SysML, stéréotypé comme <<SysMLSimConfiguration>>.

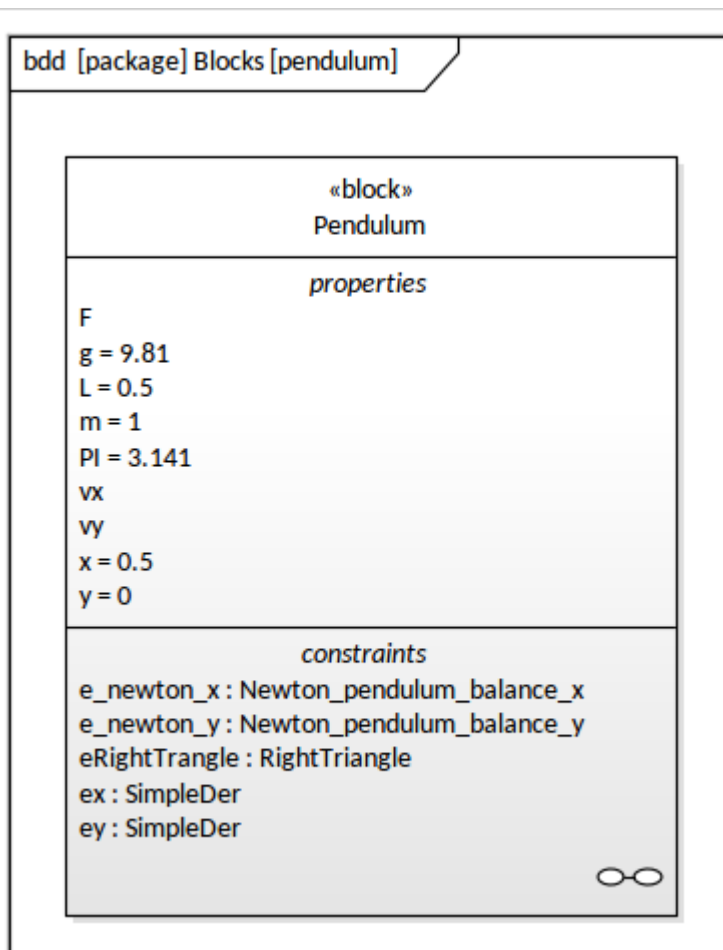
## Créer les éléments Simulation

L'effort principal dans la préparation d'une simulation réside dans la définition des éléments du modèle, en utilisant le SysML avec le niveau de précision approprié pour permettre à la plateforme OpenModelica d'exécuter la simulation. Il existe plusieurs façons de définir les modèles, et dans ce guide, nous nous concentrerons sur la méthode la plus robuste et la plus flexible, car c'est celle qui sera utilisée par la plupart des ingénieurs et des équipes en exercice.

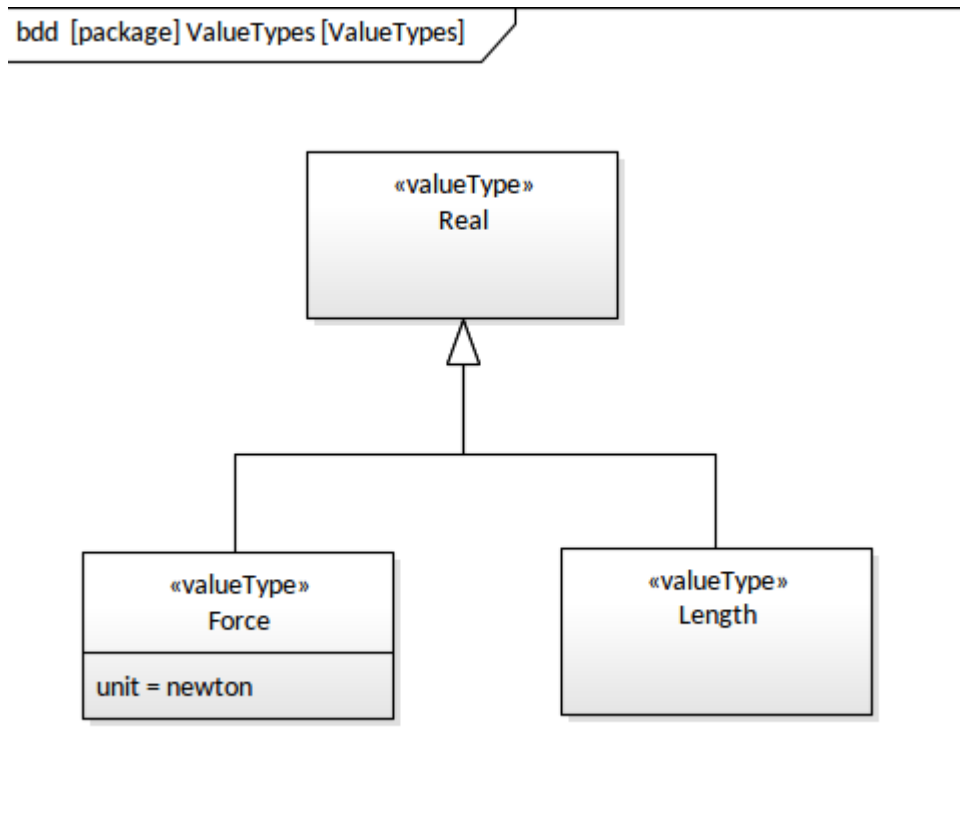
À ce stade, il est utile d'aborder le sujet de la simulation pour s'assurer que nous comprenons sa pertinence et sa connexion avec d'autres parties du modèle, et comment les éléments de simulation peuvent être connectés à d'autres

éléments du modèle tels que Exigences , les cas Test et plus encore. En règle générale, les simulations sont utilisées comme un moyen d'étudier un problème cyber-physique sans avoir à construire un modèle physique long et souvent coûteux. La simulation peut faire partie d'une analyse de problème, d'une analyse de compromis, d'une analyse de performance ou d'un certain nombre d'autres investigations. Les blocs définis dans le cadre de la simulation peuvent être alloué à des éléments comportementaux et finalement à Exigences .

Les premiers éléments à créer sont les blocs, qui sont les éléments structurels fondamentaux de la solution. Nous avons appris à le faire dans un exemple précédent ; ce diagramme montre un certain nombre de compartiments qui contiennent des éléments, à savoir les propriétés.



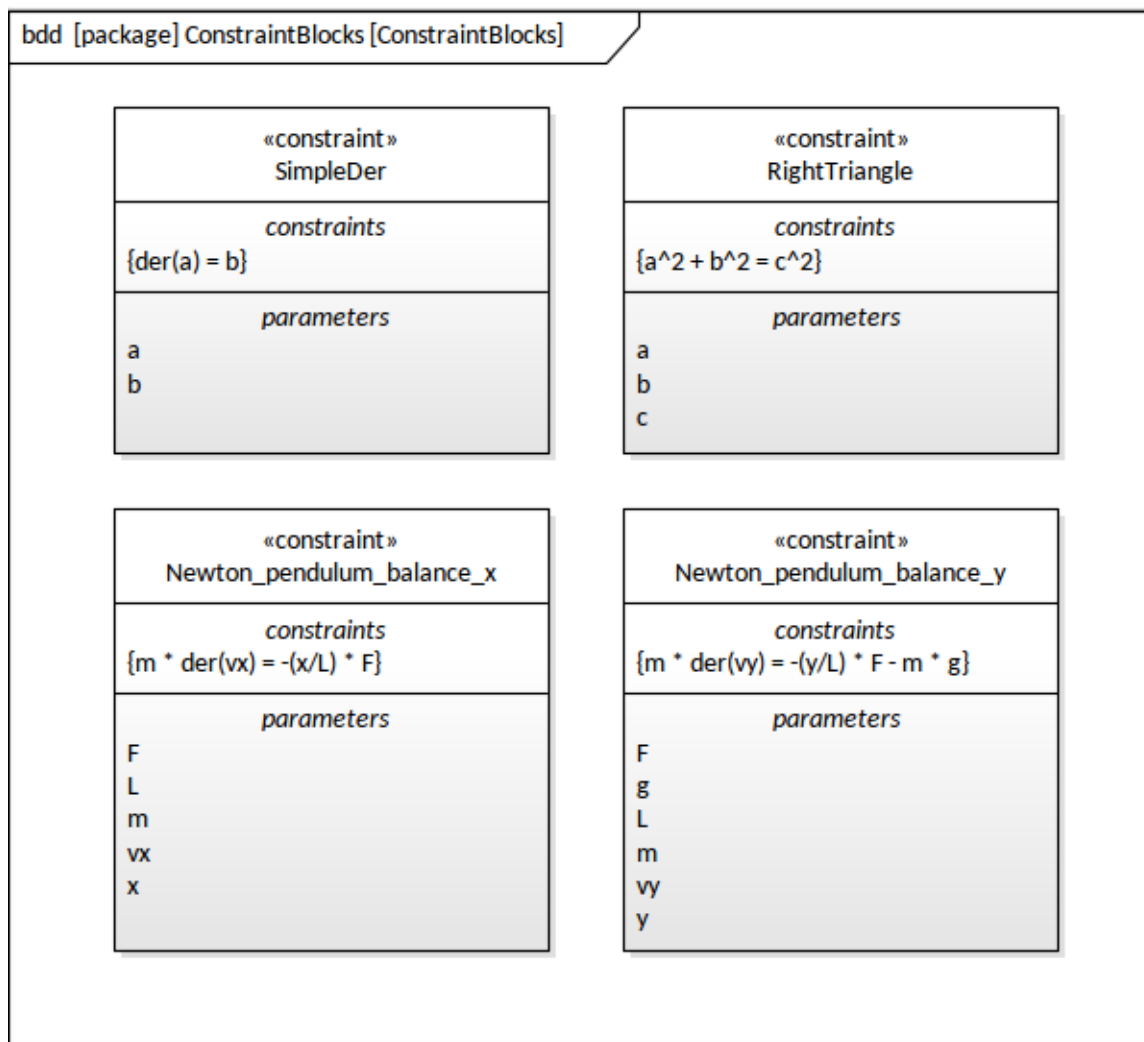
En préparation de la définition des équations qui définissent le comportement du système pendulaire, nous allons définir les types valeur nécessaires pour garantir la précision du modèle et vérifier que les paramètres de simulation sont correctement spécifiés. Cela se fait à l'aide d'un diagramme de définition Bloc (bdd), en utilisant l'élément Type de valeur disponible dans la boîte à outils Diagramme .



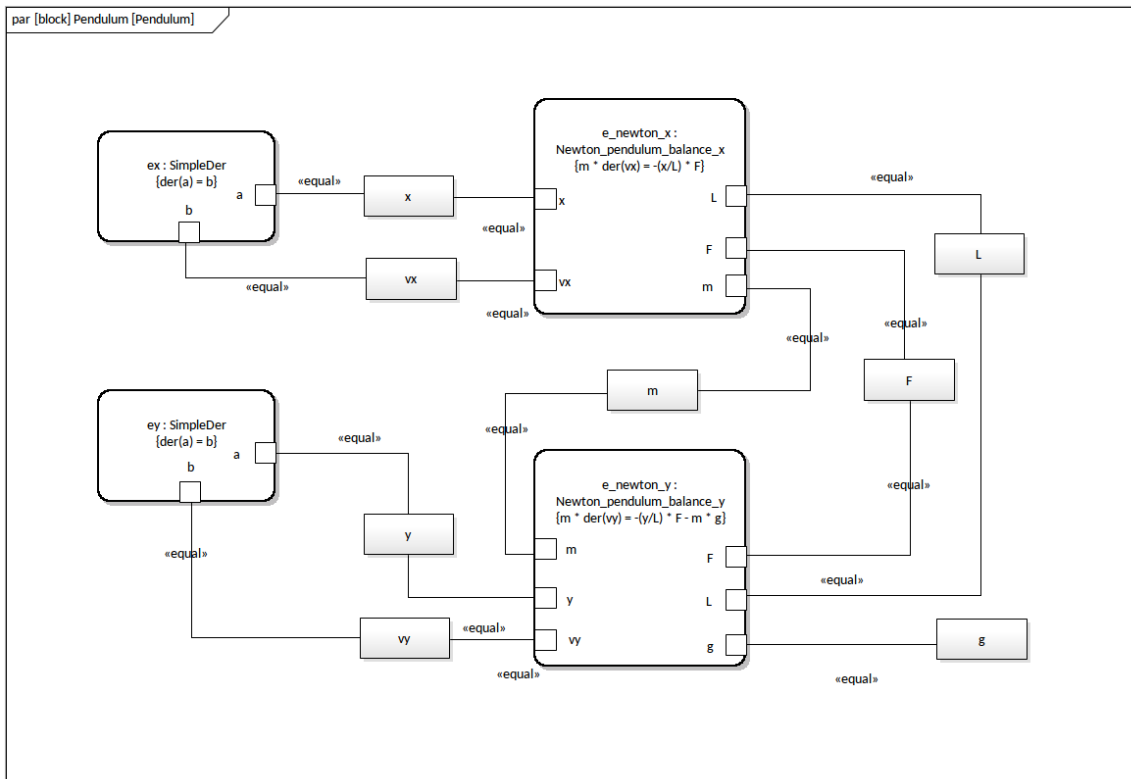
L'étape suivante consiste à définir les contraintes à l'aide de ConstraintBlocks. Ces éléments nous permettent de définir les équations Modelica qui régiront le comportement du système modélisé, dans notre cas, un pendule. Comme décrit précédemment dans ce guide, les ConstraintBlocks sont définis sur un diagramme de définition Bloc et comportent une série de paramètres définis et une contrainte qui exprime ces paramètres dans une équation écrite dans Modelica. Par exemple, l'équation qui contraint l'aspect vertical du pendule s'écrit comme suit :

$$m * \text{der}(vy) = -(y/L) * F - m * g$$

Notez les mots-clés de Modelica tels que « der » qui signifie une dérivée du premier ordre. L est la longueur (paramètre) du pendule, g est l'accélération due à la gravité (constante), m est la masse (paramètre) du pendule, x et y sont les coordonnées dans l'espace bidimensionnel et F est la force. Notez que Modelica utilise :



L'étape suivante consiste à créer les diagrammes Paramétriques qui lient les équations entre elles. Comme indiqué précédemment dans le Guide, ces diagrammes sont un type spécialisé de diagramme Bloc internes et contiennent des instances de ConstraintBlocks appelées ConstraintProperties qui exposent leurs paramètres, qui sont liés par des connecteurs à des paramètres sur d'autres ConstraintProperties.

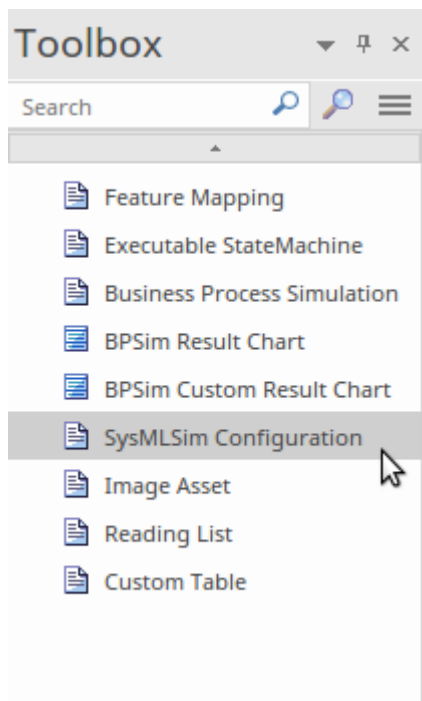


## Spécification d'un artefact de configuration


L'artefact Simulation SysML est l'élément qui lie les éléments du modèle à la plateforme OpenModelica. Les éléments du modèle exprimés dans SysML sous la forme de blocs, de blocs de contrainte et de leurs propriétés de contrainte associées liés entre eux sur diagrammes Paramétriques apparaissent dans la fenêtre Simulation et peuvent être configurés avec d'autres paramètres pour piloter la simulation.

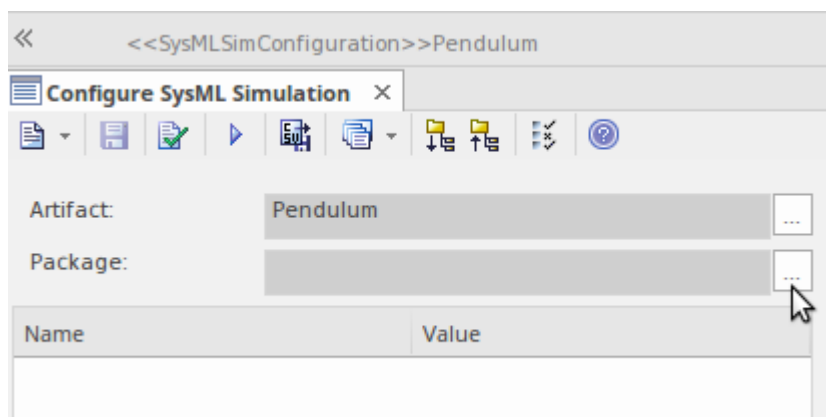
La première étape de la configuration de ce facilité consiste à créer un élément de configuration SysMLSim, disponible à partir de la page de la boîte à outils « Artefact », comme indiqué dans cette capture d'écran.



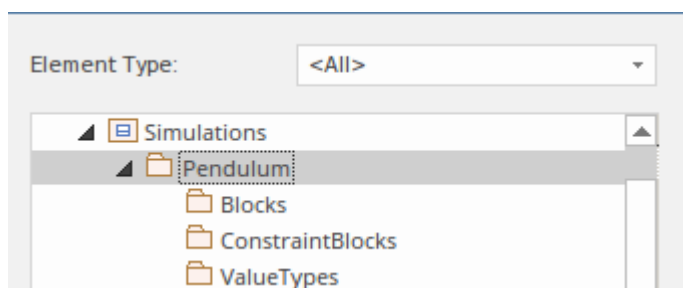


Enterprise Architect effectue la majeure partie du travail de remplissage de la fenêtre Configurer la Simulation SysML.

L'ingénieur doit simplement sélectionner le Paquetage à l'aide de l'icône  du panneau « Configuration » de la fenêtre Artefact Simulation .



Enterprise Architect affiche la fenêtre de sélection Paquetage et, une fois le Paquetage sélectionné, le panneau « Configuration » est renseigné avec les Blocs, les Blocs de Contraintes et les Types de Valeur du modèle. À partir de ce point, les valeurs peuvent être saisies pour divers paramètres ou des ensembles de données peuvent être définis. Les paramètres de configuration de la simulation peuvent être saisis et la simulation est prête à être exécuter .

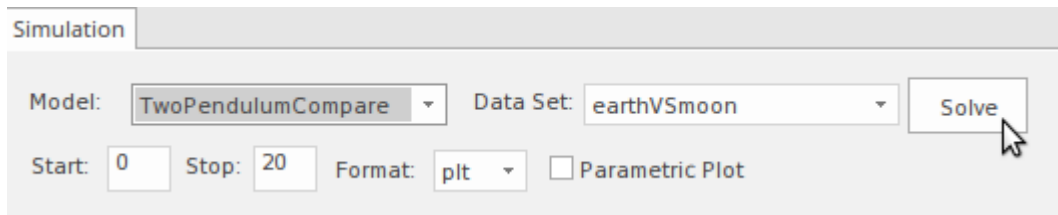


Une fois les données saisies pour la simulation, y compris les valeurs Démarrer et Arrêter et les formats de sortie, la Simulation peut être exécuter en sélectionnant le bouton Résoudre comme indiqué dans cette illustration d'écran.

Simulation

Model:  Data Set:

Start:  Stop:  Format:   Parametric Plot



## Exemple Modèle SysML

Le langage Modélisation système (SysML), comme tout autre langage, doit être appris. Nous n'avons pas la chance d'avoir grandi en entendant nos parents parler cette langue à la maison, mais un certain nombre de lecteurs connaissent peut-être déjà cette langue pour diverses raisons, notamment :

- Il a été utilisé par des collègues ou des partenaires dans des projets
- Il a été enseigné dans le cadre d'un cours universitaire
- Vous avez suivi une formation ou lu ou consulté une documentation
- Vous avez pris 6 semaines de congé et avez lu le cahier des charges de bout en bout

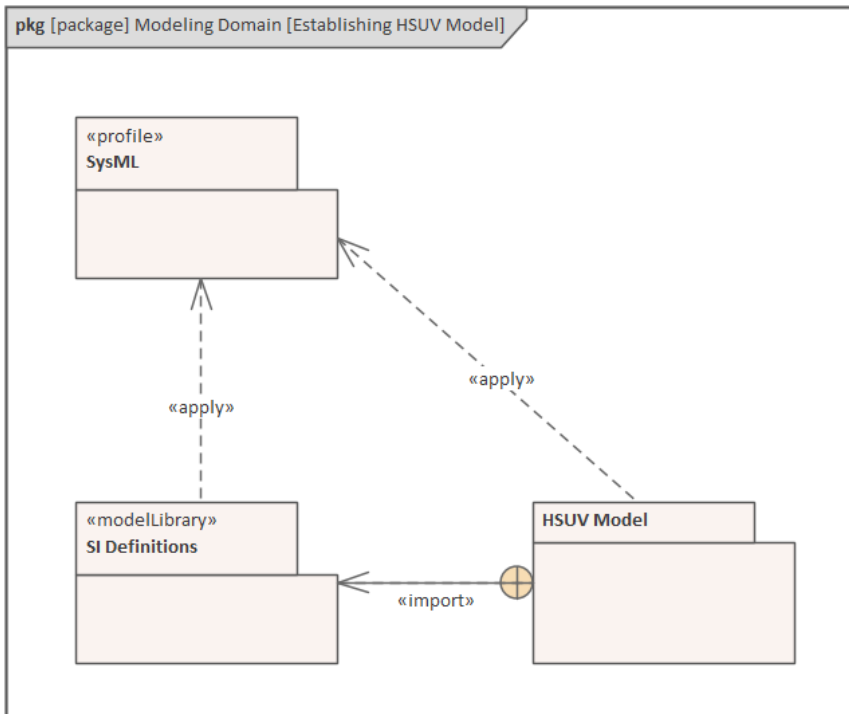
Si la dernière hypothèse est vraie, vous aurez probablement une connaissance approfondie du langage, mais vous aurez peut-être besoin de quelques congés récréatifs car il s'agit d'un document assez dense et vous aurez également dû plonger vos orteils dans la spécification Unified Modeling Language . Il est plus probable qu'un plus grand nombre de lecteurs n'auront que peu ou pas d'exposition au langage, et ce premier exemple est destiné à vous donner une vue rapide et de haut niveau de ce à quoi vous pouvez vous attendre lorsque vous travaillez dans Enterprise Architect pour modéliser un projet Ingénierie Systèmes Modèles Basée à l'aide du SysML. Il est basé sur l'exemple d'un véhicule hybride qui apparaît dans l'annexe du problème d'échantillon de la spécification

## Présentation Paquetage (Structure de l'exemple Modèle )

Le diagramme Paquetage montre une manière de visualiser le contenu du référentiel. Lorsque le contenu diagramme est visualisé dans la fenêtre Navigateur d' Enterprise Architect , la structure peut être parcourue. Il existe également des relations structurelles et d'espace de noms importantes qui peuvent être observées sur diagrammes Paquetage et qui aident à clarifier les relations de haut niveau importantes entre les groupes d'éléments dans le référentiel.

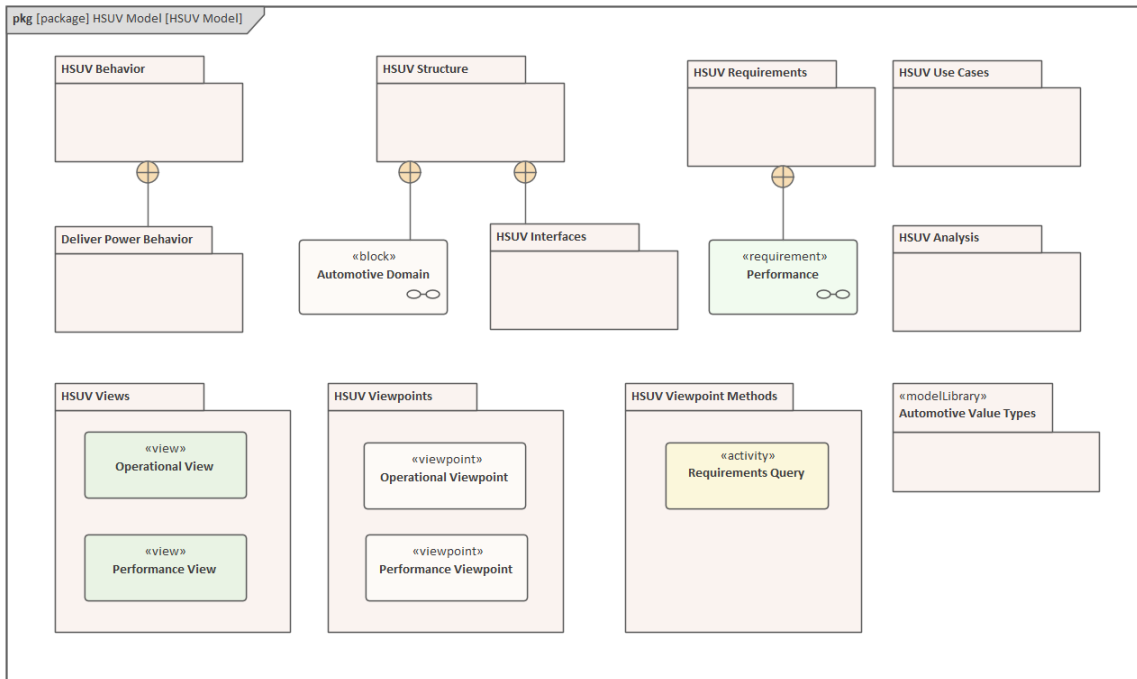
## Diagramme Paquetage - Application du profil SysML

Comme le montre ce diagramme, le HSUV Modèle est un Paquetage qui représente le modèle utilisateur. Le profil SysML est appliqué à ce Paquetage afin d'inclure les stéréotypes du profil. Le HSUVModel peut également nécessiter des bibliothèques de modèles, telles que la bibliothèque de modèles Types d'unités SI. Les bibliothèques de modèles sont importées dans le modèle utilisateur comme indiqué.

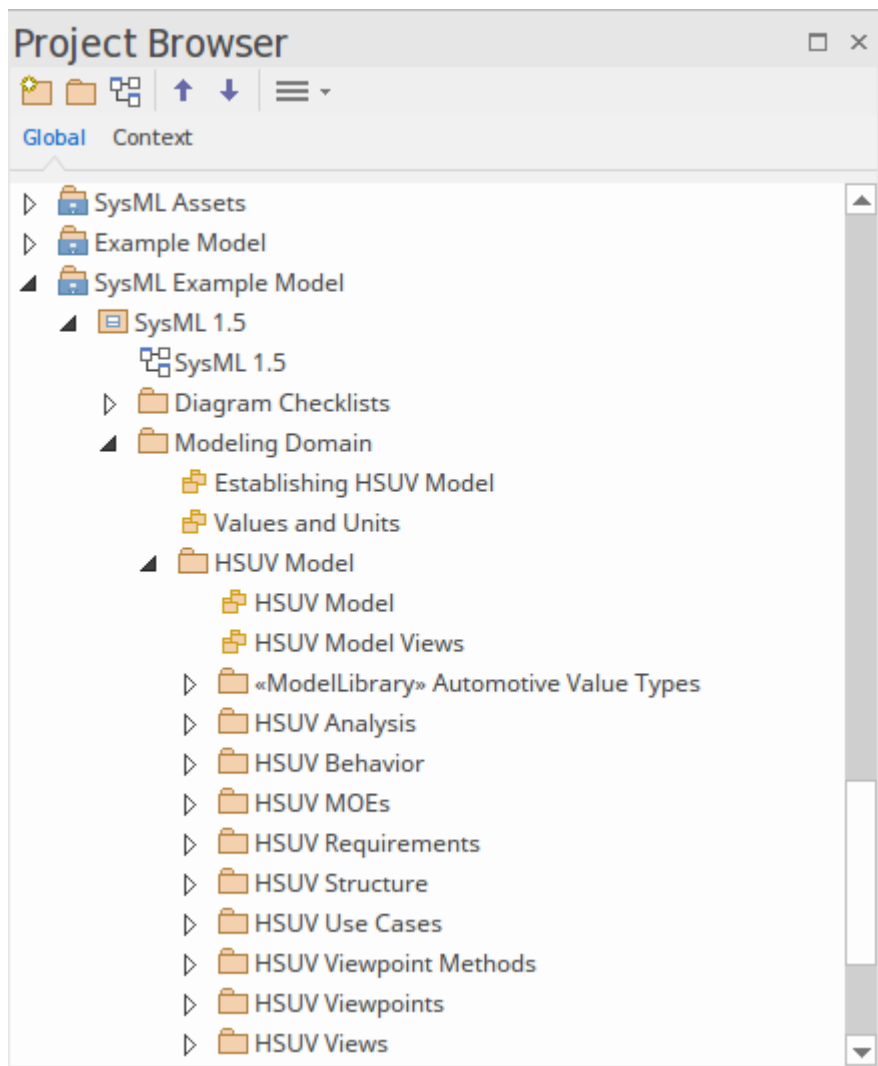


# Diagramme Paquetage - Affichage de la structure Paquetage du Modèle

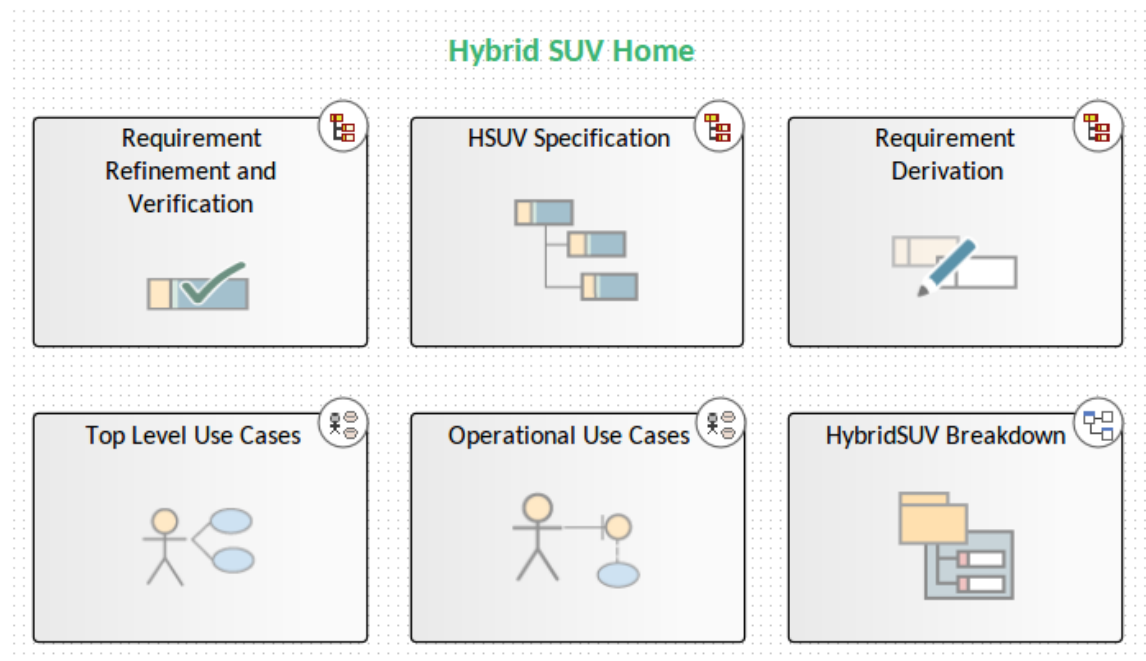
Ce diagramme Paquetage montre la structure du modèle utilisé pour définir le système HybridSUV. Le diagramme fournit un moyen utile de visualiser les conteneurs de haut niveau ( Paquetages ) utilisés pour structurer le référentiel et définir la structure, le comportement et les exigences du système HybridSUV.



Les relations entre la Vue Opérationnelle et la Vue Performance et le reste du modèle utilisateur sont exprimées explicitement à l'aide de la relation « import ». Les Paquetages qui apparaissent dans le diagramme sont définis et peuvent également être visualisés dans une vue hiérarchique à l'aide de la fenêtre Navigateur .



Bien que la fenêtre Navigateur offre un mécanisme important pour naviguer dans le référentiel, il existe une large gamme d'autres vues, notamment, dans ce cas, un diagramme . Enterprise Architect fournit également un moyen pratique de créer diagrammes définis par l'utilisateur qui peuvent servir de moyen alternatif de navigation dans le référentiel. Ce mécanisme permet aux ingénieurs système et à d'autres de créer un nombre quelconque de cellules de navigation pour fournir un accès personnalisé au modèle, évitant ainsi à l'utilisateur d'avoir à savoir ou à comprendre comment parcourir le modèle.



Ceux-ci sont particulièrement utiles lors de la visualisation des modèles via un Navigateur Web.

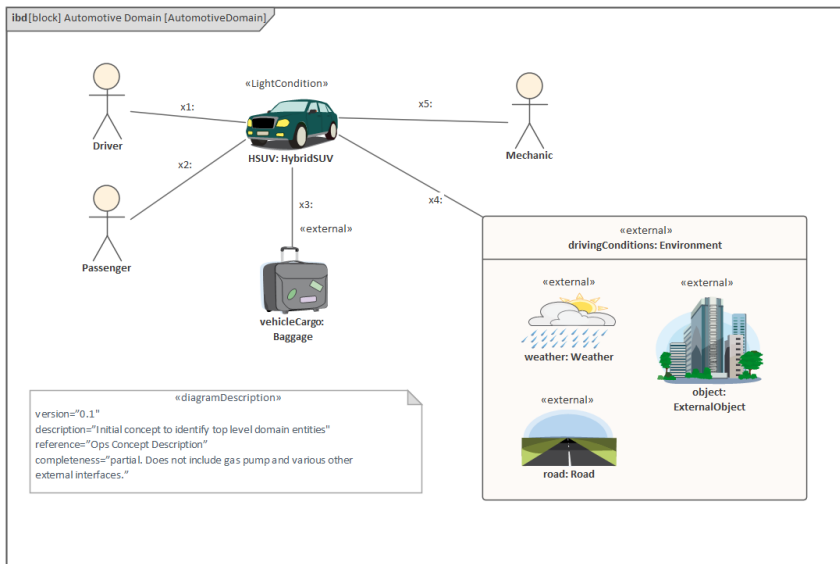


## Définir le contexte (limites et cas d'utilisation)

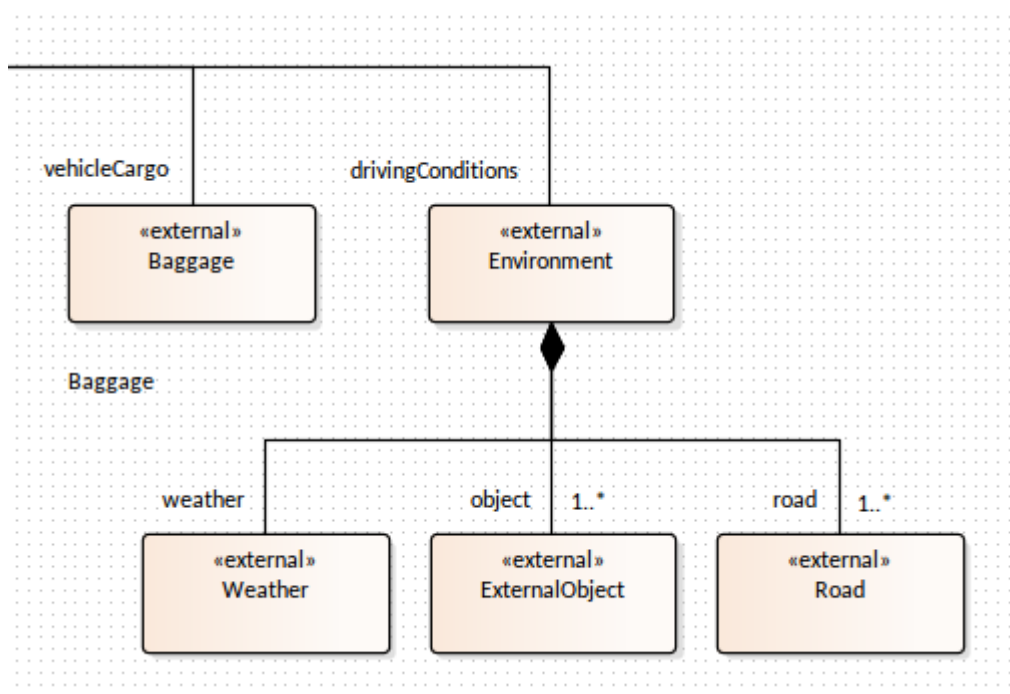
Le contexte d'un système est essentiel pour que toutes les parties prenantes puissent comprendre le système dans son environnement. Le diagramme de cas d'utilisation est l'un des diagrammes les plus simples mais les plus descriptifs de la boîte à outils SysML. Sa puissance réside dans le fait qu'il relie les entités qui résident en dehors du système (acteurs) aux avantages qu'elles souhaitent obtenir du système (cas d'utilisation) sans articuler la manière dont le système fournira la valeur. Les cas d'utilisation peuvent être rédigés à un niveau descriptif, mais si des détails supplémentaires sont nécessaires, le générateur de scénarios d'Enterprise Architect peut être utilisé pour spécifier les étapes de chaque scénario, dans un outil qui simplifie la documentation des cas d'utilisation. diagrammes de comportement tels que diagrammes d'activité et les documents de cas d'utilisation peuvent également être générés automatiquement à partir de l'outil.

# Modèle de domaine opérationnel - Définition du contexte

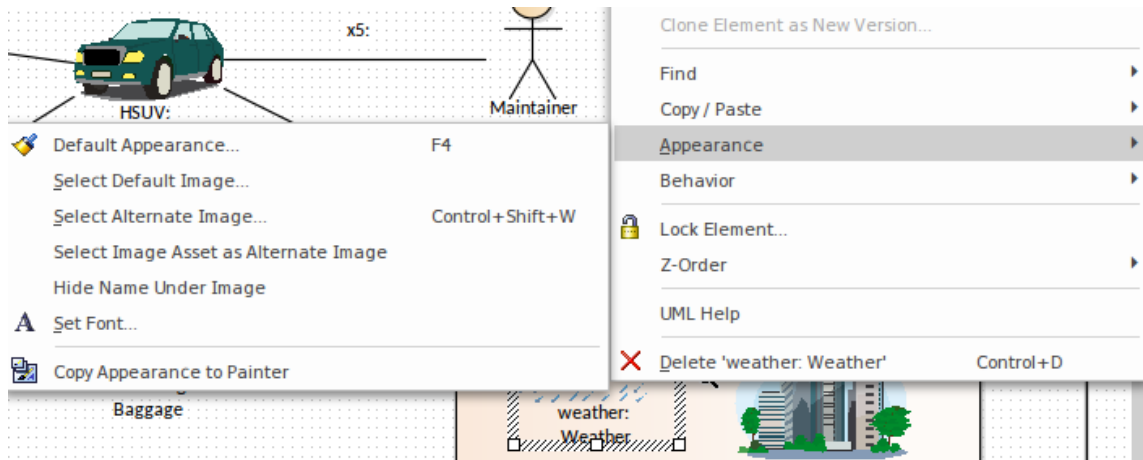
Ce diagramme Bloc internes illustre une utilisation définie par l'utilisateur du type diagramme , qui décrit le système dans le contexte de son environnement. Les stéréotypes « système » et « externe » sont définis par l'utilisateur et ne sont pas spécifiés dans SysML, mais aident l'ingénieur système à décrire le système d'intérêt par rapport à son environnement.



Enterprise Architect permet également de remplacer les symboles conventionnels du SysML par des images plus attrayantes et significatives qui contribuent à l'acceptation du diagramme par des publics non techniques.

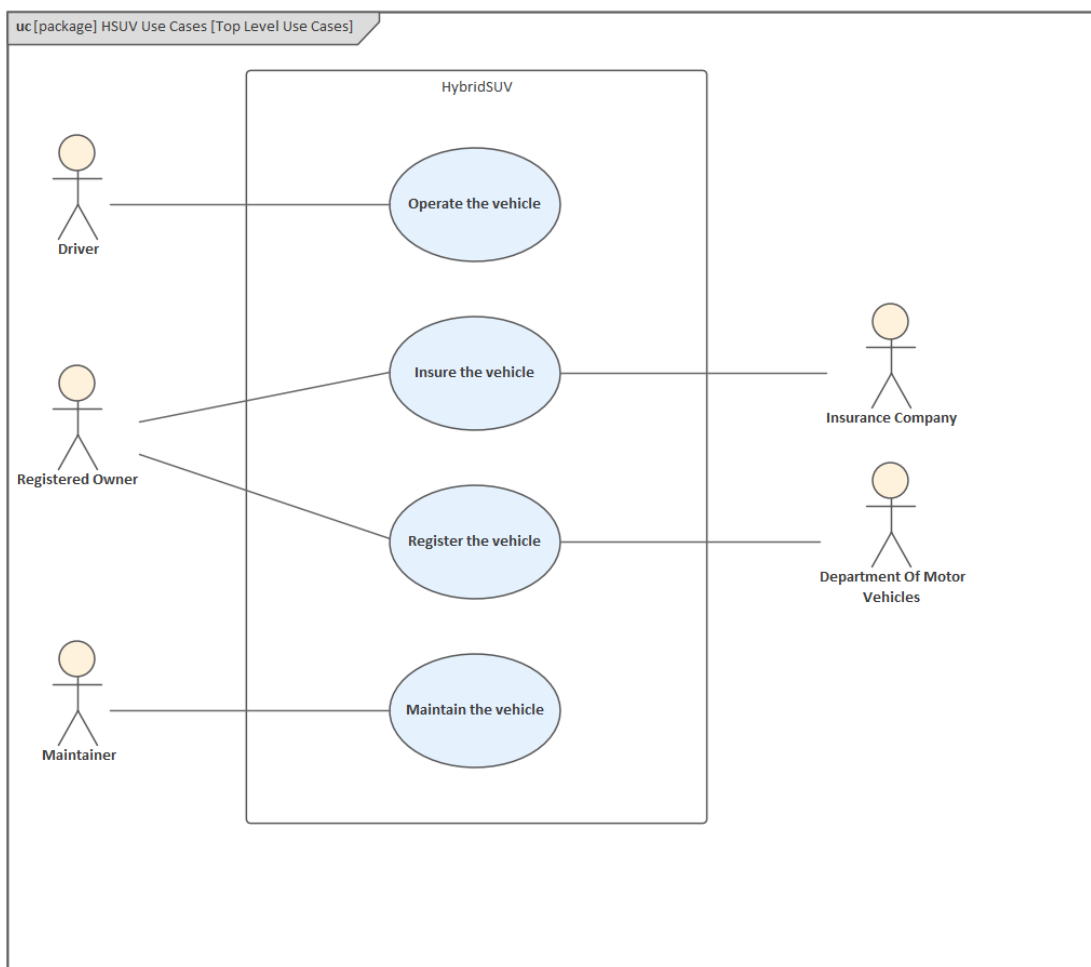


Ce changement d'apparence d'élément peut être appliqué à un niveau global (par défaut) ou à un niveau spécifique diagramme permettant de créer des présentations alternatives pour différents publics.

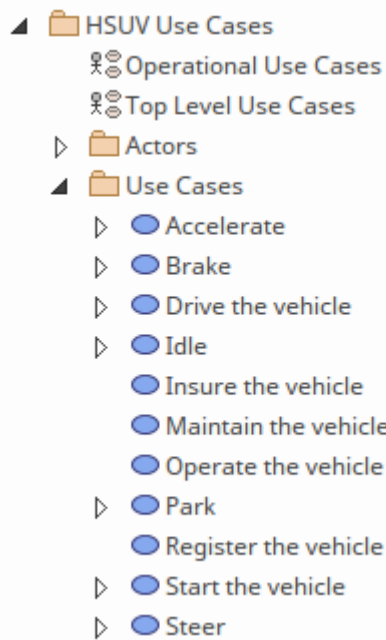


## Diagramme de cas d'utilisation - Cas d'utilisation de niveau supérieur

Le diagramme de cas d'utilisation décrit le SUV hybride comme un système utilisé par un certain nombre de rôles externes. Le diagramme permet de définir le contexte de tous les rôles (humains et autres systèmes) qui interagissent avec le système ou en tirent valeur. Le diagramme de cas d'utilisation, bien qu'il paraisse simpliste, fournit un mécanisme permettant de garantir que toutes les interactions potentielles du système sont définies et comprises. Le système lui-même est représenté dans le diagramme par un élément Bordure, qui agit comme conteneur pour les cas d'utilisation, les acteurs se trouvant à l'extérieur de la Bordure. Dans ce diagramme il existe un certain nombre de rôles externes - autres que le conducteur - qui interagiront avec le système SUV hybride, notamment le propriétaire enregistré, le responsable de la maintenance, la compagnie d'assurance et le service des véhicules automobiles.



Les cas d'utilisation apparaissent dans la fenêtre Navigateur et peuvent être facilement regroupés en acteurs et cas d'utilisation. Il est alors possible de définir un nombre quelconque de diagrammes de cas d'utilisation permettant à l'ingénieur système de visualiser les cas d'utilisation.



Enterprise Architect propose également un certain nombre d'outils utiles et uniques pour aider l'ingénieur système à décrire efficacement les cas d'utilisation et à définir des scénarios qui détaillent les étapes représentant l'interaction entre l'acteur et le système. Une fois ceux-ci définis, l'outil peut générer automatiquement diagrammes comportementaux directement à partir du modèle.

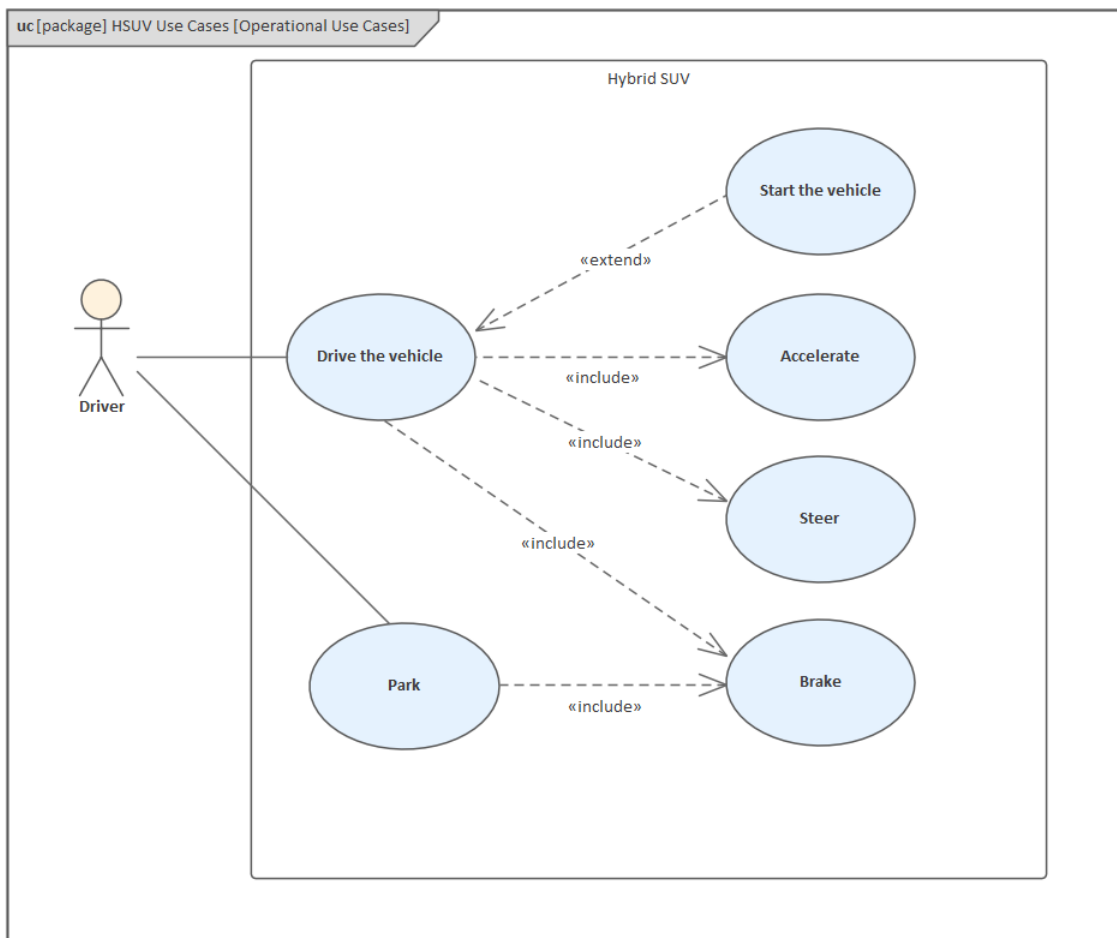
Type:		Scenario:		
Basic Path		Basic Path		
Step	Action	Uses	Results	State
1	The driver clicks the remote control for keyless entry.			
2	The system validates the signal and unlocks the car doors.			
3	The driver opens the driver's door and sits in the driving seat.			
4	<i>new step...</i>			

Une fois les étapes générées en tant qu'éléments de modèle, la traçabilité peut être ajoutée :

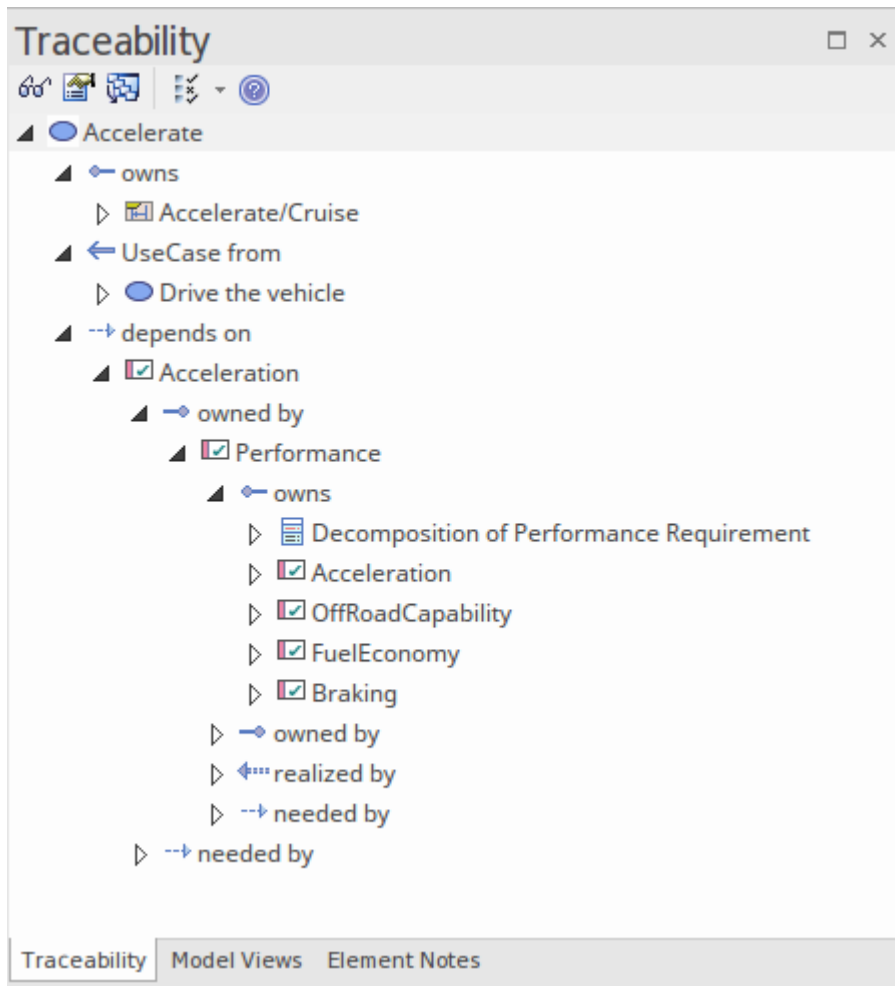
- Les étapes de l'acteur peuvent être retracées jusqu'aux modèles d'interface homme-machine
- Les étapes du système peuvent être retracées jusqu'aux modèles de composants.

## Diagramme de cas d'utilisation - Cas d'utilisation opérationnels

Les cas d'utilisation au niveau des objectifs associés à « Faire fonctionner le véhicule » sont représentés dans le diagramme suivant. Ces cas d'utilisation aident à préciser les types d'objectifs spécifiques associés à la conduite et au stationnement du véhicule. Le diagramme se concentre sur le conducteur du véhicule en tant qu'acteur central. Les cas d'utilisation de niveau supérieur tels que l'entretien, l'immatriculation et l'assurance du véhicule sont définis dans un ensemble distinct de cas d'utilisation contextuels.

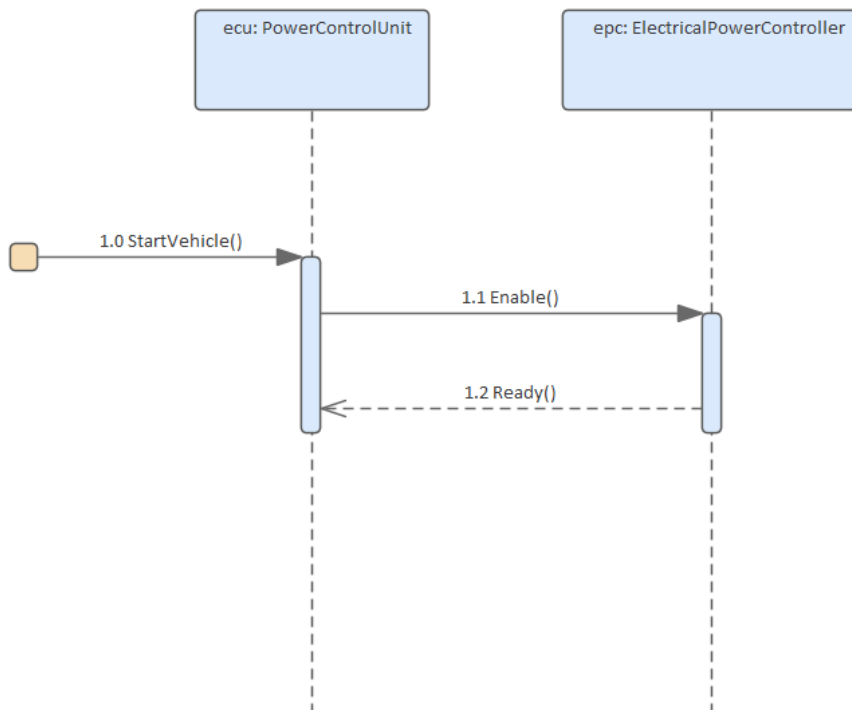


En plus des nombreuses fonctionnalités décrites dans la section précédente pour articuler les scénarios et leurs étapes détaillées, il existe un certain nombre d'autres fonctionnalités fournies par l'outil qui aident l'ingénieur système à travailler avec des cas d'utilisation. L'une des fonctionnalités les plus utiles est la fenêtre de traçabilité, qui fournit une visualisation convaincante de ce à quoi un cas d'utilisation donné est connecté et à son tour de ce à quoi l'élément connecté est lié. Lorsque différents cas d'utilisation (ou tout autre élément) sont sélectionnés dans la fenêtre Navigateur ou dans un diagramme, la fenêtre s'actualise pour afficher les connexions de l'élément sélectionné.



# Élaboration du comportement ( Diagrammes Séquence et StateMachine )

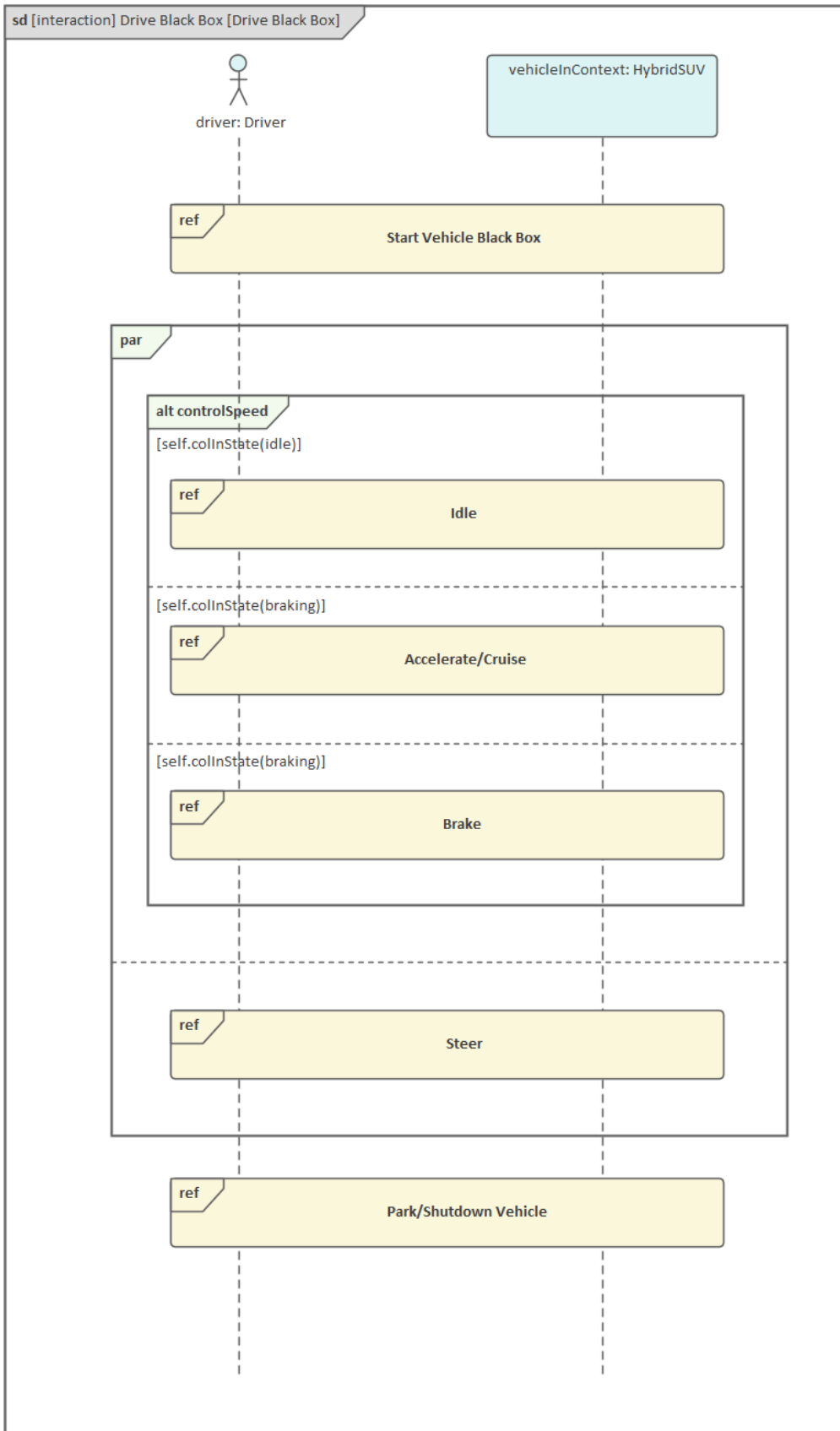
Ces rubriques et sections montrent comment le comportement peut être représenté et élaboré à l'aide diagrammes Séquence et StateMachine .



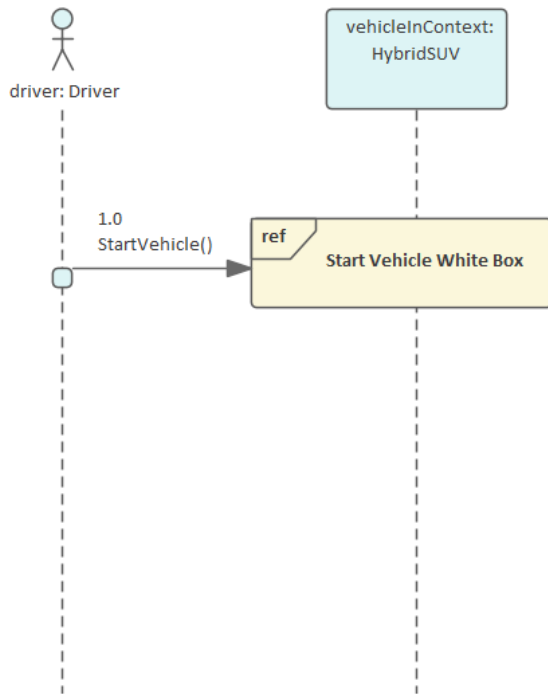


## Diagramme de Séquence - Drive Black Box

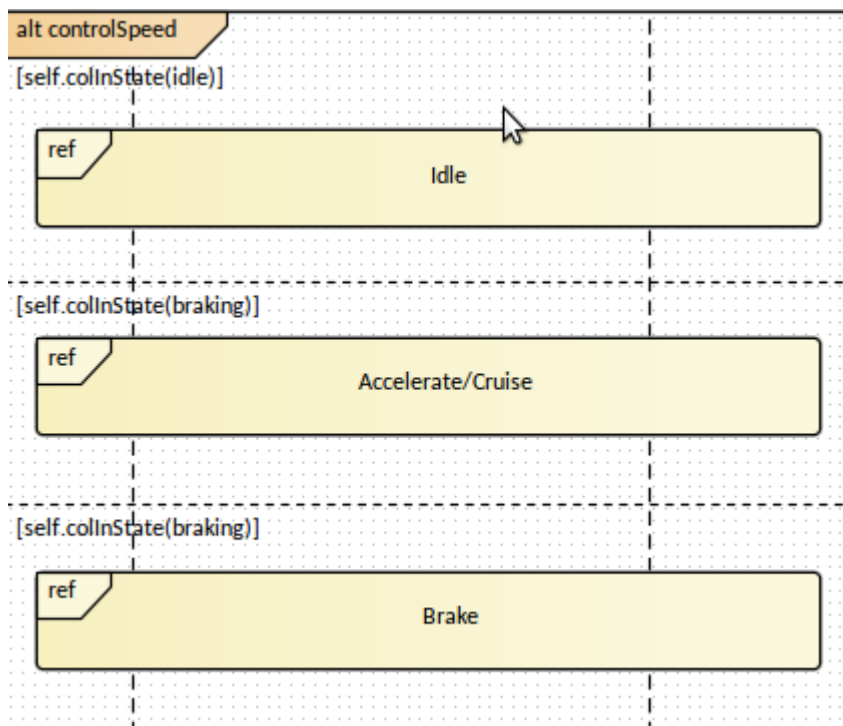
Le diagramme Séquence est un type de diagramme d'interaction et décrit la manière dont le conducteur interagit avec le véhicule dans un contexte particulier. Les éléments qui participent sont, par convention, listés de gauche à droite sur l'accès horizontal du diagramme, et le temps se déroule verticalement dans le diagramme. Les lignes pointillées qui émanent des objets sont appelées lignes de vie et représentent le temps d'existence des éléments.



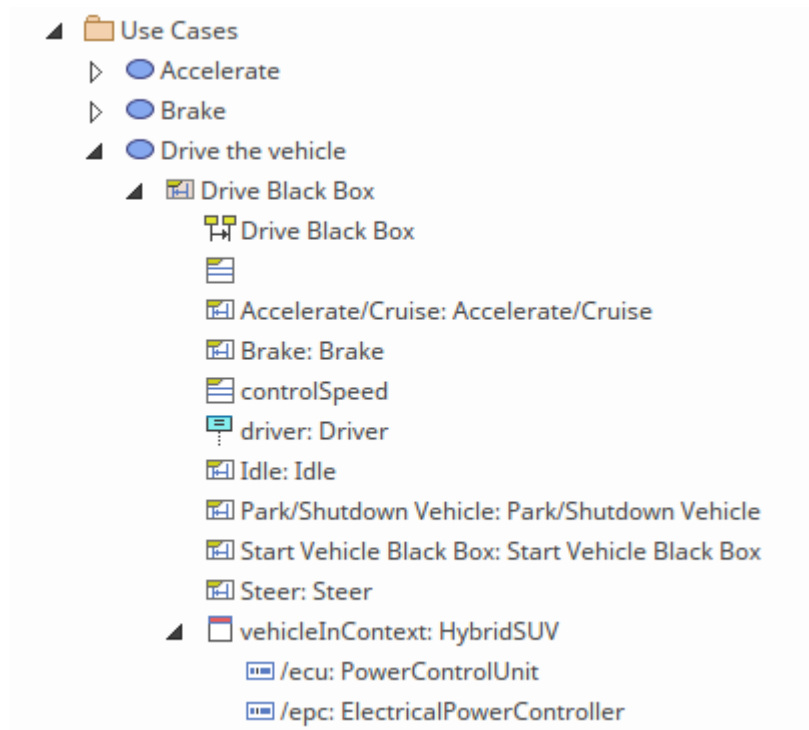
Ce diagramme utilise un type d'élément appelé Référence, qui agit comme un espace réservé pour un autre Diagramme de Séquence . Enterprise Architect permet d'ouvrir ces diagrammes en double-cliquant sur l'élément « ref » dans le diagramme .



Le diagramme utilise un fragment combiné parallèle (par) pour spécifier que les interactions de contrôle de la vitesse et de la direction se produisent en même temps (en parallèle). Le diagramme utilise également un fragment combiné avec une désignation alternative (alt) qui spécifie un certain nombre de façons (alternatives) par lesquelles le conducteur peut contrôler la vitesse.

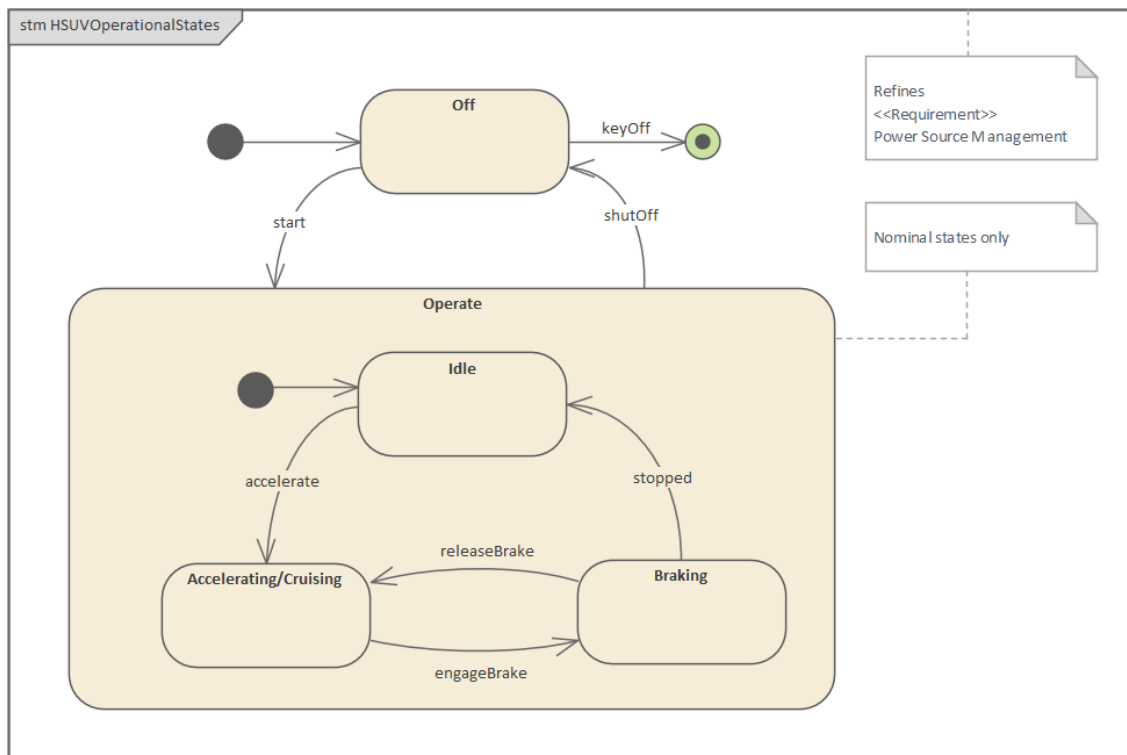


Enterprise Architect permet de définir le Diagramme de Séquence comme un enfant du cas d'utilisation « Conduire le véhicule », ce qui permet au modélisateur d'accéder facilement au diagramme et de le visualiser dans le contexte de l'objectif que le conducteur souhaite atteindre par rapport au véhicule.

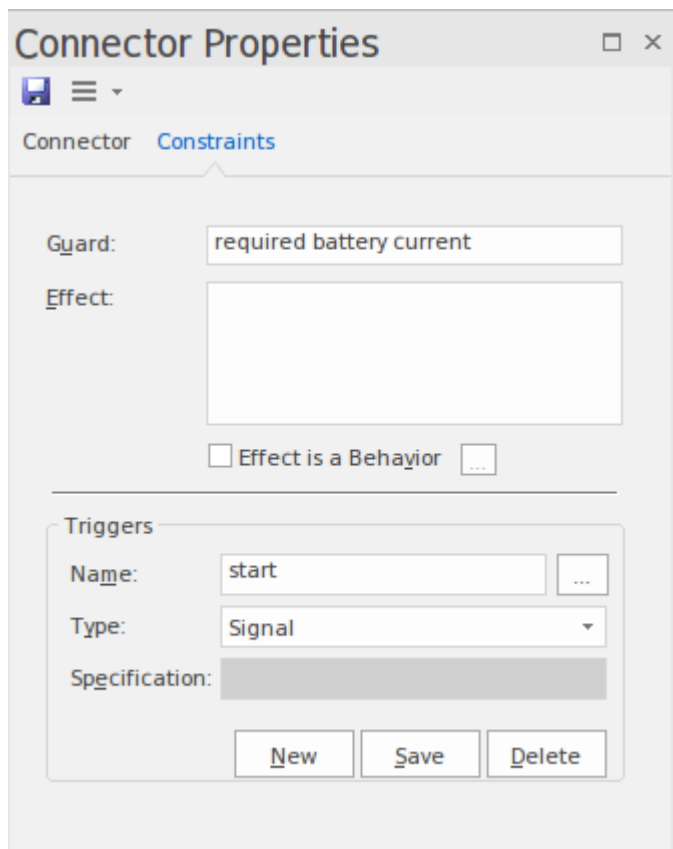


## Diagramme Statemachine - States opérationnels HSUV

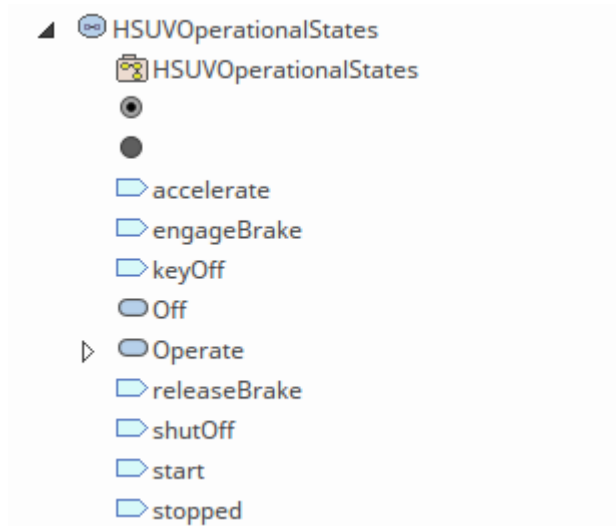
Ce diagramme Statemachine est utilisé pour décrire les états discrets que le SUV hybride présente pendant une partie spécifiée de sa durée de vie. Un système ou une partie d'un système peut avoir une large gamme d'états en fonction de la perspective de l'observateur, de sorte qu'un modélisateur doit toujours spécifier la perspective ou la vue utilisée pour le diagramme. Le diagramme articule ensuite les conditions importantes et pertinentes au cours de la durée de vie spécifiée de l'entité.



Le message du diagramme Séquence Démarrer Véhicule est le déclencheur qui va provoquer la transition du véhicule de l'état Arrêt à l'état Marche (On). Enterprise Architect permet de définir ces transitions en détail.

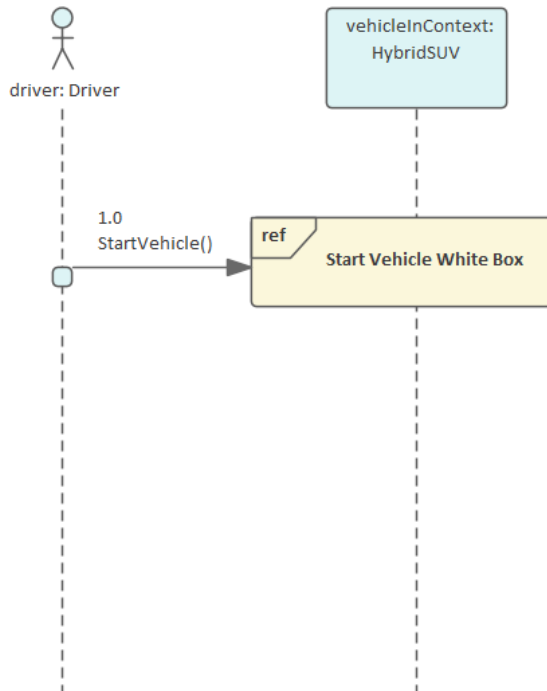


Les States apparaissent dans la fenêtre Navigateur et sont regroupés de manière pratique sous le nœud Statemachine .

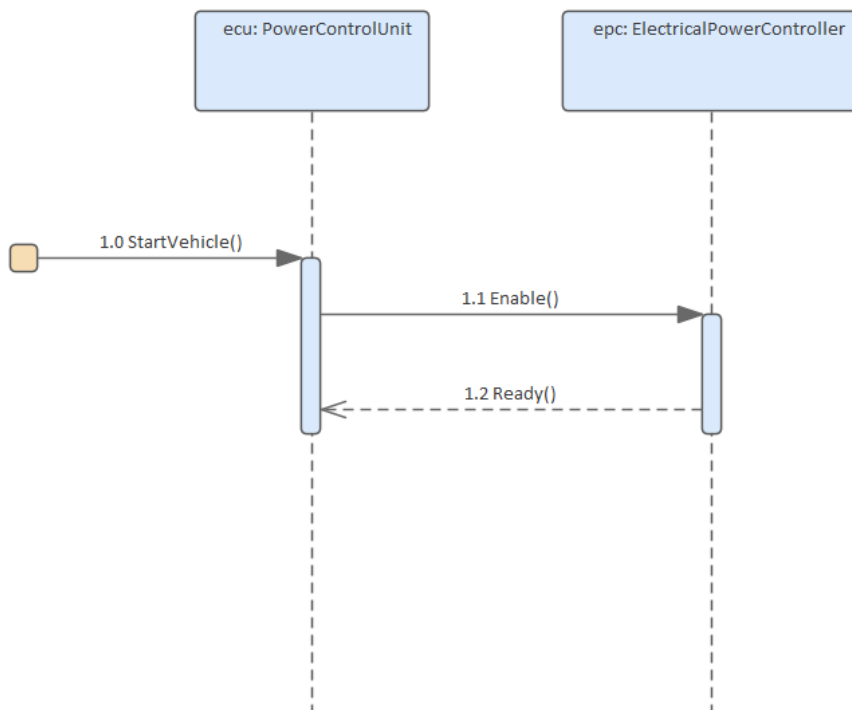


## Diagramme de Séquence - Démarrer le véhicule Black Box et White Box

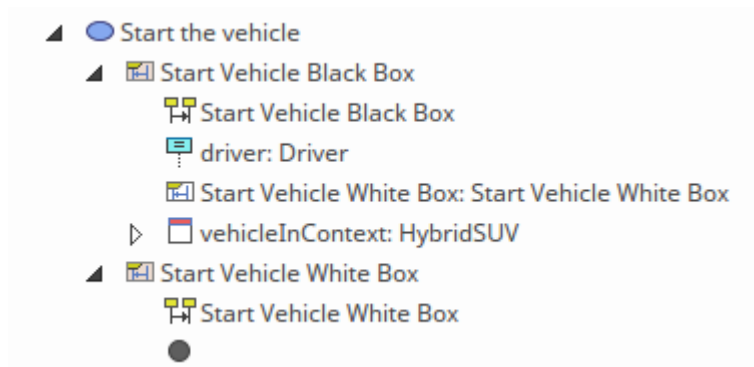
Ce diagramme Séquence décrit l'interaction entre le conducteur et le démarrage du véhicule. Il utilise l'élément Interaction Use indiqué par (ref) pour faire référence à un autre diagramme Séquence . Le diagramme est intrinsèquement simple mais ces diagramme peuvent être complétés par un certain nombre d'autres éléments et connecteurs qui élaborent les détails d'autres interactions plus complexes.



Enterprise Architect fournit un mécanisme pratique permettant au modélisateur de cliquer sur le diagramme référencé, dans ce cas le diagramme Démarrer Vehicle White Box .



Les éléments d' Enterprise Architect peuvent apparaître dans plusieurs diagrammes ce qui permet de construire des récits expressifs dans un modèle et de fournir aux modélisateurs un mécanisme pour créer plusieurs vues du même élément. Les éléments d'un diagramme peuvent être situés dans la fenêtre Navigateur montrant leur relation structurelle avec d'autres parties du modèle. Dans ce cas, les vues Boîte noire et White Box de l'interaction Démarrer Vehicle sont situées en tant qu'enfants du cas d'utilisation Démarrer Vehicle, ce qui permet de les relier facilement les unes aux autres.





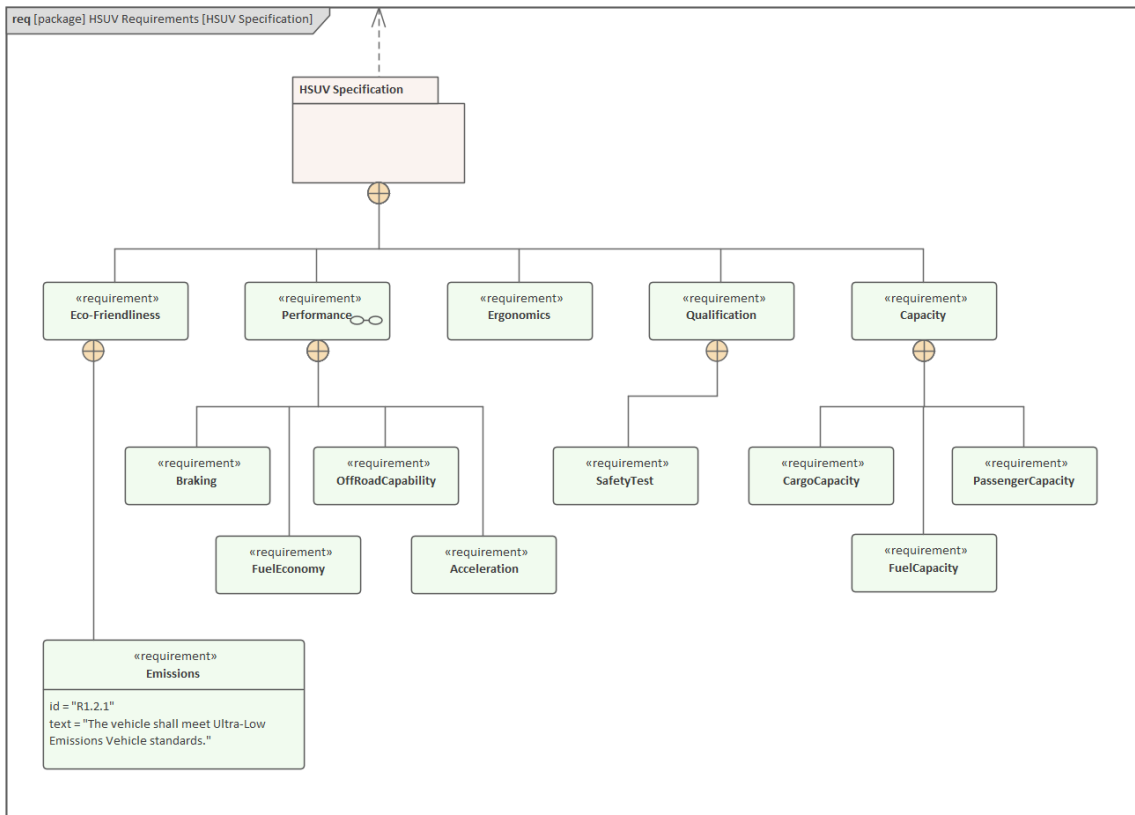
## Etablir Exigences ( Diagrammes et Tableaux Exigences )

L'objectif de cette partie du modèle est de démontrer la modélisation visuelle des exigences. La spécification du système du véhicule contient de nombreuses exigences textuelles qui ont été recréées dans Enterprise Architect . Une série d'exigences ont été modélisées, notamment l'exigence selon laquelle le véhicule doit satisfaire aux normes d'émissions, qui est développée à des fins d'illustration.

Enterprise Architect propose une large gamme d'outils pour créer, développer, analyser, gérer et tester les exigences. Il intègre également l'outil de gestion des exigences DOORS .

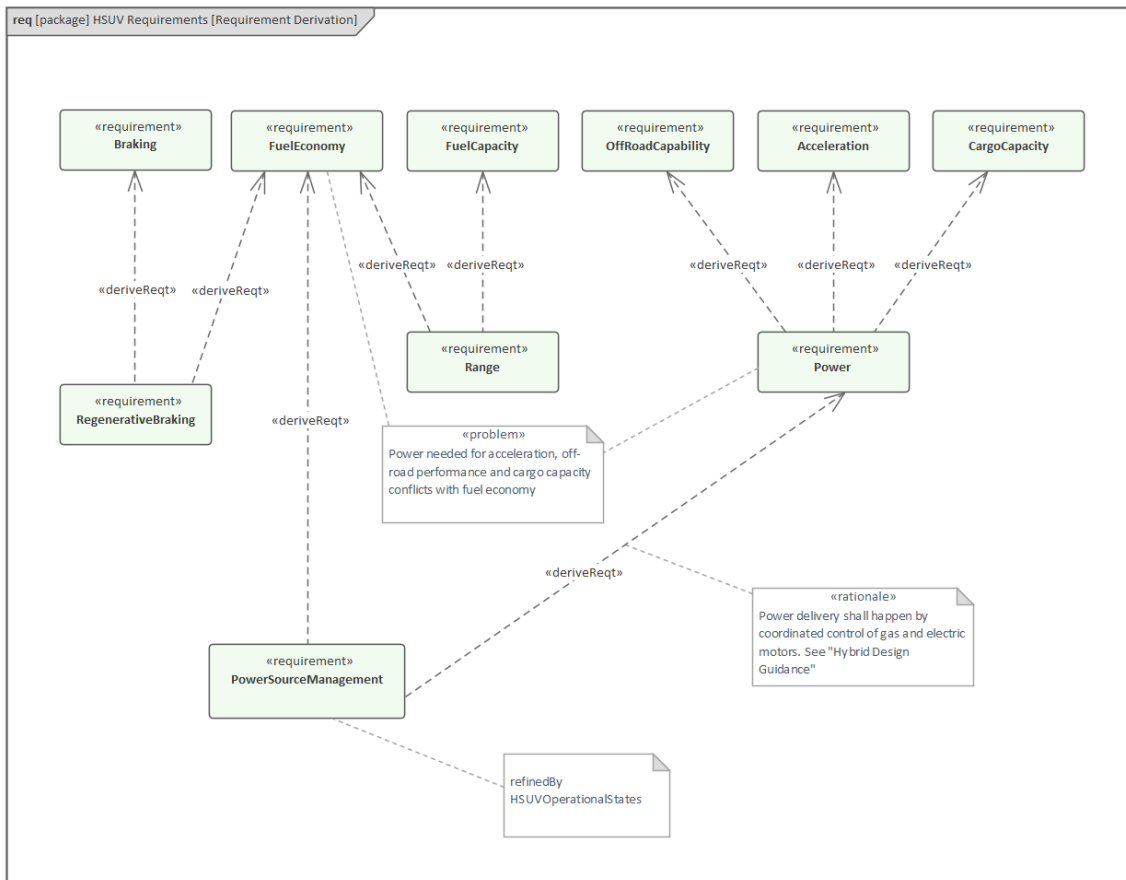
# Diagramme des exigences - Hiérarchie des exigences HSUV

Ce diagramme illustre la visualisation des exigences dans une hiérarchie à l'aide du connecteur de confinement (hachures croisées) pour montrer la relation parent-enfant. Les exigences de niveau supérieur agissent comme un type de système de regroupement ou de confinement et couvrent une gamme de préoccupations de haut niveau qui sont décomposées en énoncés de niveau inférieur et vraisemblablement mesurables.



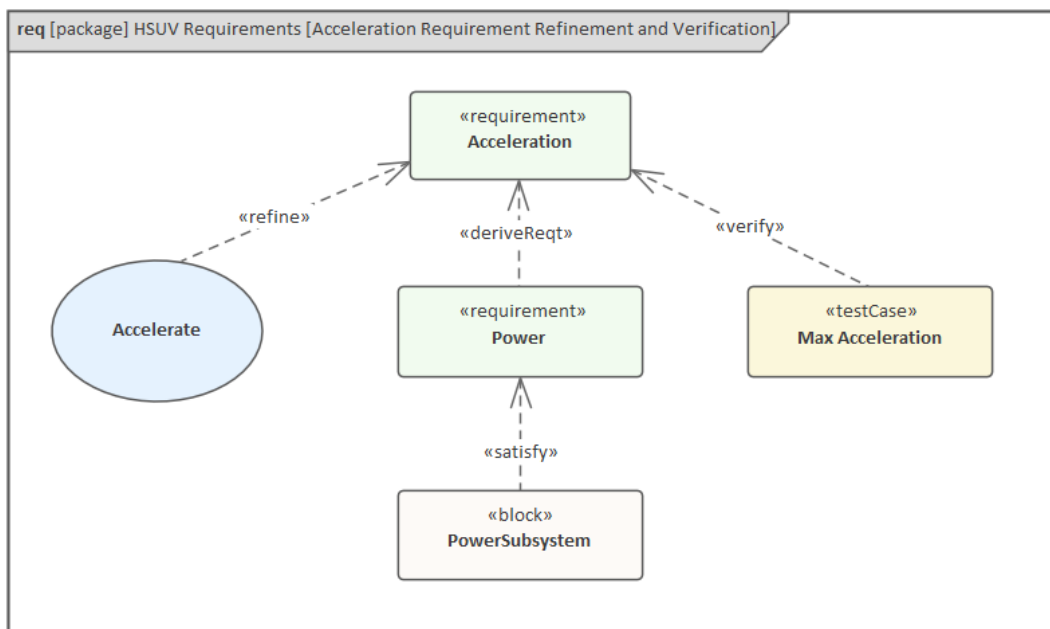
# Diagramme d'exigences - Exigences dérivées

Ce diagrammes d'exigences montre qu'un certain nombre d'exigences ont été dérivées d'autres exigences. Cette relation implique qu'une analyse a été menée et que l'exigence dérivée représente un besoin qui peut répondre à l'intention de l'exigence initiale. Par exemple, le freinage régénératif n'était pas une exigence, mais il a été dérivé du besoin d'avoir à la fois le freinage et l'économie de carburant comme exprimé dans les exigences à l'extrémité de la flèche de la relation.



## Diagramme d'exigences - Relations d'exigences d'accélération

Ce diagramme illustre un certain nombre de relations entre exigences qui apparaissent généralement sur un diagramme d'exigences après une analyse et modélisation. L'élément central est l'exigence d'accélération et le diagramme montre comment un certain nombre d'autres éléments sont liés à cette exigence. La relation d'affinage est introduite comme un moyen de relier un cas d'utilisation portant le même nom à l'exigence d'accélération. Nous avons une autre exigence dérivée qui est ensuite satisfaite par un Bloc. Un cas Test « Accélération maximale » est également représenté sur le diagramme et lié à l'exigence centrale par une relation Vérifie. Le diagramme montre également que des éléments autres que les exigences peuvent être ajoutés et contribuer à rendre le diagramme plus expressif.



# Tableau - Exigences Tableau

Ces exemples montrent comment Exigences peuvent être affichées sous forme de tableau comme alternative à la représentation graphique dans diagrammes . Il s'agit d'un style de présentation apprécié par un certain nombre d'intervenants qui sont plus habitués à travailler avec des feuilles de calcul. Le premier tableau répertorie les exigences avec leurs identifiants et leurs déclarations textuelles. Le deuxième tableau répertorie les exigences sources et cibles qui participent à la relation de dérivation.

Enterprise Architect fournit également une gamme d'outils et de moyens de visualisation des exigences (et d'autres éléments), notamment Vues de liste, des tableaux Kanban, Vues Spécification , des diagrammes de Gantt, des graphiques et bien plus encore.

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, but have dramatically better fuel economy.
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.
2	Performance	The Hybrid SUV shall have the braking, acceleration, and off-road capability of a typical SUV, but have dramatically better fuel economy.

Showing 1 - 5 of 10 items

req [requirement] Performance [Tree of Performance Requirements]

Tree of Performance Requirements				
DERIVED_ID	DERIVED_NAME	RELATION	ID	NAME
2.2	FuelEconomy	deriveReq	d.3	Range
2.2	FuelEconomy	deriveReq	d.2	PowerSourceManagement
2.2	FuelEconomy	deriveReq	d.1	RegenerativeBraking
2.1	Braking	deriveReq	d.1	RegenerativeBraking
4.2	FuelCapacity	deriveReq	d.3	Range
4.1	CargoCapacity	deriveReq	d.4	Power
d.4	Power	deriveReq	d.2	PowerSourceManagement

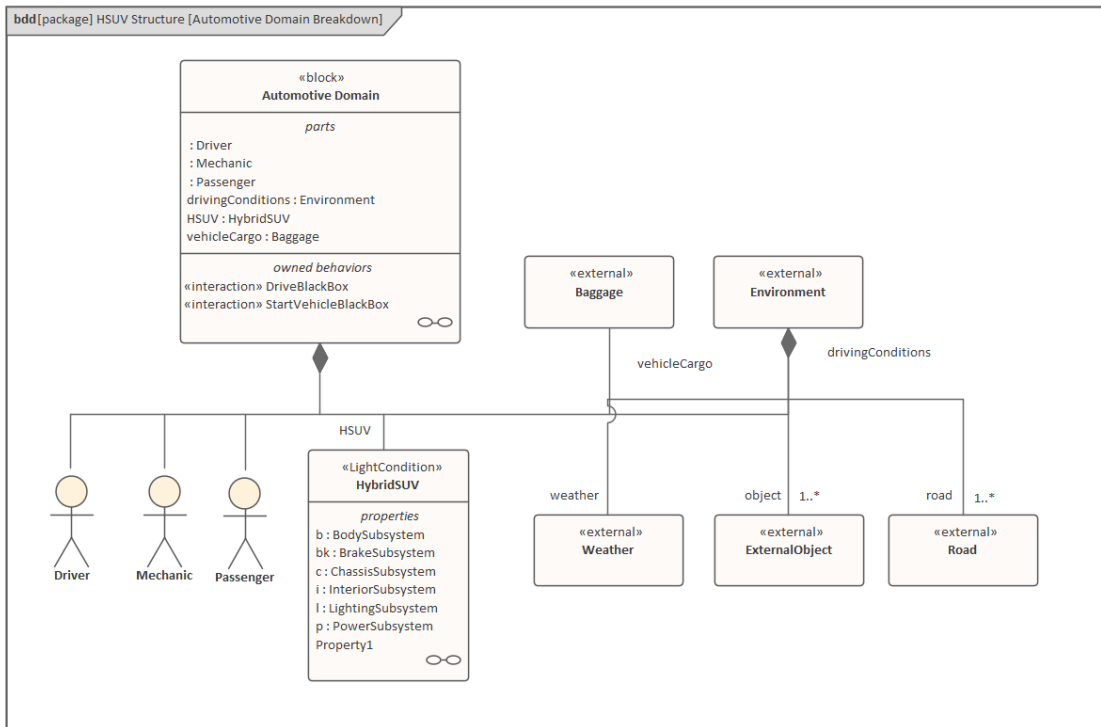
Showing 1 - 7 of 23 items

## Décomposer les pièces ( Diagrammes de définition Bloc , Diagrammes Bloc internes)

Les blocs sont les unités fondamentales de la structure et peuvent être utilisés à la fois sur les diagrammes de définition Bloc et Diagrammes Bloc internes pour décrire les aspects structurels d'un système. Le diagramme de définition Bloc est souvent un point de départ pour de nombreux ingénieurs souhaitant acquérir des connaissances sur un système et voir comment il est structuré. Le Bloc lui-même est composé d'une structure et celle-ci est représentée par Fonctionnalités , cela inclut des pièces qui sont elles-mêmes typées par d'autres blocs, par exemple un ensemble de roue pourrait avoir une pièce d'étrier à disque. Il existe également des propriétés de valeur, qui sont des éléments qui ont une quantité et qui représentent des dimensions physiques et autres dimensions mesurables, par exemple une voiture pourrait avoir un poids et une couleur et avoir un temps d'accélération de 0 à 100 km de 5 secondes. Il existe également des points d'interaction qui montrent les points avec lesquels un bloc peut interagir avec son environnement.

# Interrompre lorsqu'une Variable Change de Valeur - Domaine Automobile

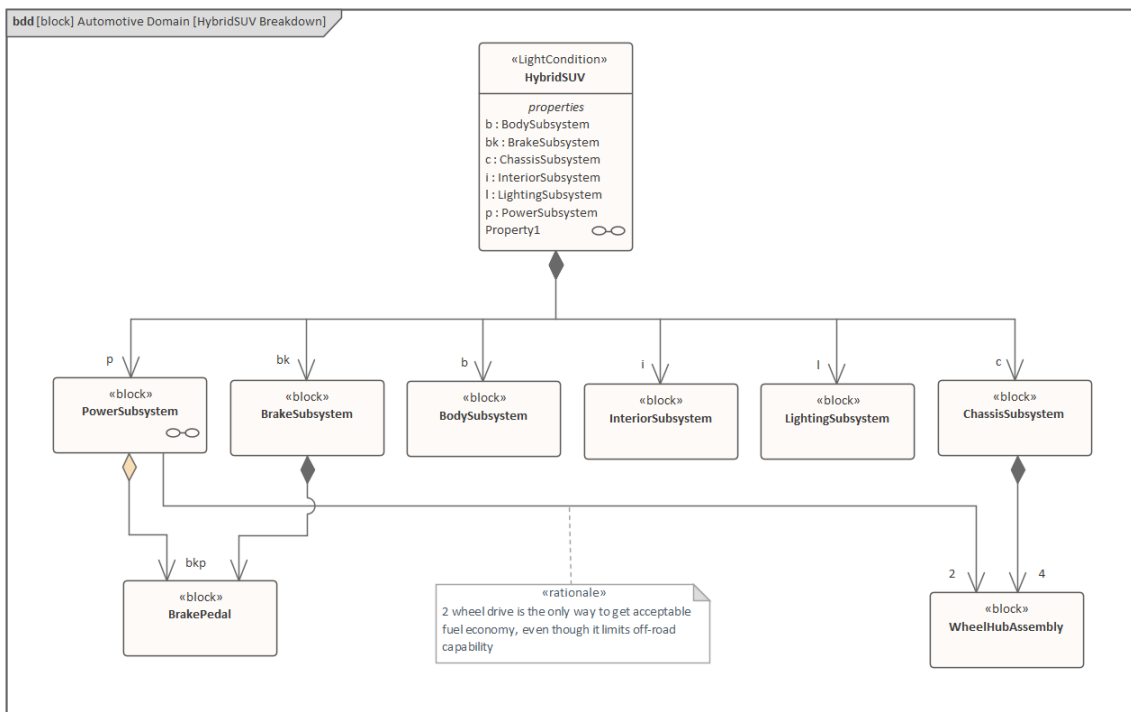
Ce diagramme illustre l'utilisation d'un diagramme de définition Bloc pour décrire les parties qui composent le domaine automobile. Le domaine comprend des personnes qui, tout en exécutant un rôle, interagissent avec le système, notamment un conducteur, un passager et un mécanicien. Il peut donc y avoir un certain nombre de personnes spécifiques exécutant le rôle de conducteur et le mécanicien, lors de l'exécution d'un test après la réparation du sous-système de freinage, changerait également son rôle de mécanicien à conducteur.



# Interrompre lorsqu'une Variable Change de Valeur - Hybrid SUV

Tous les systèmes qui nécessitent une analyse et une conception techniques seront d'un niveau de complexité qui nécessitera de les décomposer en un certain nombre de parties pour aider à réduire la complexité et faciliter la gestion du projet. Le premier niveau de décomposition est généralement appelé sous-système et dans le cas du SUV hybride ou de toute autre automobile, les sous-systèmes comprendraient des blocs tels que l'alimentation, le freinage, l'éclairage et le châssis.

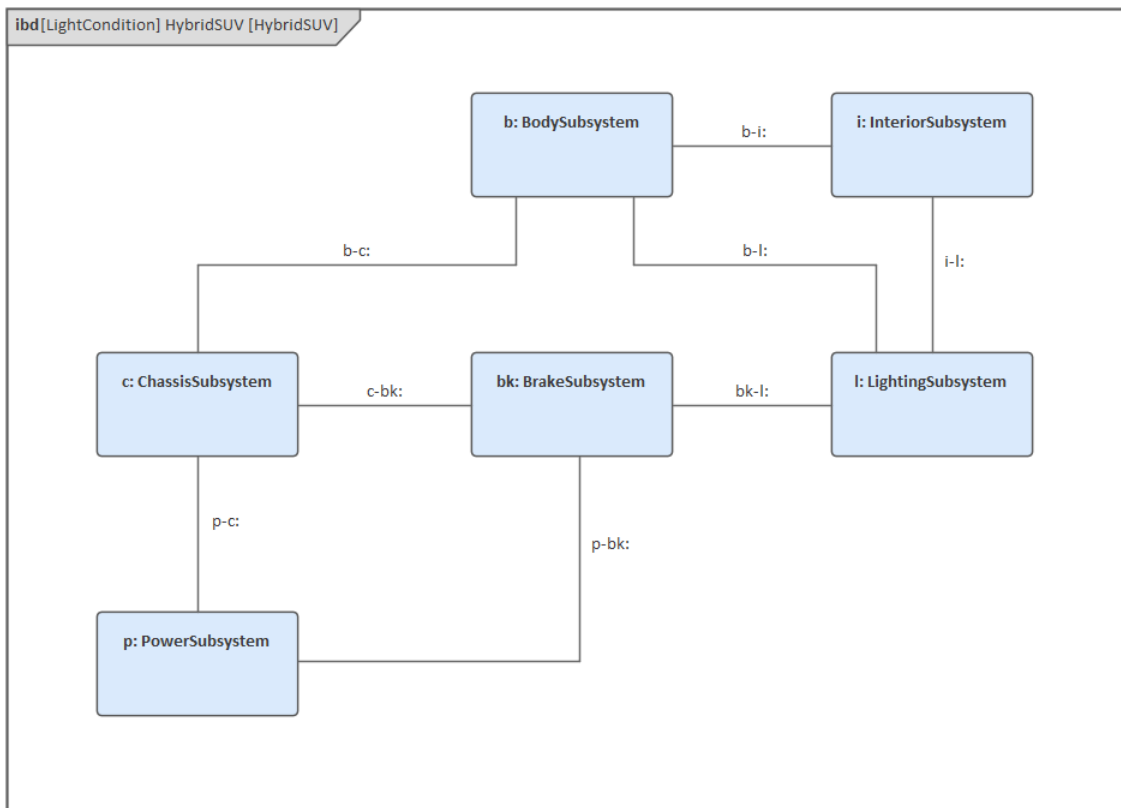
Ces sous-systèmes seraient eux-mêmes décomposés en un certain nombre d'éléments constitutifs, par exemple le sous-système de freinage pourrait être décomposé en ensembles de disques et en éléments hydrauliques.





## Diagramme Interne de Bloc - SUV Hybride

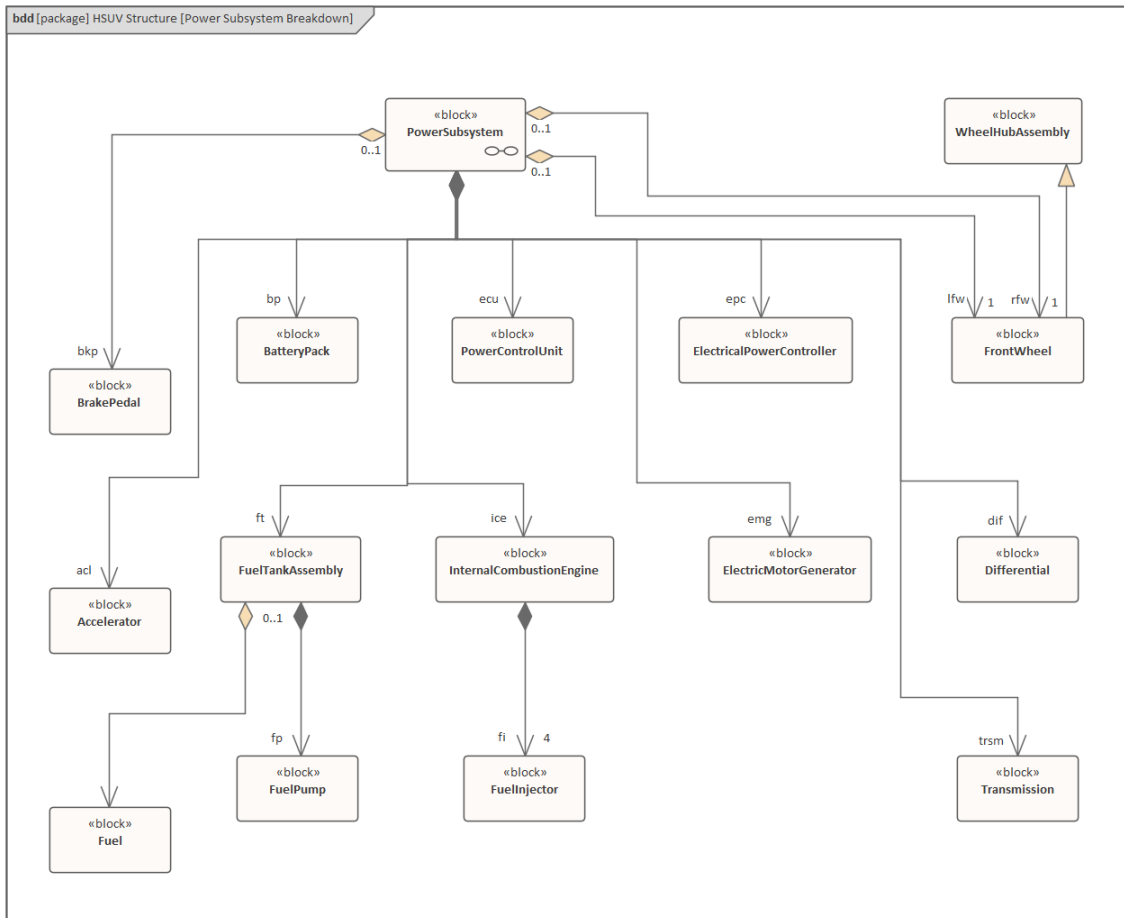
Le diagramme Bloc interne fournit un mécanisme permettant de décrire la manière dont les pièces sont liées ou connectées les unes aux autres dans le contexte du bloc entier ou propriétaire. Dans notre exemple de SUV hybride, nous pouvons voir une connexion entre le sous-système d'alimentation et le sous-système de freinage, probablement pour modéliser le freinage assisté. Ainsi, alors que le diagramme de définition Bloc montre la structure en termes de composition, le Bloc interne est capable de regarder à l'intérieur du bloc et de voir comment il est « câblé » ensemble.



Nous verrons dans une section ultérieure comment une forme spécialisée du diagramme Bloc internes, à savoir le diagramme Paramétriques, est utilisée pour modéliser des systèmes d'équations mathématiques.

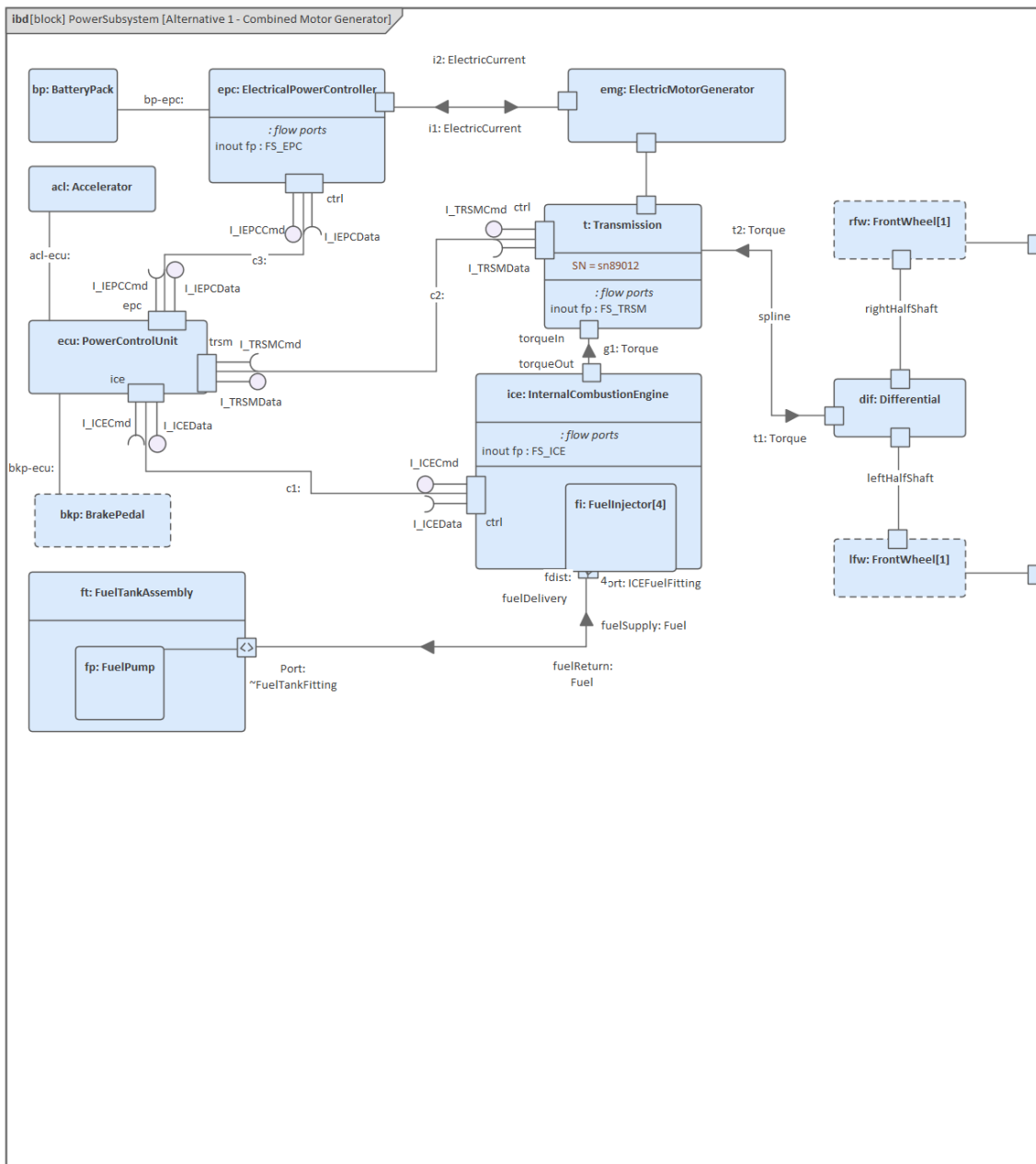
# Interrompre lorsqu'une Variable Change de Valeur - Power Subsystem

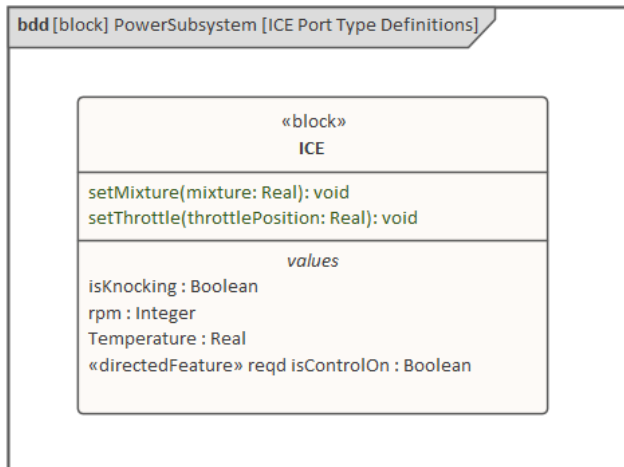
Dans les deux sections précédentes, nous avons vu comment un système peut être décomposé en un certain nombre de sous-systèmes et que cette décomposition peut être représentée sur un diagramme de définition Bloc . Nous avons également vu comment les sous-systèmes peuvent être reliés les uns aux autres dans le contexte du système global et comment ceux-ci peuvent être représentés sur un diagramme définition Bloc internes. Nous allons maintenant voir comment l'un de ces sous-systèmes (le sous-système de puissance) peut être décomposé en un certain nombre de parties constitutives en utilisant à nouveau le diagramme de définition Bloc .



# Diagramme Interne de Bloc pour le Sous-Système de Puissance

De la même manière que nous avons montré les connexions entre les sous-systèmes, en utilisant un diagramme Bloc internes, nous pouvons faire la même chose pour représenter la manière dont les parties d'un sous-système sont connectées entre elles. Nous voyons donc à nouveau comment les deux types diagramme , diagramme de définition Bloc et un diagramme Bloc internes, peuvent être utilisés en tandem pour décrire la structure d'un système et comment nous pouvons descendre dans une hiérarchie de parties jusqu'à un point où la complexité est comprise et ne nécessite pas modélisation supplémentaire.





## Définition des ports et des flux

Les diagrammes des rubriques de cette section montrent comment les éléments qui circulent peuvent être modélisés, à l'aide diagrammes Ports, Flux et Spécification de flux sur Définition Bloc , Bloc interne et Paramétriques . La plupart des systèmes physiques auront des éléments qui circulent, ce qui peut être une partie importante du fonctionnement du système. Nous pourrions considérer un certain nombre d'exemples, notamment :

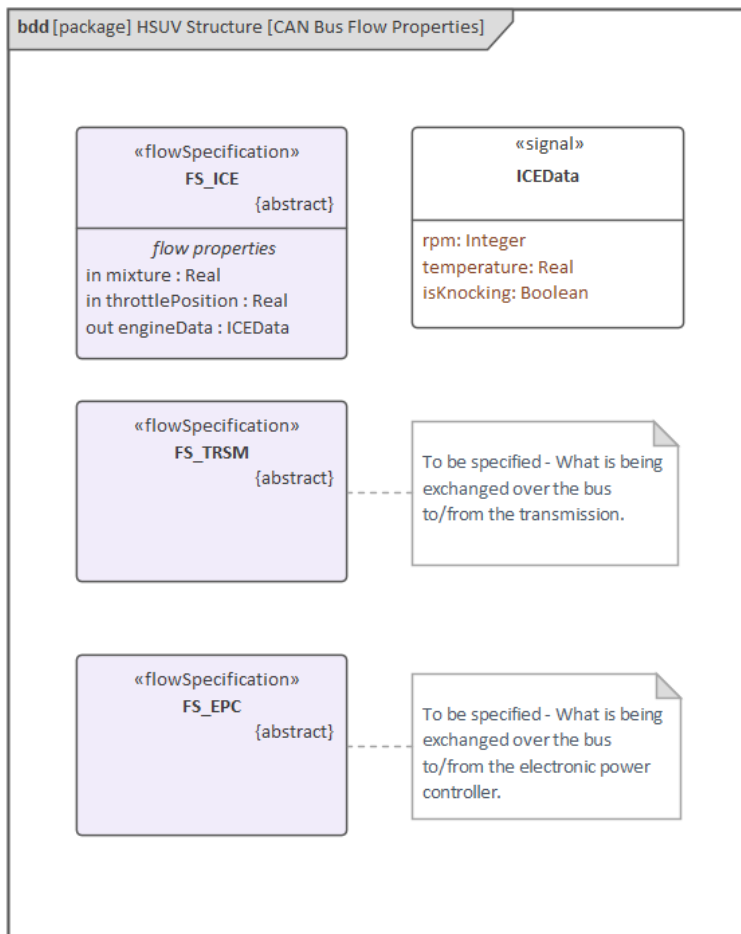
- Une usine de dessalement - où l'eau salée et l'eau douce ainsi que l'électricité circulent à travers le système
- Une ligne de production où les assemblages, les pièces, l'alimentation et les instructions de contrôle du robot circulent dans le système
- Un système de transport urbain dans lequel les trains, les tramways, les bus, les ferries et les passagers circulent
- Un aéronef dans lequel le carburant, l'air, les signaux de commande, le liquide hydraulique, les passagers et les agents de bord circulent dans le système

Les diagrammes commencent par définir une architecture de bus CAN (Controller Area Network) et montrent comment un certain nombre de spécifications de flux peuvent être utilisées pour définir la manière dont les éléments circulent entre les parties du sous-système d'alimentation. Le flux de carburant est modélisé à l'aide d'un diagramme de définition Bloc qui présente des ports de flux (obsolètes dans la version 1.5 de SysML) qui montrent le « conduit » logique permettant au carburant de s'écouler depuis l'assemblage de combustible et le moteur à combustion interne. diagrammes Bloc internes vont plus loin, et enfin un diagramme Paramétriques est utilisé pour montrer comment une équation mathématique pour le débit de carburant, définie dans une contrainte, peut être utilisée pour modéliser l'équation. Les tracés simulés sont ensuite visualisés à l'aide des capacités de simulation d' Enterprise Architect .

## Interrompre lorsqu'une Variable Change de Valeur - ICE Flow Propriétés

Ce diagramme montre les premières étapes (non terminées) de la définition et du perfectionnement d'une architecture de bus. Le modélisateur a utilisé des spécifications de flux pour modéliser la manière dont les éléments circuleront dans le véhicule ; par exemple, une spécification de flux a été définie pour :

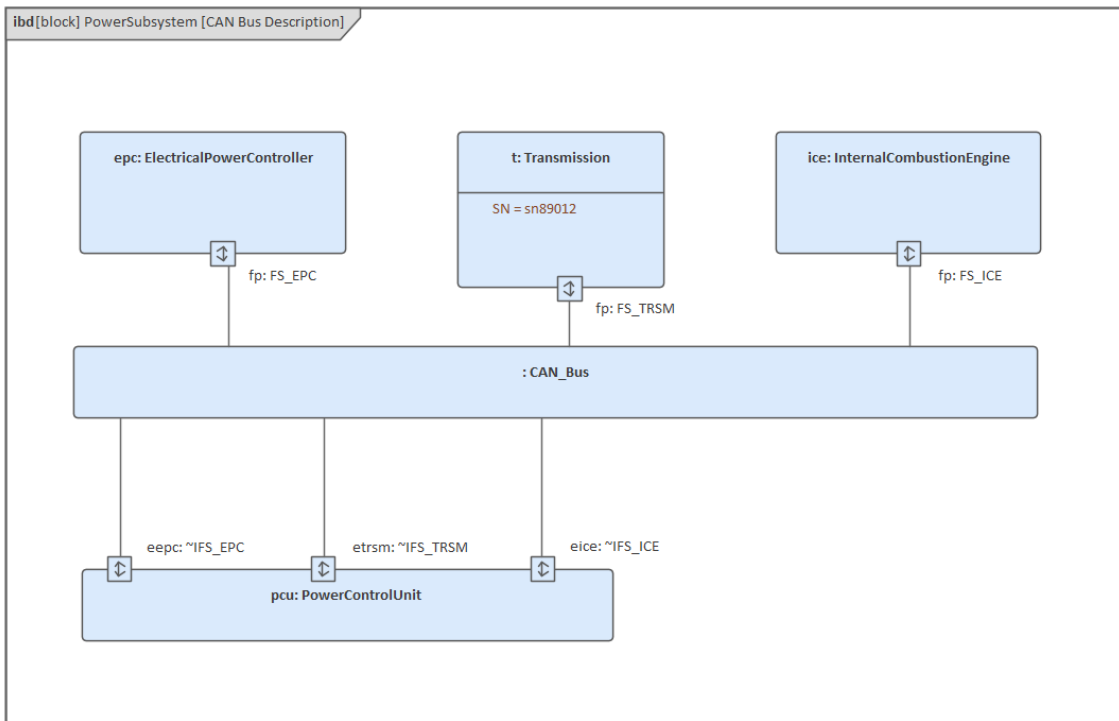
- Le moteur à combustion interne FS\_ICE
- Le système de transmission
- Le contrôleur de puissance électronique



Le diagramme sera affiné dans les itérations ultérieures du processus, et les ports et les flux seront utilisés pour modéliser les éléments qui circulent à travers et entre les différents sous-systèmes.

## Diagramme Interne de Bloc - CAN Bus

Ce diagramme poursuit le perfectionnement de l' architecture du bus, en utilisant un diagramme Bloc interne pour montrer comment les différents systèmes sont intégrés dans le bus CAN (Controller Area Network). Cette architecture de bus CAN est un dispositif central pour contrôler et intégrer différentes parties des sous-systèmes du SUV hybride.

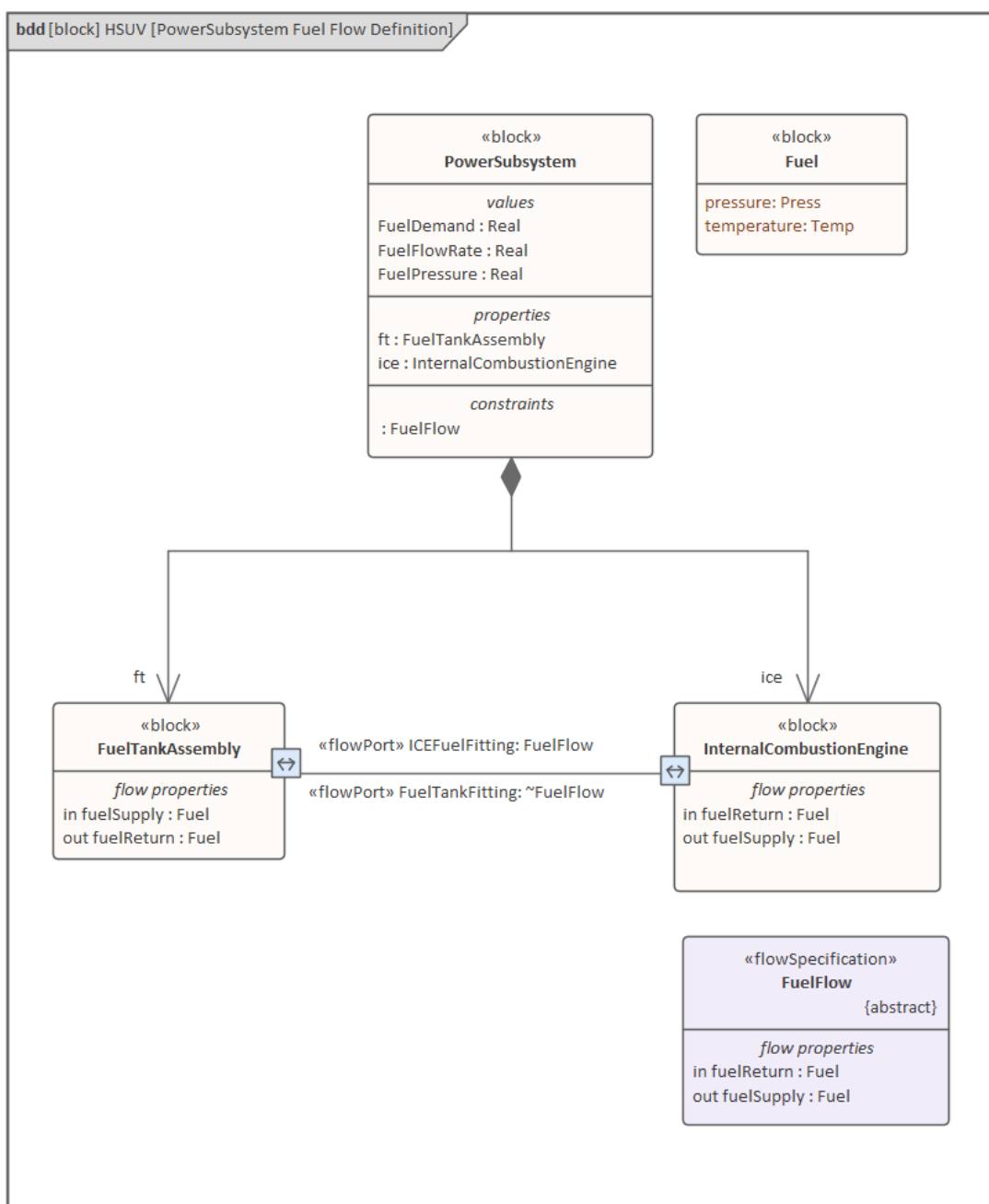


# Interrompre lorsqu'une Variable Change de Valeur - Fuel Flow Propriétés

Ce diagramme de définition Bloc continue modélisation de la façon dont le carburant s'écoule de l'ensemble réservoir de carburant vers le moteur à combustion interne, en élaborant la définition du débit de carburant. Le carburant lui-même est modélisé comme un Bloc et possède deux propriétés valeur qui définissent les caractéristiques physiques importantes, à savoir :

- Température du carburant
- Pression de carburant

Le Bloc de sous-système d'alimentation est divisé en deux parties importantes, à savoir l'ensemble réservoir de carburant et le moteur à combustion interne. Les deux parties ont des ports d'écoulement définis et un connecteur a été dessiné entre les deux ports indiquant que l'élément « carburant » peut s'écouler du réservoir de carburant vers le moteur.





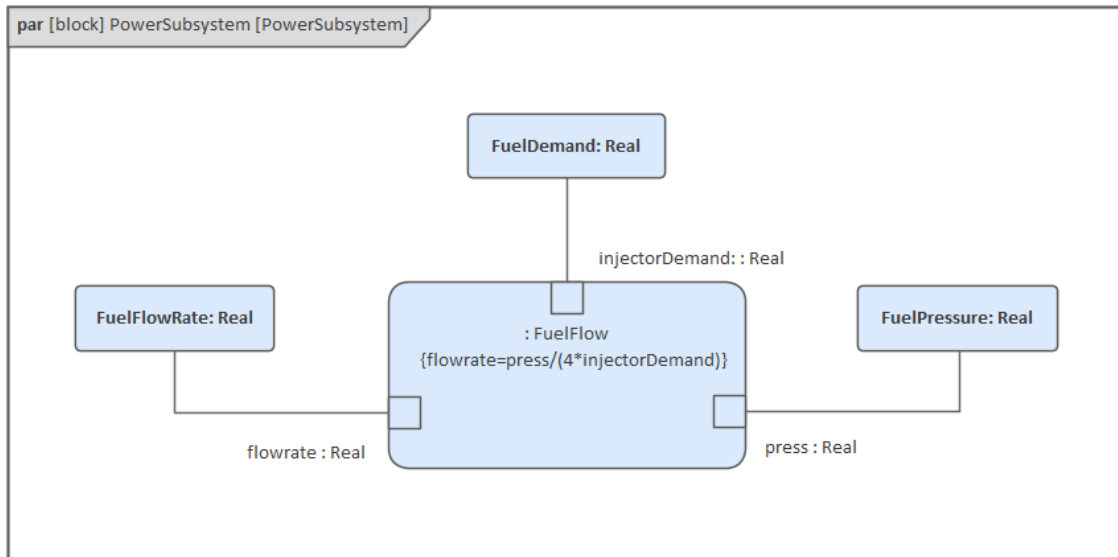


## Diagramme Paramétriques - Débit De Carburant

Ce diagramme Paramétriques montre comment les équations mathématiques peuvent être modélisées à l'aide de ConstraintProperties et de paramètres liés au périmètre de ConstraintProperty. Dans ce diagramme nous voyons que le débit est lié à la fois à la demande de carburant et à la pression de carburant, à l'aide de l'équation :

{débit=pression/(4\*demandeinjecteur)}

La contrainte est modélisée dans le Bloc de contraintes et peut être utilisée dans un certain nombre de contextes différents, en utilisant ConstraintProperties sur diagrammes Paramétriques . Enterprise Architect dispose d'une facilité de simulation avancé qui utilise OpenModelica ou Simulink pour créer des tracés d'équations modélisées.



## Analyser les performances ( Diagrammes de contraintes, Diagrammes , Vues )

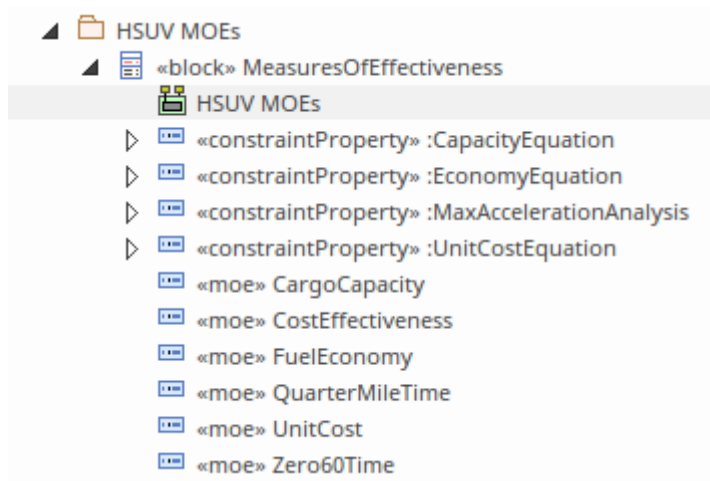
Les diagrammes de cette section de l'exemple sont en grande partie diagrammes Paquetage qui décrivent les points de vue et Vues utilisés pour répondre aux préoccupations des parties prenantes. Les rubriques présentent également les mesures d'efficacité (MOE) qui peuvent être utilisées dans les études commerciales pour évaluer les solutions et les architectures candidates. Les concepts de point de vue et Vue sont articulés dans la norme ISO-42010 (anciennement IEEE-1471) et les constructions SysML Viewpoint et Vue sont cohérentes avec cette norme. Les exemples typiques de Vues incluent les aspects opérationnels, de fabrication ou de sécurité, et ceux-ci sont ensuite liés aux éléments du modèle.

Le modèle Viewpoint et Vue est mieux considéré comme un modèle narratif ou descriptif, qui permet de clarifier et d'expliquer un modèle de système. Un modèle Viewpoint et Vue expose les éléments d'un ou plusieurs modèles de système. Plus précisément, un Viewpoint est un cadre particulier à partir duquel visualiser les modèles de système et est une spécification de règles pour construire une Vue afin de répondre à un ensemble de préoccupations qui sont importantes pour les parties prenantes. Par exemple, un architecte de performance aura des préoccupations différentes de celles d'un architecte de sécurité. La Vue est destinée à visualiser le système à partir du Viewpoint spécifié. Cela fournit un mécanisme permettant aux parties prenantes de spécifier les aspects du modèle de système qui sont importants pour elles à partir de leur Viewpoint, puis de représenter ces aspects du système dans une Vue spécifique.

Le point de vue décrit le point de vue ou la lentille à travers laquelle un groupe de parties prenantes examine un modèle de système et, en encadrant les préoccupations des parties prenantes ainsi que la méthode de production d'une Vue , leurs préoccupations peuvent être prises en compte. La méthode décrit :

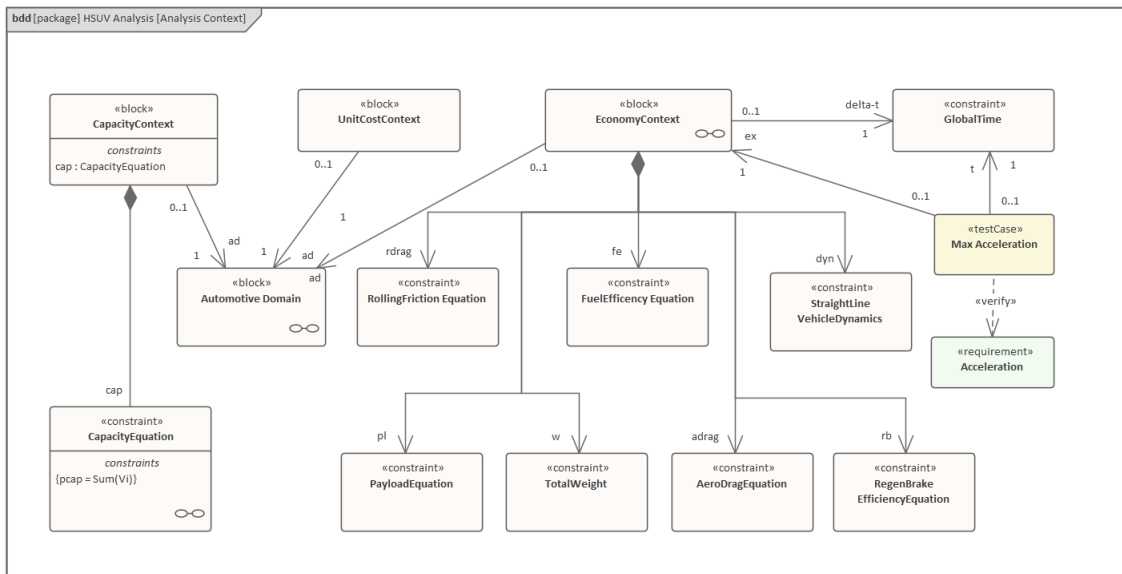
- L'attente de ce que les parties prenantes souhaitent voir exposé à partir du modèle
- Comment la partie prenante souhaite que l'information soit structurée et présentée
- Dans quel type d'artefact la partie prenante souhaite-t-elle consommer l'information ?

En d'autres termes, le processus est l'ensemble des règles qui décrivent comment Vue doit exprimer les informations du modèle pour répondre aux préoccupations des parties prenantes. Lorsque les Vues et les points de vue sont modélisés dans Enterprise Architect la relation avec les éléments modélisation peut être définie.



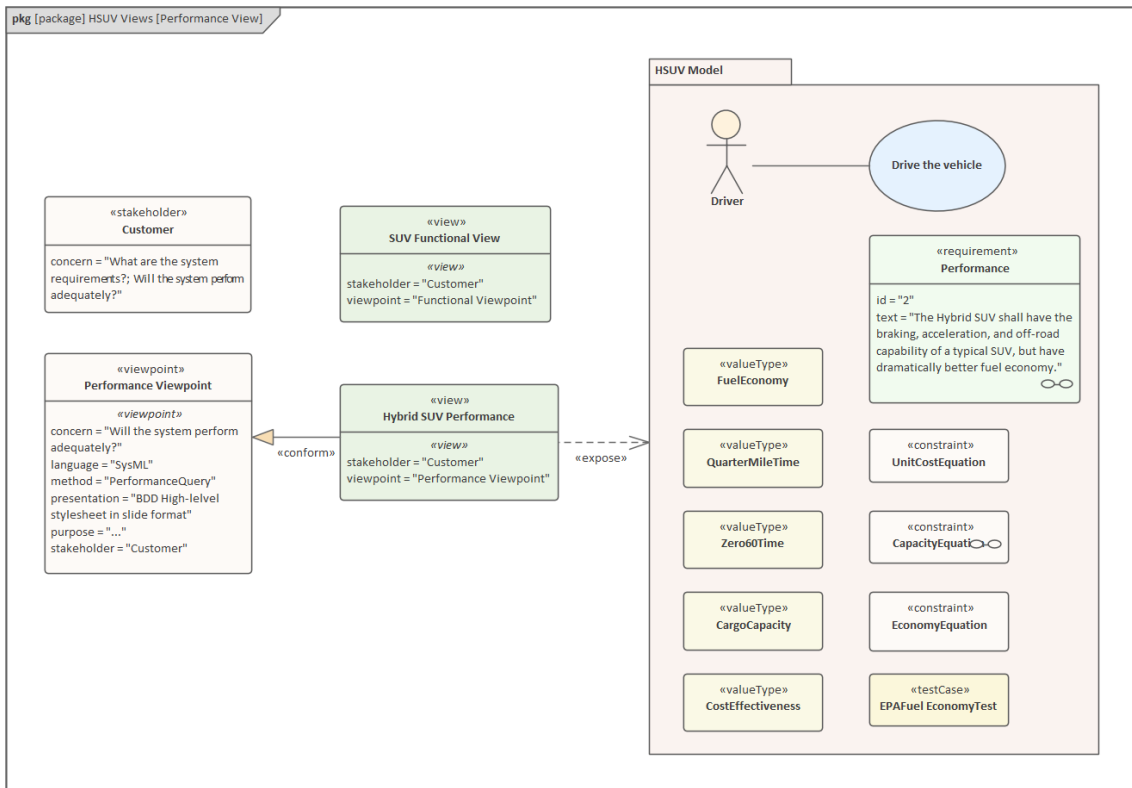
# Interrompre lorsqu'une Variable Change de Valeur - Contexte d'analyse

L'une des principales raisons de la modélisation d'un système est de pouvoir effectuer des analyses sur les modèles, ce qui permet de disposer d'un moyen rentable d'obtenir des informations sur la manière dont le système construit fonctionnera in situ. Effectuer une analyse sur un modèle est moins coûteux et plus pratique que de construire des prototypes. Ce diagramme de définition Bloc définit les différents éléments du modèle qui seront utilisés pour effectuer l'analyse dans cet exemple. Il décrit chacun des blocs de contrainte et les équations associées qui seront utilisés pour l'analyse, ainsi que les relations clés entre eux. Deux types d'éléments sont présents sur le diagramme : les blocs et les contraintes. Le diagramme montre également la relation de vérification entre une exigence et un cas Test .



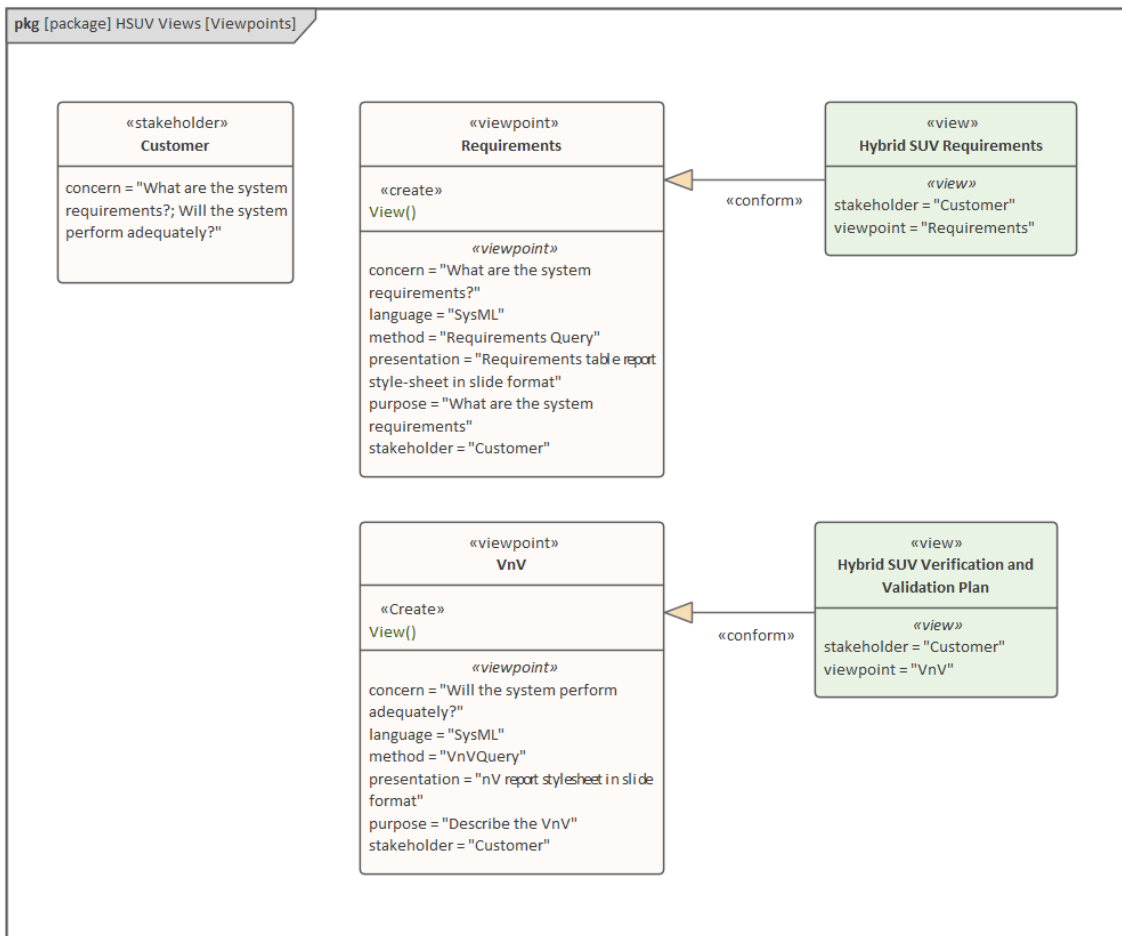
# Diagramme Paquetage - Définition Vue performance

Le SysML permet à une équipe de définir ses propres points de vue. Dans cet exemple, nous voyons un point de vue de performance défini par l'utilisateur et les éléments qui remplissent la Vue performance spécifique à HSUV. La Vue de performance elle-même peut contenir un certain nombre de diagrammes décrivant les éléments qu'elle contient. Nous pouvons voir dans le diagramme qu'un certain nombre de vues ont été définies, notamment la vue de performance du SUV hybride et Vue fonctionnelle du SUV. Chaque vue a une partie prenante définie et un point de vue. La relation d'exposition a été utilisée pour relier la Vue de performance au modèle SUV et la relation de conformité montre que la Vue de performance est conforme au point de vue de performance.



# Diagramme Paquetage - Définition du point de vue

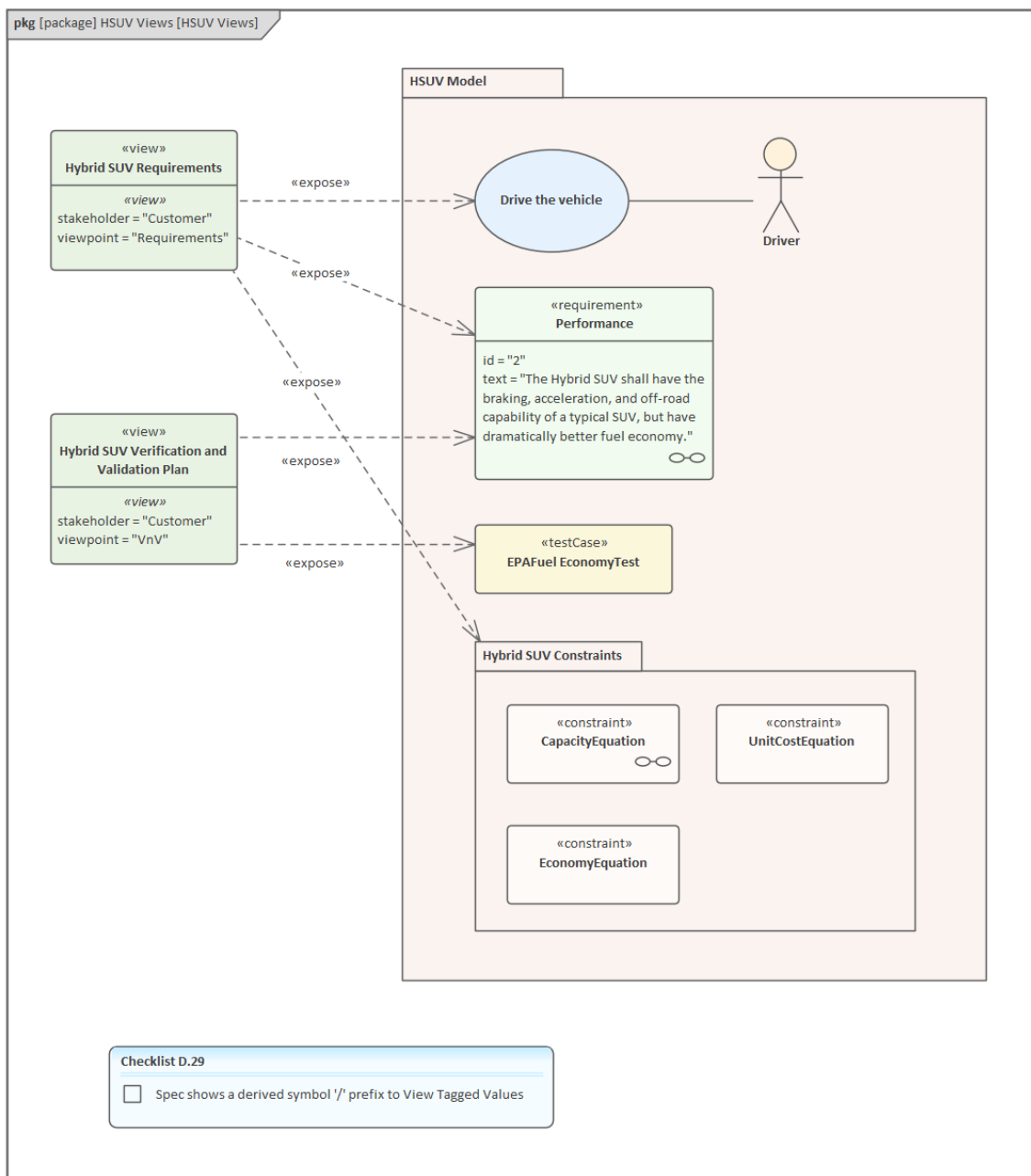
Ce diagramme Paquetage montre les définitions des points de vue Exigences et Verification et Validation (VnV) avec les relations avec les parties prenantes, les préoccupations et les vues. La partie prenante et le point de vue partagent la même préoccupation via des commentaires qui sont affichés textuellement comme des valeurs de la propriété de préoccupation. Les commentaires peuvent être affichés graphiquement avec des relations d'annotation avec les parties prenantes et les points de vue, si nécessaire. Note que la valeur de la propriété de partie prenante est une instance du stéréotype et non la classe à laquelle le stéréotype est appliqué.



# Diagramme Paquetage - Définition Vue

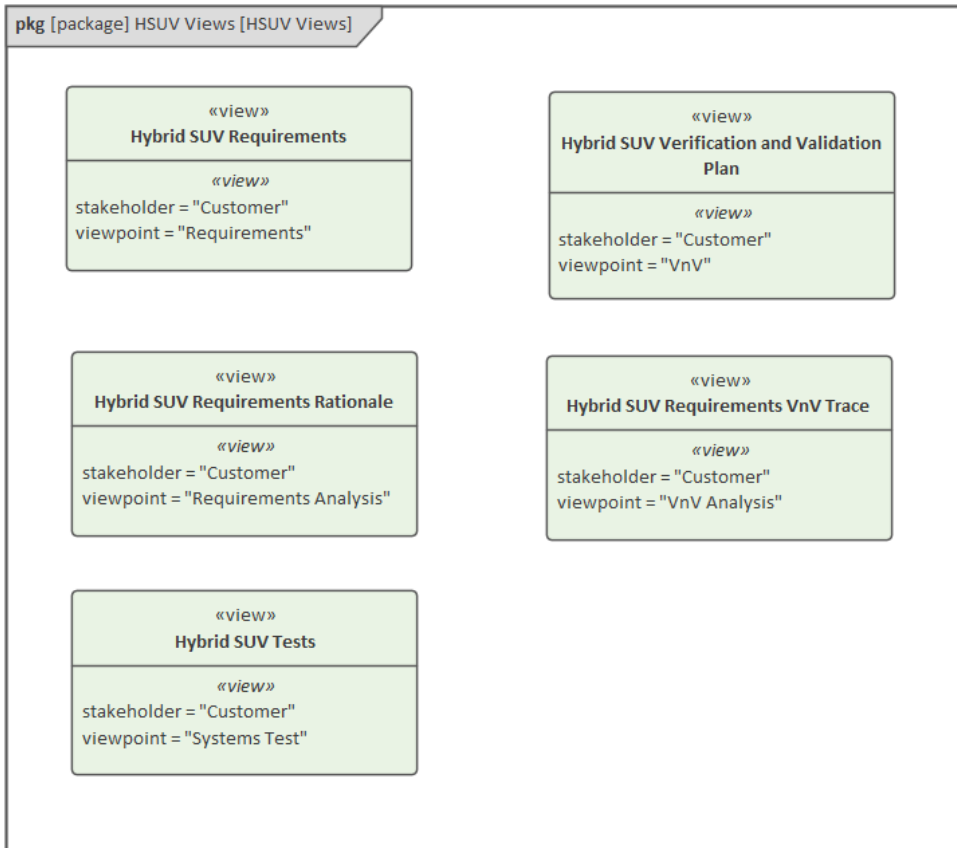
Ce diagramme montre l'utilisation de la relation Expose, qui est un dispositif graphique permettant d'indiquer les éléments (y compris Paquetages) qui font partie de la vue. Les Vues Exigences et Verification et Validation ont des relations Expose sortantes qui ciblent un certain nombre d'éléments du modèle. Cela fournit un moyen utile d'indiquer les éléments impliqués dans la vue ; par exemple, on peut voir à partir du diagramme que la vue Exigences SUV hybride expose le cas d'utilisation du véhicule de conduite, une exigence de performance et un Paquetage contenant un groupe de contraintes SUV.

En utilisant certains des outils de visualisation riches fonctionnalité, il serait également possible de visualiser à quelles Vues un élément donné a participé, par exemple le cas d'utilisation du véhicule de conduite pourrait apparaître dans un certain nombre de Vues différentes.



# Diagramme Paquetage - Hiérarchie Vue

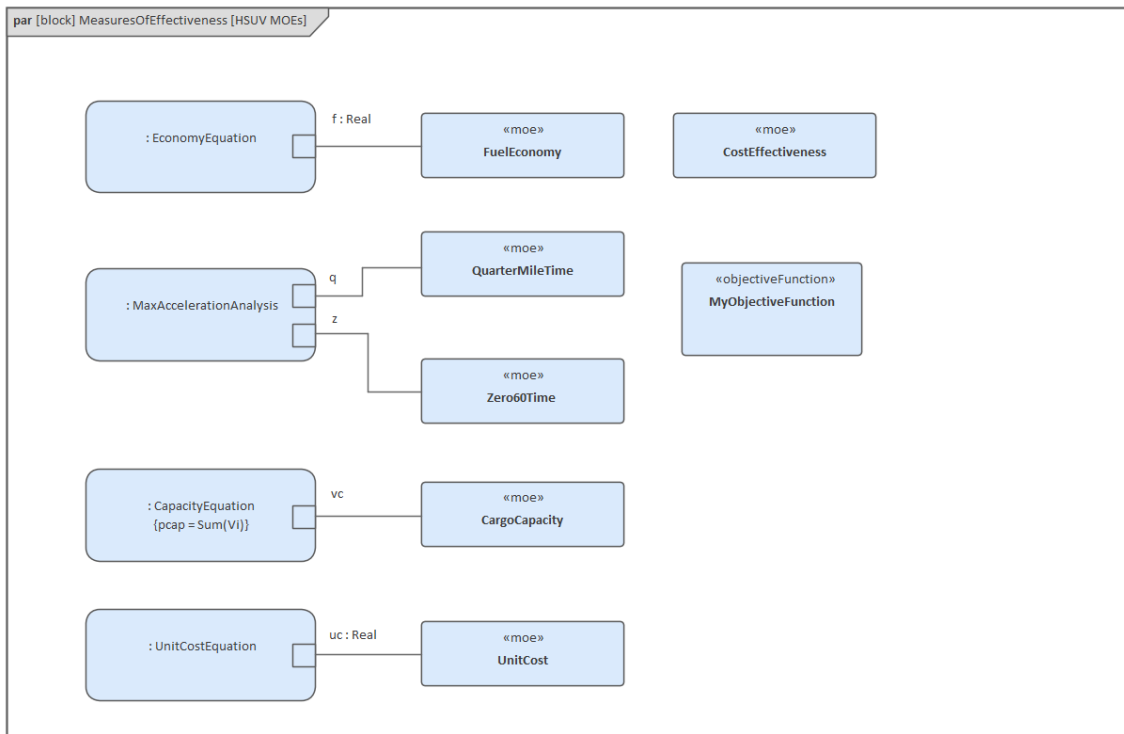
Ce diagramme Paquetage montre comment les vues, ou d'ailleurs tout autre élément avec le même stéréotype, peuvent être rassemblés dans un Paquetage et présentés visuellement.





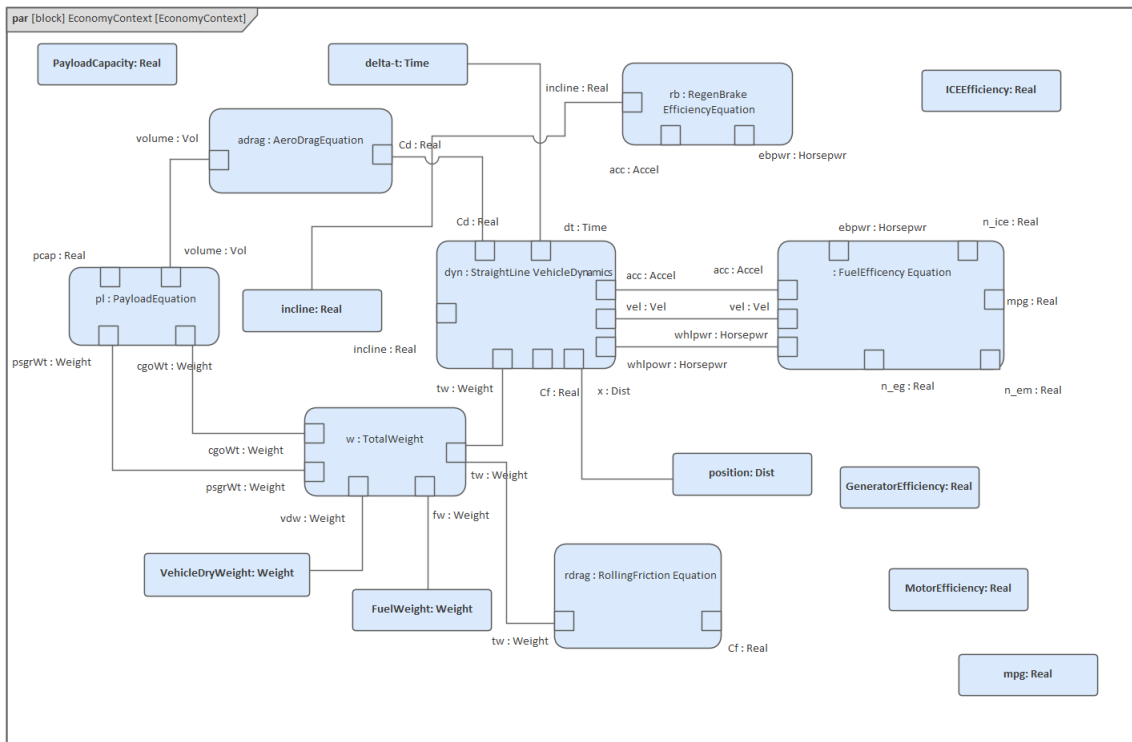
## Diagramme Paramétriques - Mesures d'Efficacité

La mesure de l'efficacité est un mécanisme permettant d'évaluer une solution en établissant un ensemble de mesures qui permettront à l'équipe d'ingénierie d'évaluer deux ou plusieurs solutions à un problème. Cette technique est généralement appelée étude commerciale et les mesures d'efficacité (MOE) sont calculées pour deux ou plusieurs solutions et comparées à l'aide d'une fonction d'utilité (objectif). La MOE est un stéréotype défini par l'utilisateur et ne fait pas formellement partie du langage de base SysML ; elle s'appuie sur le mécanisme d'extension de stéréotype qui permet d'étendre la grammaire du langage. Ce diagramme Paramétriques montre comment la rentabilité globale du HSUV sera évaluée. Il montre les MOE particuliers pour une alternative pour la conception du HSUV et peut être réutilisé pour évaluer d'autres alternatives.



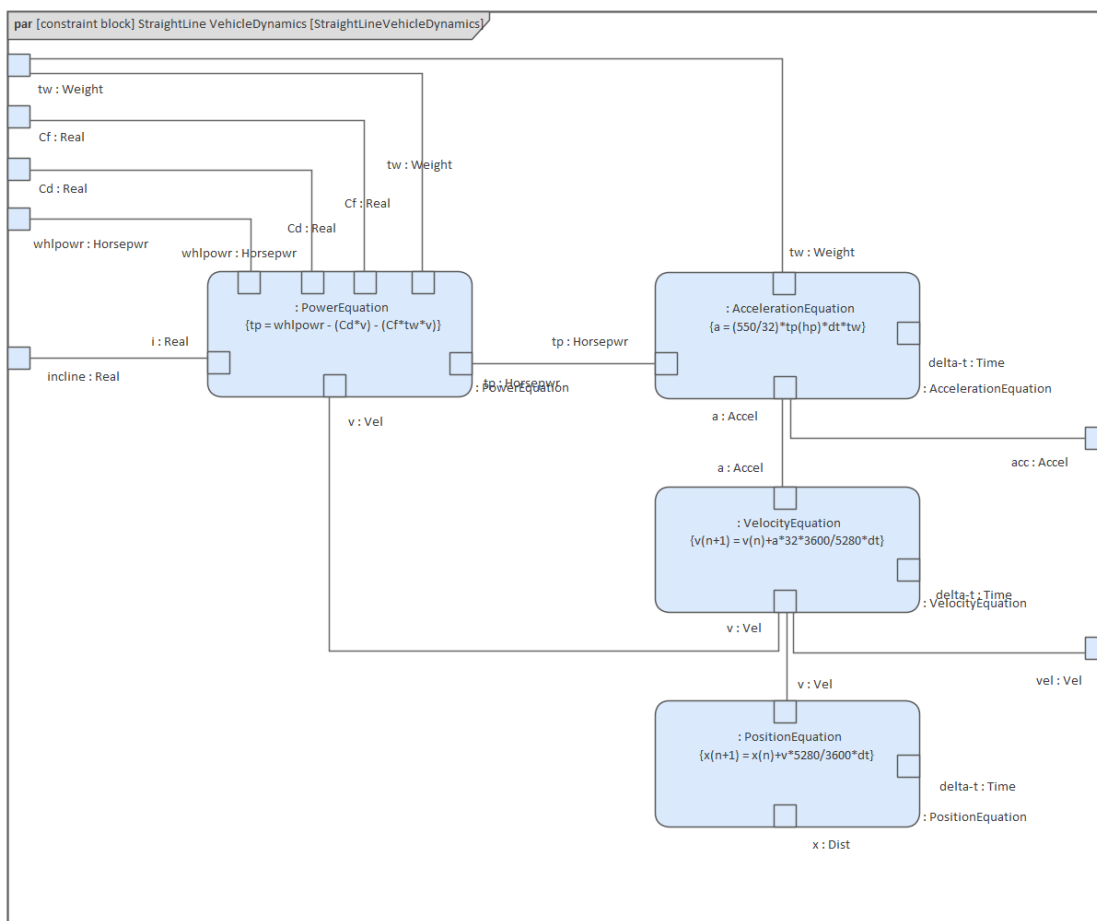
# Diagramme Paramétriques - Economie

Ce diagramme Paramétriques est utilisé pour modéliser l'économie de carburant, car elle a été définie comme une exigence de haut niveau importante pour le SUV et doit donc être évaluée pour garantir que le produit final fonctionnera correctement. Le diagramme Paramétriques contient des propriétés de contrainte (instances des blocs de contrainte). Il existe un certain nombre d'équations qui contribuent à l'évaluation de l'économie globale, notamment : l'équation de traînée aérodynamique, l'équation d'efficacité de freinage régénératif, la dynamique du véhicule en ligne droite, l'équation de frottement de roulement et l'équation d'efficacité énergétique. Les propriétés de contrainte contiennent des paramètres montés bordure qui sont connectés à d'autres paramètres par des connecteurs de liaison.



# Diagramme Paramétriques - Dynamique

Dans ce diagramme Paramétriques, le Bloc de contraintes - Dynamique des véhicules en ligne droite - de l'exemple précédent a été développé pour montrer comment il peut être modélisé avec un certain nombre de propriétés de contrainte. La contrainte Dynamique des véhicules en ligne droite est représentée par le cadre diagramme et les équations constitutives qui contribuent à l'équation globale sont modélisées dans le diagramme en tant que propriétés de contrainte. Chaque contrainte sur laquelle les propriétés de contrainte sont basées possède une équation de contrainte définie, qui est indiquée entre accolades {} sur le diagramme ; par exemple, l'équation d'accélération est définie dans le Bloc de contraintes comme  $\{a = (550/32)*tp(hp)*dt*tw\}$ . Les connecteurs de liaison sont utilisés pour relier les paramètres (variables) d'une équation aux paramètres (variables) d'une autre équation.



## Diagramme de synchronisation (non normatif) - accélération 100 ch

Enterprise Architect dispose d'une puissante capacité à générer des tracés de diagrammes Paramétriques à l'aide de son intégration OpenModelica ou Simulink. L'un des grands avantages de modélisation de systèmes physiques est de pouvoir analyser la manière dont un système se comporterait dans un contexte réel, sans avoir à construire des prototypes coûteux ou à effectuer le test sur le système construit lui-même. La capacité à modéliser les équations mathématiques qui régissent le fonctionnement d'un système et à créer des modèles de ces équations sous forme de contraintes à l'aide de diagrammes de définition Bloc et diagrammes Paramétriques constitue les précurseurs de la simulation de modèles.

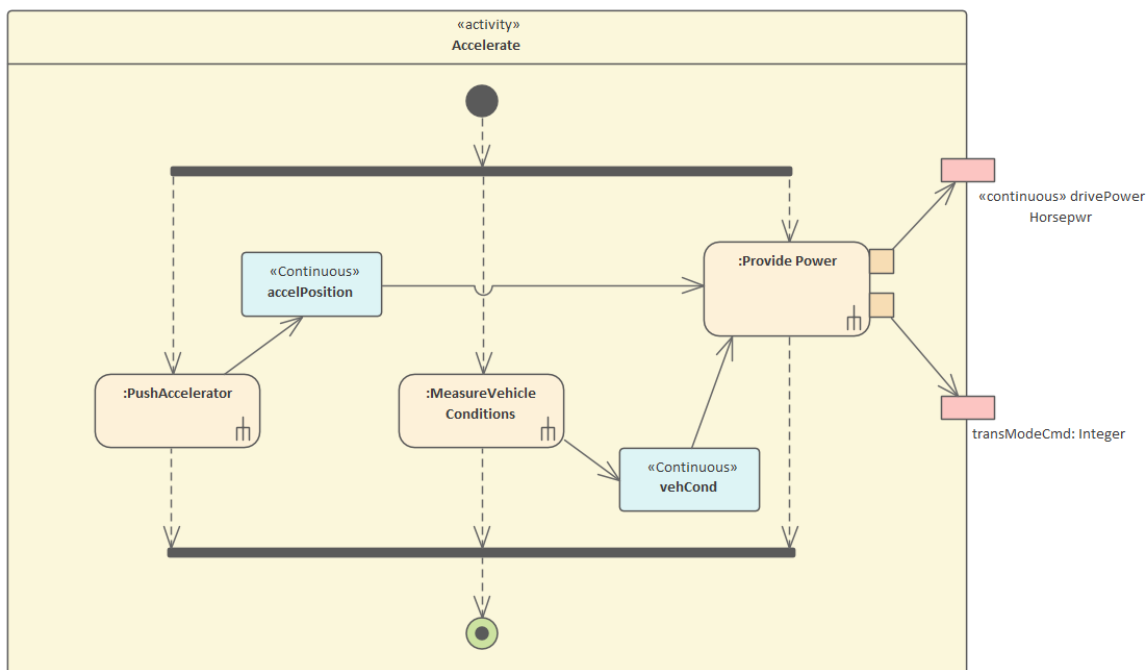
Enterprise Architect exploite la puissance d'un outil ouvert appelé OpenModelica, qui s'appuie sur le langage Modelica pour générer des tracés et des représentations graphiques d'équations en mouvement.

## Définition, décomposition et répartition des activités

Les exemples des rubriques de cette section utilisent diagrammes d'activité qui décrivent les aspects comportementaux du SUV hybride, à l'aide d'actions qui sont responsables de la définition du travail, qui est finalement effectué par des instances de blocs. Il existe également un certain nombre de diagrammes Bloc internes qui illustrent la manière dont les allocations peuvent être représentées.

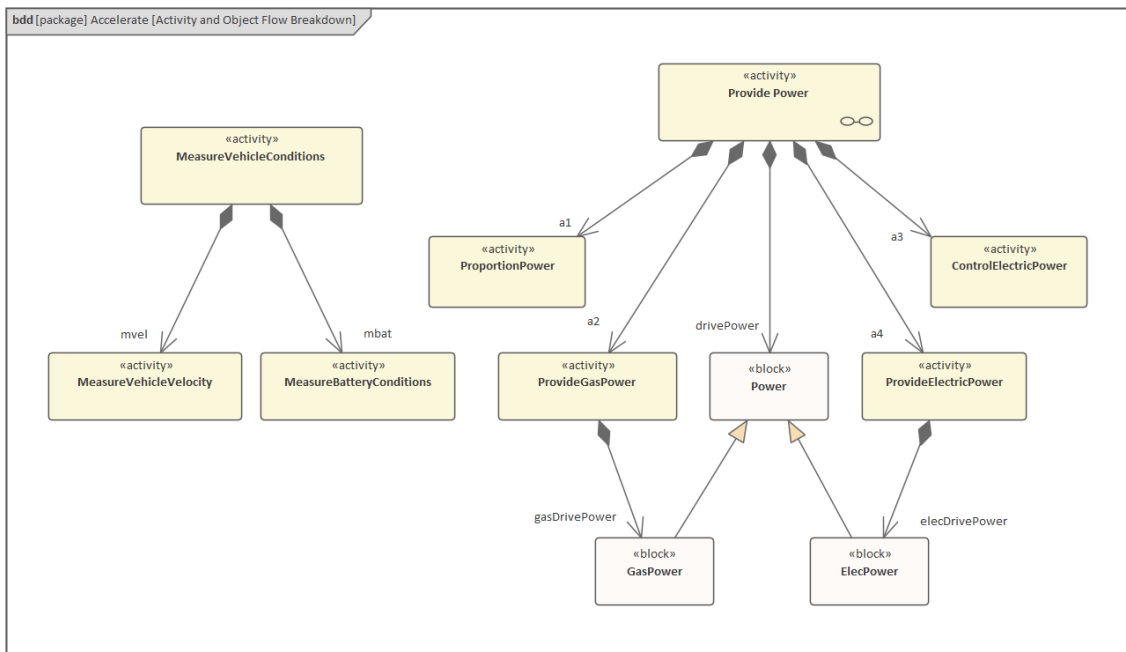
## Diagramme d'activité - Accélération (niveau supérieur)

Ce diagramme d'activité montre le comportement de niveau supérieur d'une activité représentant l'accélération du HSUV. Dans cet exemple, l'ingénieur système a pour objectif d'attribuer ce comportement à des parties du sous-système d'alimentation. On découvre cependant rapidement que le comportement tel qu'il est représenté ne peut pas être alloué et doit être décomposé davantage. Les stéréotypes sur les nœuds objet entre les actions de la figure s'appliquent aux paramètres des comportements ou des opérations appelés par les actions



# Interrompre lorsqu'une Variable Change de Valeur - Accélération

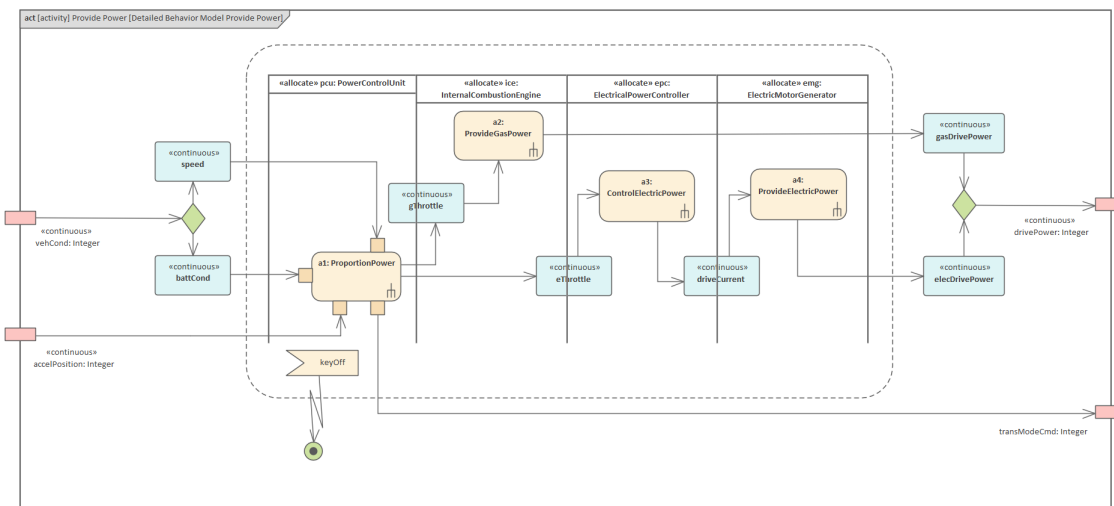
Ce diagramme de définition Bloc montre la décomposition de l'activité Fournir de l'énergie à partir du diagramme de la rubrique précédente. Il est important de noter qu'il s'agit d'une décomposition fonctionnelle et qu'en tant que telle, elle définit les relations structurelles entre les activités de la hiérarchie et doit donc être modélisée sur un diagramme de définition Bloc .



# Diagramme d'activité (EFFBD) - Accélération (détail)

Ce diagramme d'activité comporte l'activité Fournir de l'énergie comme cadre diagramme , qui inclut des actions invoquant les activités décomposées et les nœuds Object du diagramme précédent. Il utilise également des partitions d'activité d'allocation, qui sont orientées verticalement dans le diagramme . Ces partitions incluent l'unité de contrôle de l'énergie, le moteur à combustion interne, le contrôleur d'énergie électrique et le générateur d'énergie électrique, qui sont utilisés pour montrer quelle partie du système est responsable des actions définies dans le diagramme . Il existe également une légende d'allocation pour allouer explicitement des activités et un flux object aux parties du Bloc de sous-système d'énergie.

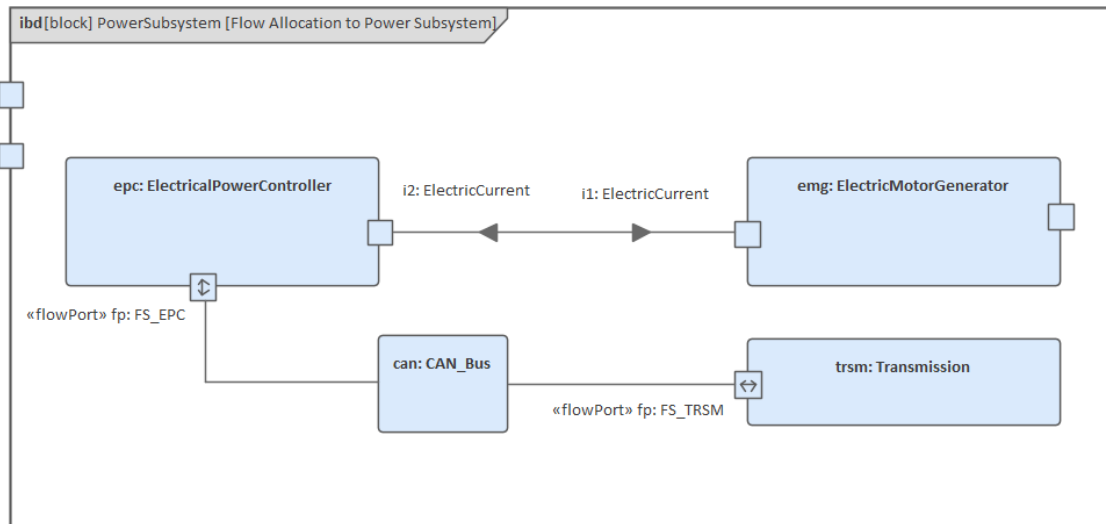
L'ingénieur modélisation a utilisé des flux object entrants et sortants pour l'activité ProvidePower. Cela a été fait pour distinguer le flux de puissance mécanique générée électriquement et de puissance mécanique générée par le gaz, et pour fournir un aperçu plus approfondi des conditions spécifiques du véhicule surveillé.





## Diagramme Interne de Bloc - Power Subsystem Comportementale and Flow Allocation

Ce diagramme Bloc internes partiellement complété développe certaines des relations d'allocation présentées dans l'exemple précédent. Nous voyons ici comment les blocs qui ont été ajoutés au diagramme en tant que propriétés communiquent entre eux, et nous pouvons voir le flux d'éléments d'une instance d'un Bloc à une autre. Plus précisément, le contrôleur d'énergie électrique est connecté au générateur de moteur électrique et nous pouvons voir que le courant électrique circule entre les deux propriétés.



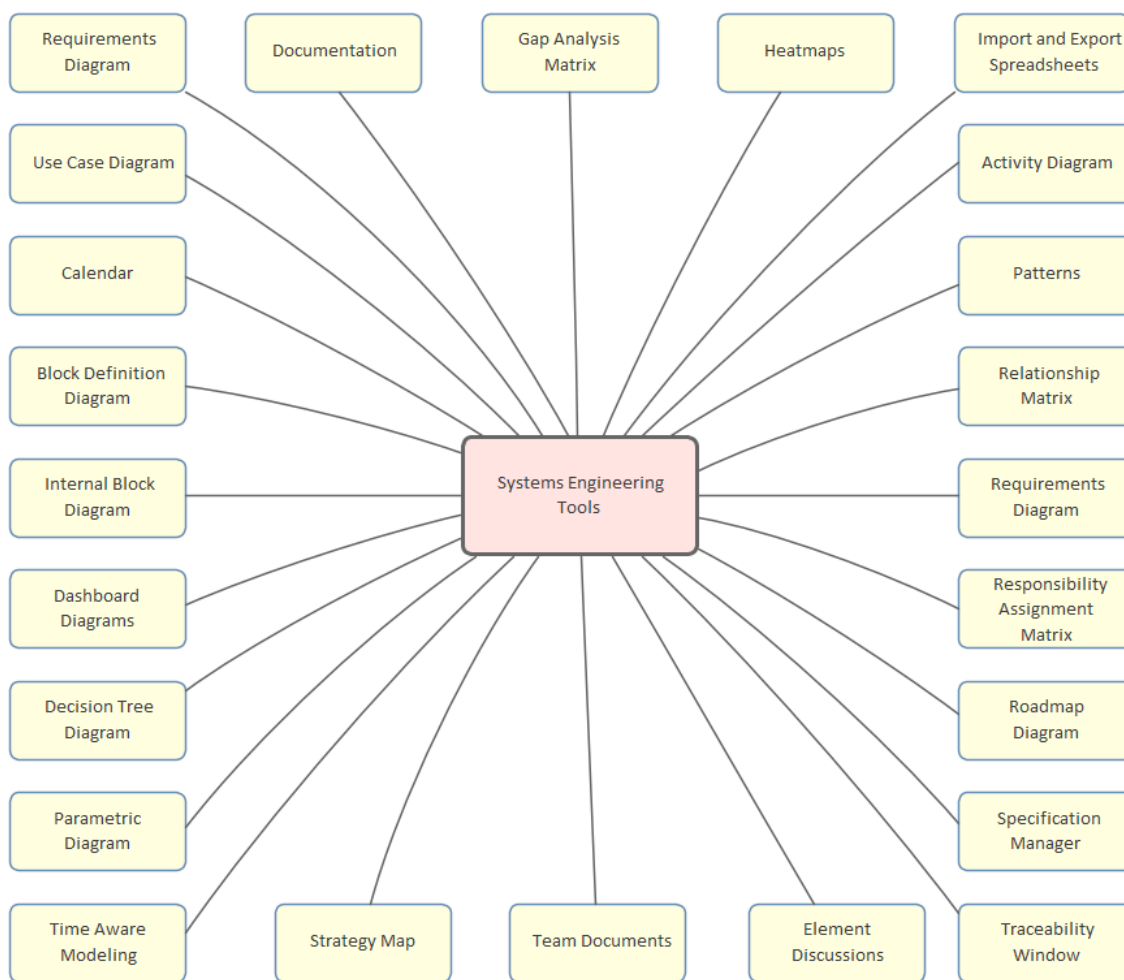
## Diagramme Interne de Bloc : Valeurs spécifiques à la propriété - Test d'économie de carburant de l'EPA

Ce diagramme Bloc internes illustre la manière dont les cas Test peuvent être modélisés dans le contexte de la représentation de sous-systèmes. Les tests ont été effectués pour une instance spécifique du SUV hybride qui vient vraisemblablement de sortir de la chaîne de production et qui possède un numéro d'identification du véhicule (VIN) spécifique. Le test qui a été effectué est le test d'économie de carburant de l'EPA. Les numéros de série de pièces spécifiques pertinentes peuvent également être indiqués pour identifier les problèmes que pourraient présenter les instances des pièces.

# Rencontrez les outils d' Ingénierie des Systèmes

Enterprise Architect est une plate-forme d'ingénierie système sophistiquée et flexible basée sur Modèle qui peut être utilisée à la fois comme référentiel et comme outil de gestion de projets d'ingénierie. Elle peut être utilisée tout au long du cycle de vie, de la mise en place d'un programme ou d'une pratique Ingénierie des Systèmes , en passant par la planification, la gestion, le développement et la documentation des efforts d'ingénierie, jusqu'à la gouvernance des projets de mise en œuvre qui consomment les conceptions et les résultats architecturaux. L'outil peut être utilisé avec n'importe quel processus et n'importe quel nombre de langages de représentation, y compris SysML, UML , ArchiMate ou BPMN. Il existe une large gamme de facilités et d'outils qui permettent à l'ingénieur de travailler en utilisant ses méthodes préférées, telles que les vues de traitement de texte, les vues de feuille de calcul, diagrammes , les matrices de relations et une gamme d'autres fonctionnalités de base et étendues.

Cette carte mentale présente le paysage des principaux outils Ingénierie des Systèmes qui peuvent être utilisés pour mettre en place et maintenir un certain nombre d'initiatives Ingénierie Systèmes Modèles Basée . Bien qu'il s'agisse des principaux outils, il existe une gamme d'autres outils que les équipes ou les ingénieurs individuels trouveront utiles ; ceux-ci peuvent être explorés via le Guide de l'utilisateur.

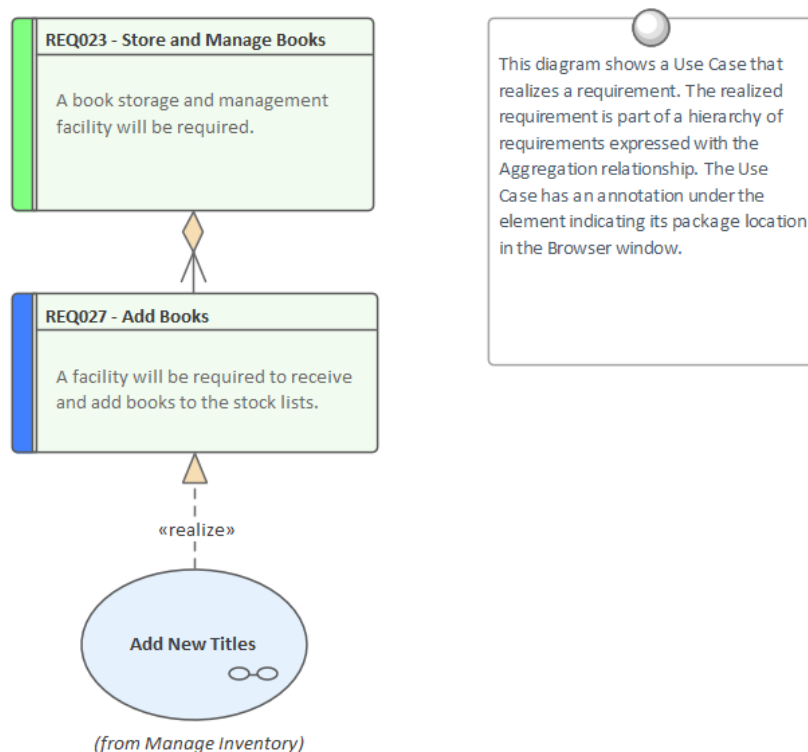


# Diagramme Exigences


## Connaître le Diagramme Exigences

### Présentation du Diagramme Exigences

Le diagramme Exigences fournit une représentation visuelle de la relation entre Exigences et avec d'autres éléments du modèle, notamment les facteurs Métier , les contraintes, Règles Métier , les cas d'utilisation, les histoires d'utilisateur, les composants de conception, etc. Le diagramme est l'un des types diagramme étendus d' Enterprise Architect . Il fournit une représentation graphique attrayante des Exigences , qui constituera un changement bienvenu pour les analystes Exigences habitués à travailler avec des outils textuels.



### Où trouver le Diagramme Exigences

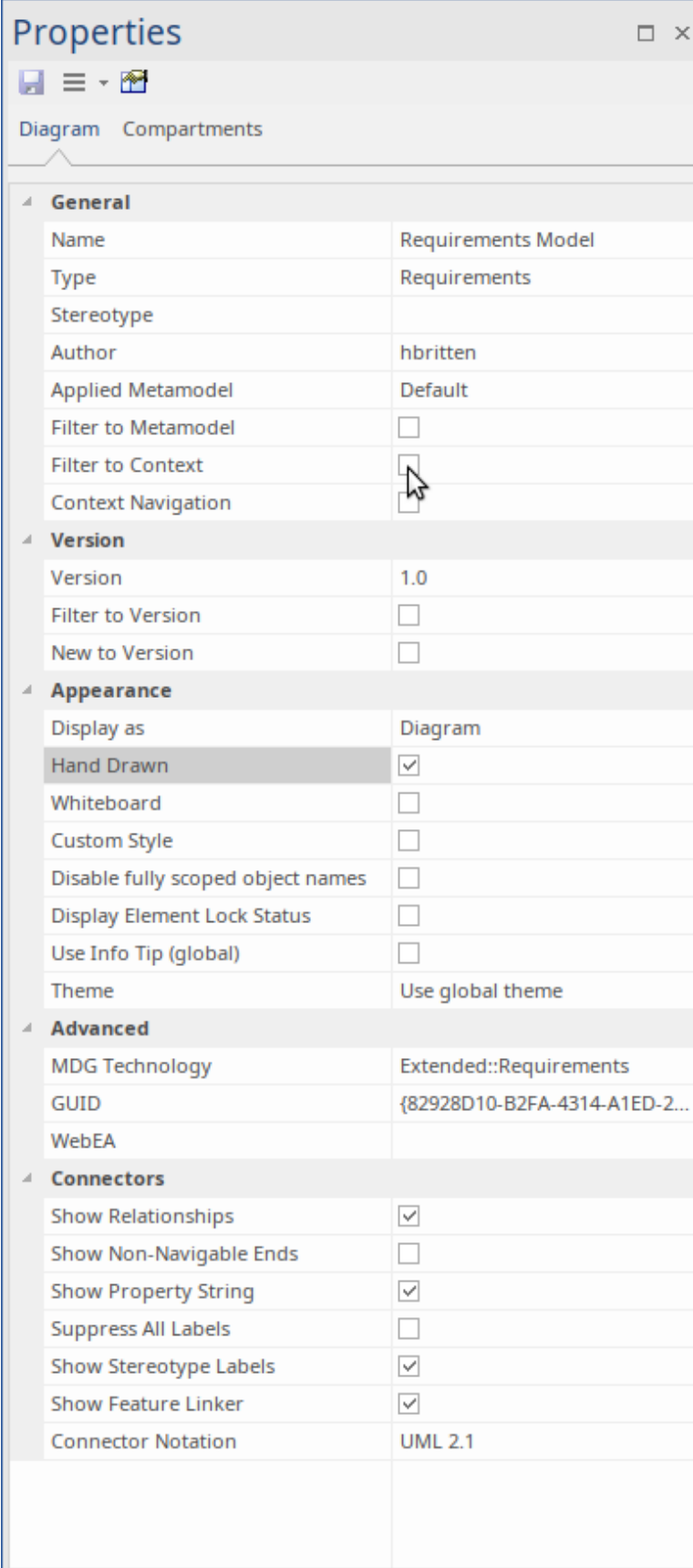
Fenêtre Navigateur Menu Contexte : Ajouter Diagramme :  | Gérer | Afficher toutes Perspectives | Étendu | Exigences

### Utilisation du Diagramme Exigences

L'une des utilisations est de montrer comment Exigences sont connectées entre elles dans une hiérarchie ou, plus important encore, comment Exigences sont connectées à d'autres éléments. Le modélisateur expérimenté définira et gèrera les Exigences dans le Gestionnaire de Spécification , puis utilisera le diagramme Exigences pour montrer comment chaque exigence est liée aux éléments de processus en amont tels que les pilotes Métier et aux éléments de processus en aval tels que les cas d'utilisation, les récits d'utilisateurs, les conceptions d'expérience utilisateur et les composants de solution.

### Options pour le Diagramme Exigences

L'apparence d'un diagramme peut être modifiée pour s'adapter au public visé et des détails peuvent être inclus, supprimés ou modifiés pour garantir que le diagramme réponde à son objectif principal de communication. Il existe un large éventail d'options, allant de la création d'un diagramme de style dessiné à la main au filtrage du contenu diagramme .



The screenshot shows the 'Properties' window for a 'Requirements Model' diagram. The window is divided into several sections: General, Version, Appearance, Advanced, and Connectors. The 'General' section includes fields for Name, Type, Stereotype, Author, Applied Metamodel, Filter to Metamodel, Filter to Context, and Context Navigation. The 'Version' section includes Version, Filter to Version, and New to Version. The 'Appearance' section includes Display as, Hand Drawn, Whiteboard, Custom Style, Disable fully scoped object names, Display Element Lock Status, Use Info Tip (global), and Theme. The 'Advanced' section includes MDG Technology, GUID, and WebEA. The 'Connectors' section includes Show Relationships, Show Non-Navigable Ends, Show Property String, Suppress All Labels, Show Stereotype Labels, Show Feature Linker, and Connector Notation.

General	
Name	Requirements Model
Type	Requirements
Stereotype	
Author	hbritten
Applied Metamodel	Default
Filter to Metamodel	<input type="checkbox"/>
Filter to Context	<input type="checkbox"/>
Context Navigation	<input type="checkbox"/>

Version	
Version	1.0
Filter to Version	<input type="checkbox"/>
New to Version	<input type="checkbox"/>

Appearance	
Display as	Diagram
Hand Drawn	<input checked="" type="checkbox"/>
Whiteboard	<input type="checkbox"/>
Custom Style	<input type="checkbox"/>
Disable fully scoped object names	<input type="checkbox"/>
Display Element Lock Status	<input type="checkbox"/>
Use Info Tip (global)	<input type="checkbox"/>
Theme	Use global theme

Advanced	
MDG Technology	Extended::Requirements
GUID	{82928D10-B2FA-4314-A1ED-2...}
WebEA	

Connectors	
Show Relationships	<input checked="" type="checkbox"/>
Show Non-Navigable Ends	<input type="checkbox"/>
Show Property String	<input checked="" type="checkbox"/>
Suppress All Labels	<input type="checkbox"/>
Show Stereotype Labels	<input checked="" type="checkbox"/>
Show Feature Linker	<input checked="" type="checkbox"/>
Connector Notation	UML 2.1

**Apprenez Plus sur le  
Diagramme Exigences**

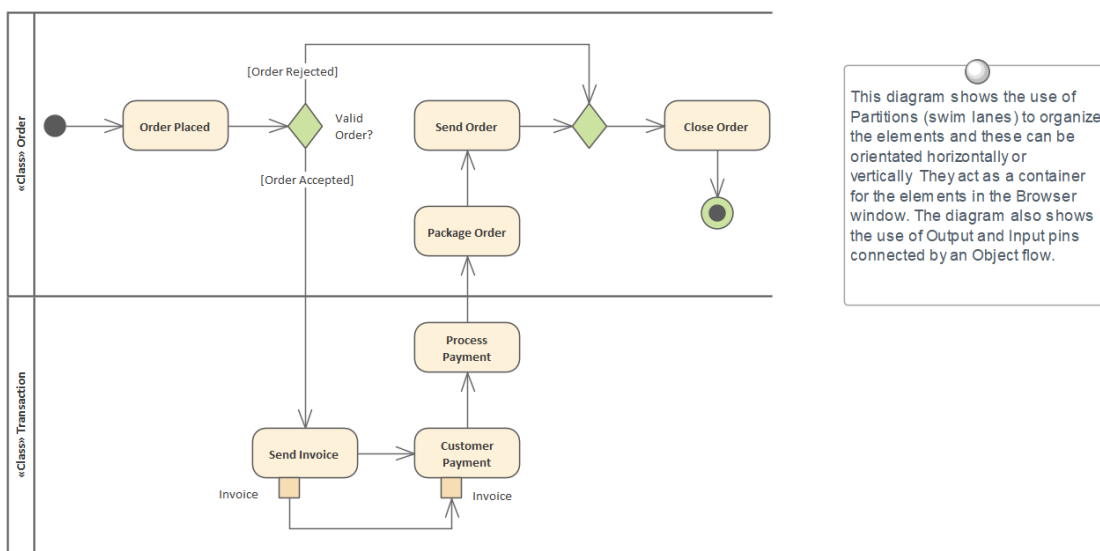
[Working In Diagrams](#)



# Diagramme d'activité

Le diagramme d'activité est l'un des diagrammes Comportementale du Unified Modeling Language (UML) qui peut être utilisé pour modéliser un processus ou un algorithme sous forme d'une séquence d'étapes. Il s'agit d'une version plus sophistiquée de son proche cousin le diagramme de flux. Les diagrammes d'activité peuvent être utilisés pour modéliser un processus métier comme une alternative UML au diagramme Processus Métier ; ils ont la même capacité à créer une hiérarchie d'activités dans la fenêtre Navigateur.

Activity Diagram showing the use of Partitions



Les éléments peuvent être nommés et des descriptions détaillées peuvent être ajoutées aux Notes. En connectant les Activités, les Décisions et les Forks avec des connecteurs (Flux de contrôle), une séquence d'éléments peut décrire le processus métier. Une hiérarchie de processus peut être construite en imbriquant des Activités dans la fenêtre Navigateur et en utilisant la fonctionnalité diagramme enfant pour permettre une analyse en profondeur du niveau de la chaîne valeur jusqu'aux processus de niveau le plus bas.

## Connaître le Diagramme d'activité

### Où trouver le Diagramme d'activité

Ruban : Conception > Diagramme > Ajouter Diagramme > UML > Comportementale > UML Comportementale > Activité

Barre d'outils de la fenêtre Navigateur : Icône Nouveau Diagramme > UML > Comportementale > UML Comportementale > Activité

Menu contextuel de la fenêtre Navigateur | Ajouter Diagramme ... > Comportementale UML > Activité

### Utilisation du Diagramme d'activité

Le diagramme d'activité peut être utilisé pour modéliser toute activité ou notion commerciale ou technique comportant une série d'étapes. Cela inclut les processus commerciaux et techniques ainsi que les algorithmes informatiques. Les étapes sont reliées par des relations de flux de contrôle qui montrent le séquençage des étapes. Les décisions et les fusions peuvent être utilisées pour modéliser le choix et pour contrôler davantage le flux à travers l'activité. Des fourches et Jointures peuvent être ajoutées pour diviser et réunir le flux de contrôle et des objets ajoutés pour montrer comment les données sont fournies et consommées.

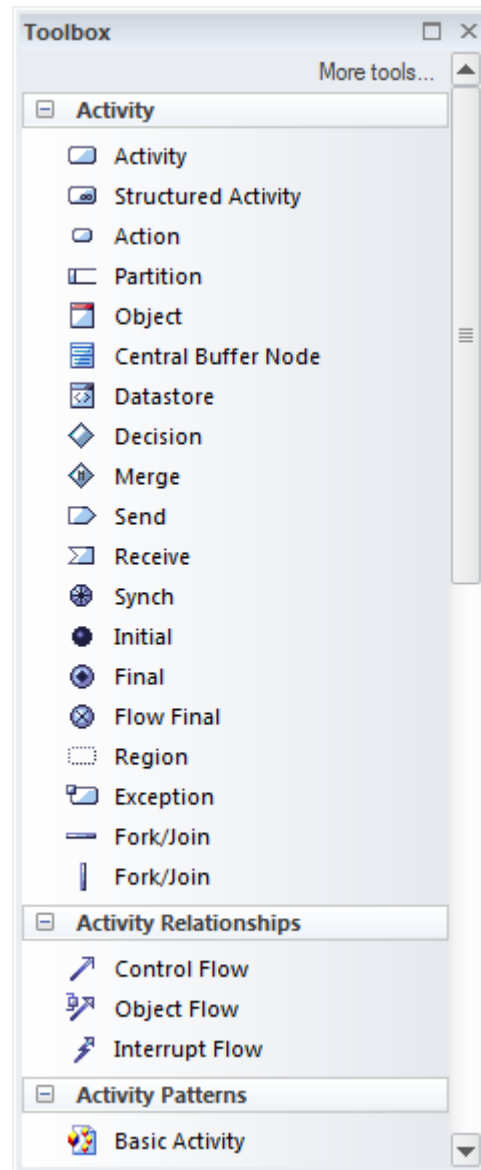
### Options pour le Diagramme d'activité

Les diagrammes d'activité peuvent être dessinés à différents niveaux de formalité, depuis un diagramme de style organigramme de base utilisé pour représenter un

simple Processus Métier jusqu'à un diagramme sophistiqué basé sur l'action qui peut être utilisé pour modéliser un système complexe. Il existe une boîte à outils qui contient une gamme d'éléments, de relations et Motifs pour créer les modèles.

Le diagramme d'activité (comme tout diagramme ) peut être visualisé comme une liste d'éléments, ce qui facilite le travail avec les propriétés des éléments.

Filtres de Diagramme peuvent également être utilisés lors de la présentation des diagrammes , pour attirer l'attention sur certaines parties des diagrammes , et les diagrammes peuvent être présentés sous forme dessinée à la main ou sous forme de tableau blanc en modifiant les propriétés du diagramme .



**Apprenez Plus sur le  
Diagramme d'Activités**

[Activity Diagram](#)

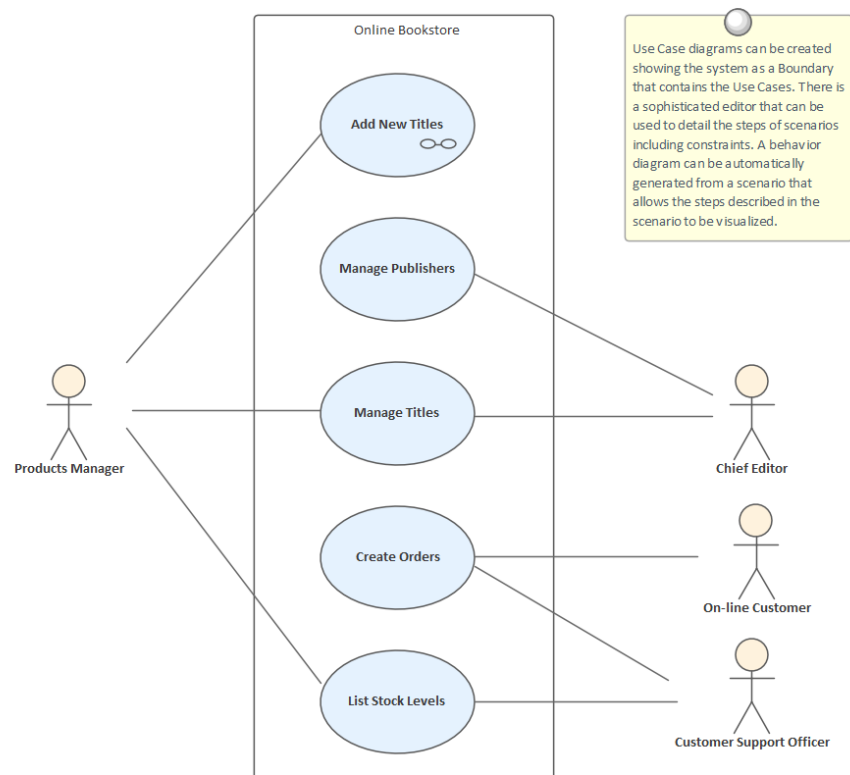


# Diagramme de cas d'utilisation

## Apprendre à connaître le diagramme de cas d'utilisation

### Présentation du Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation est l'un des diagrammes Comportementale du Unified Modeling Language (UML) qui peut être utilisé pour décrire les objectifs des utilisateurs et des autres systèmes qui interagissent avec le système modélisé. Ils sont utilisés pour décrire les exigences fonctionnelles d'un système, d'un sous-système ou d'une entité et présentent une image simple mais convaincante de la manière dont le système sera utilisé.



Elles sont généralement utilisées en conjonction avec les Exigences Métier et celles des parties prenantes de niveau supérieur et sont souvent complétées par un ensemble d' Exigences non fonctionnelles.

### Où trouver le Diagramme de cas d'utilisation

Ruban : Conception > Diagramme > Ajouter un diagramme > Comportementale UML > Cas d'utilisation

Barre d'outils de la fenêtre Navigateur : icône Nouveau Diagramme > UML Comportementale > Use Case

Menu contextuel de la fenêtre Navigateur | Ajouter Diagramme ... > Comportementale UML > Cas d'utilisation

### Utilisation du Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation est utilisé pour décrire les objectifs que les utilisateurs ou d'autres systèmes souhaitent atteindre en interagissant avec le système. Ils décrivent toujours l'objectif du point de vue des acteurs, les détails du

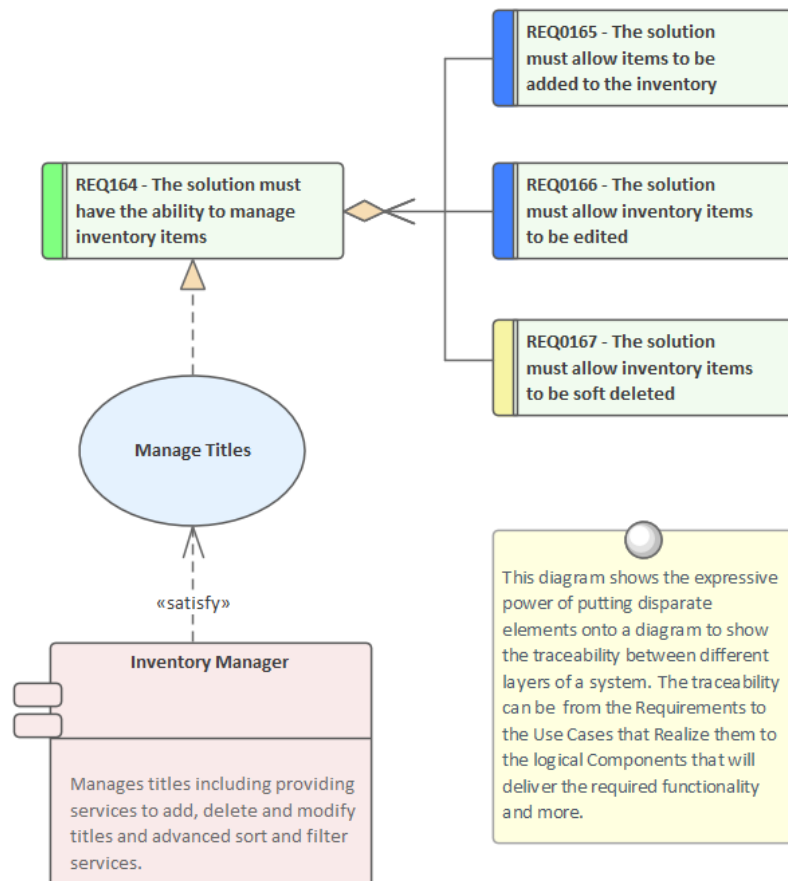
cas d'utilisation décriront l'objectif avec plus de précision.

Les cas d'utilisation serviront souvent de base à la définition des cas Test .

**Options pour le Diagramme de cas d'utilisation**

Il est possible de créer un nombre illimité de diagrammes de cas d'utilisation pour représenter différentes parties d'un système ou Paquetages de cas d'utilisation. Les diagrammes peuvent être simples ou structurés par l'application d'un certain nombre de connecteurs supplémentaires tels que les relations d'inclusion, d'extension et de généralisation.

Une Bordure système peut être incluse et utilisée pour nommer le système, le sous-système ou l'entité en discussion ; les acteurs se trouvent à l'extérieur de la Bordure et les cas d'utilisation à l'intérieur.



diagrammes de cas d'utilisation peuvent être utilisés pour montrer comment les cas d'utilisation sont liés à d'autres éléments du système, y compris les éléments en amont tels que Exigences et les éléments en aval tels que les composants.

Le diagramme de cas d'utilisation (comme pour tout diagramme ) peut être considéré comme une liste d'éléments, ce qui facilite le travail avec les propriétés de l'élément.

Filtres de Diagramme et les calques Diagramme peuvent également être utilisés lors de la présentation des diagrammes , pour attirer l'attention sur certaines parties des diagrammes et les diagrammes peuvent être présentés comme dessinés à la main ou dans un style tableau blanc en modifiant les propriétés du diagramme .

**Apprenez Plus sur le Diagramme de Cas d'Utilisation**

[Use Case Diagram](#)

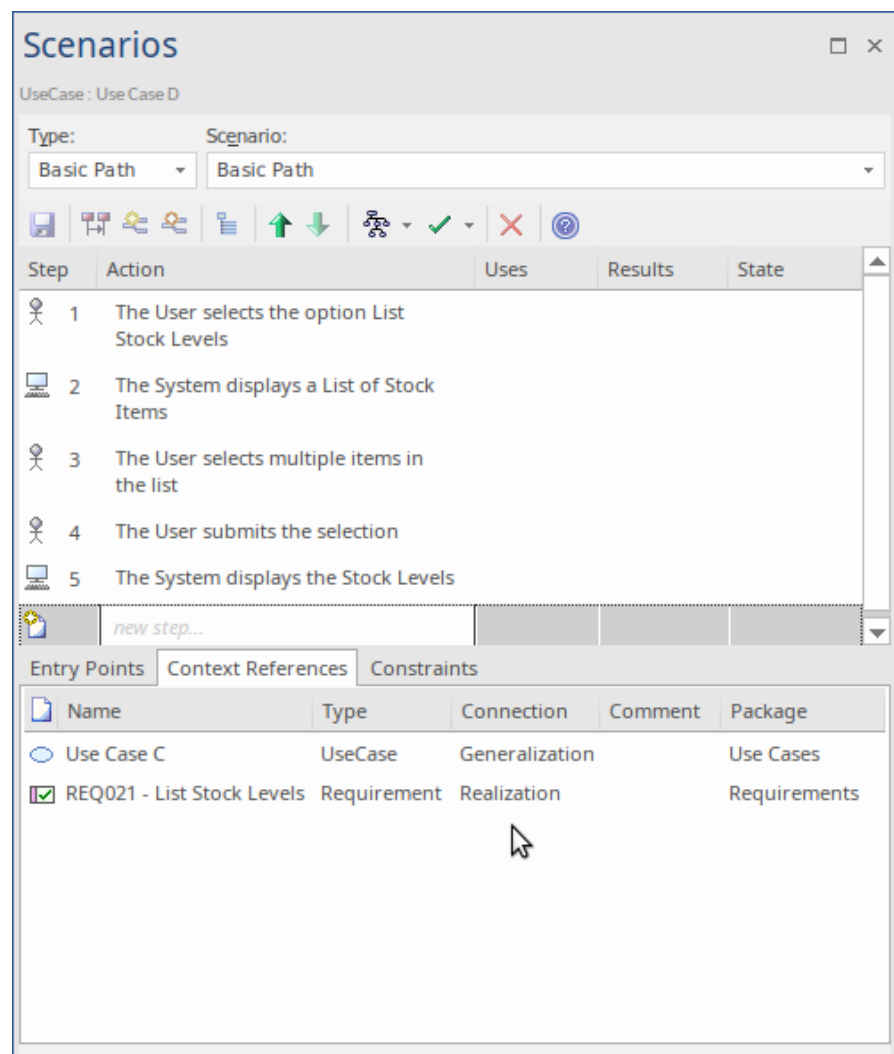


# Générateur de scénarios

## Découvrir le générateur de scénarios

### Présentation du générateur de scénarios

Le générateur de scénarios permet de définir les détails d'un cas d'utilisation, notamment la définition de descriptions détaillées, la création d'un ou plusieurs scénarios et la définition de conditions préalables, de conditions postérieures et d'autres contraintes. Les étapes détaillées d'un cas d'utilisation peuvent être enregistrées et liées à d'autres éléments du modèle, puis générées sous forme de diagramme fournissant une représentation visuelle du cas d'utilisation et de ses scénarios. Le diagramme et le texte peuvent être synchronisés et les étapes individuelles peuvent ensuite être reliées à d'autres éléments tels que les composants qui réaliseront l'exigence spécifiée dans le cas d'utilisation.



### Où trouver le générateur de scénarios

Démarrer > Application > Modifier > Responsabilités > Scénarios structurés  
 Conception > Élément > Éditeurs > Responsabilités > Scénarios structurés  
 Menu Contexte élément : Propriétés | Responsabilités > Scénarios | clic droit |  
 Ajouter nouveau : éditeur de structure

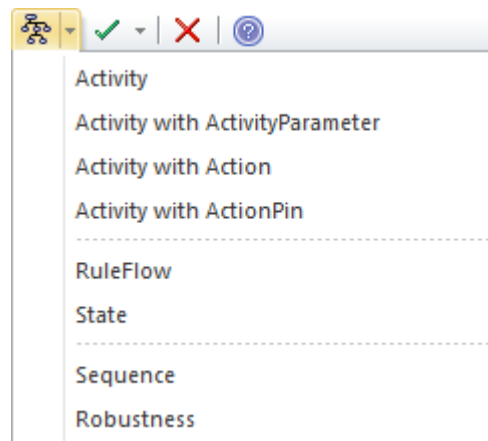
### Utilisation du générateur

Pour définir les détails d'un cas d'utilisation, ses scénarios et ses contraintes, qui

**de scénarios** peuvent être utilisés pour remplacer l'approche traditionnelle basée sur des documents texte pour définir les cas d'utilisation. Cela garantit que le diagramme de cas d'utilisation et les détails textuels des cas d'utilisation, de leurs scénarios et de leurs contraintes sont tous contenus dans le même modèle et peuvent être tracés. Si les cas d'utilisation sont requis dans un format de document pour des raisons contractuelles ou de processus, un Rapport de cas d'utilisation peut être généré automatiquement à partir des modèles à l'aide du moteur de documentation intégré.

**Options pour le générateur de scénarios**

Le Générateur de Scénarios peut être visualisé sous forme de fenêtre à onglets ou ancrée ou dans la fenêtre Propriétés d'un élément. Les étapes d'un Cas d'Utilisation, y compris ses Scénarios, peuvent être générées automatiquement dans un certain nombre de types de diagramme différents disponibles à partir de l'icône Générer Diagramme de la barre d'outils.



**Apprenez Plus sur le générateur de scénarios**

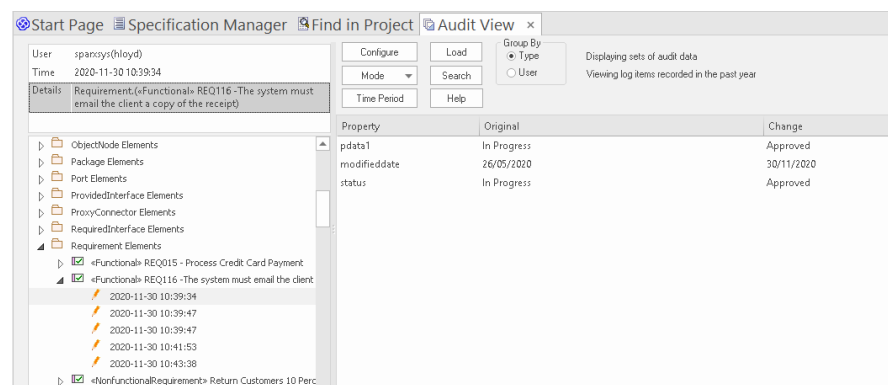
[Scenarios](#)

# Audit

## Découvrir l'audit

### Présentation de l'audit

La fonctionnalité Audit permet de suivre les modifications apportées aux Exigences notamment ce qui a été modifié, quand et par qui. L'audit est désactivé par défaut et doit être activé avant que les modifications apportées aux exigences ne soient enregistrées. Une fois activé, il s'agit d'un outil passif qui enregistre silencieusement les modifications apportées aux éléments. Il ne remplace pas Contrôle de Version ou Lignes de base et, contrairement à ces outils, il ne peut pas être utilisé pour revenir à un état antérieur du modèle. La gestion des changements, la gouvernance et le contrôle qualité sont tous facilités par l'utilisation de l'audit.



### Où trouver l'audit

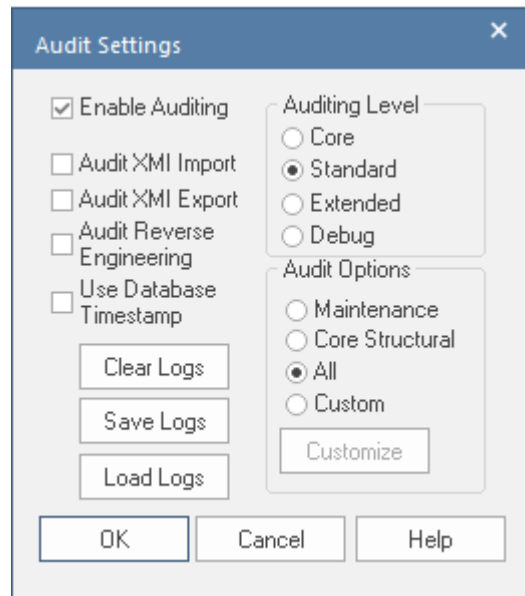
Ruban : Paramètres > Modèle > Audit

### Utilisation de l'audit

L'audit peut être utilisé pour suivre les modifications apportées à un modèle, la personne qui les a apportées et à quel moment. Il existe un certain nombre de modes et un administrateur de référentiel peut utiliser les paramètres pour spécifier ce qui est enregistré dans l'audit. Alors qu'une ligne de base peut être utilisée pour montrer la différence entre un modèle et un instantané à un moment donné, l'outil d'audit enregistre chaque modification individuelle ; il ne peut cependant pas être utilisé pour revenir à un état antérieur.

### Options d'audit

Il existe une large gamme de paramètres permettant de configurer l'audit, en commençant par l'activation ou la désactivation des paramètres qui déterminent les éléments qui ont une piste d'audit et le niveau de détail enregistré. Les journaux d'audit peuvent être exportés depuis le référentiel pour augmenter les performances.



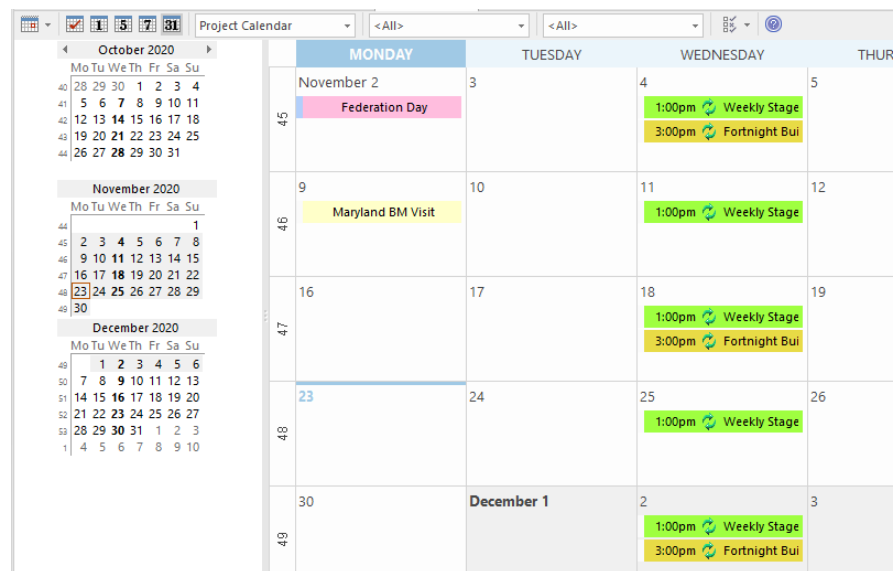
Apprenez Plus sur l'audit [Auditing](#)

# Calendrier

## Découvrir le calendrier

### Présentation du calendrier

Le calendrier est un mécanisme complet permettant d'enregistrer les événements importants d'une initiative et d'afficher d'autres informations telles que l'allocation des ressources. Il existe des vues journalières, hebdomadaires et mensuelles et l'affichage peut être configuré pour afficher les entrées du calendrier, les tâches du projet et l'allocation des ressources. Lorsqu'une ressource a été alloué - par exemple pour analyser un ensemble d'exigences - un utilisateur peut accéder à l'emplacement des exigences dans la fenêtre Navigateur à partir du calendrier.



Il existe également des types d'événements, des catégories et des couleurs entièrement configurables. Le travail d'un Analyste Métier impliquera une large gamme d'événements, notamment des ateliers, des entretiens, des groupes de discussion, des jeux collaboratifs, des séances de brainstorming, révisions, des observations et des réunions. Tous ces événements peuvent être facilement enregistrés et gérés dans le calendrier. Lorsque des ressources ont été alloué à des éléments et des tâches ont été assignées à des individus, elles peuvent être affichées dans le calendrier.

### Où trouver le calendrier

Ruban : Démarrer > Collaborer > Calendrier

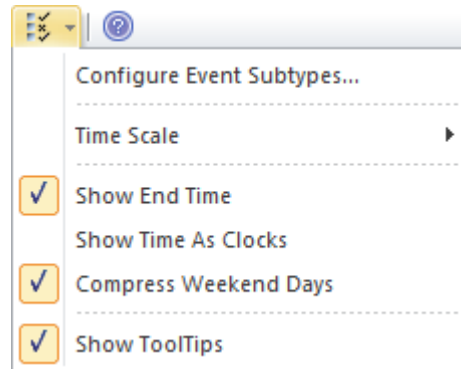
### Utilisation du calendrier

Le calendrier peut être utilisé pour planifier et visualiser des événements tels que des réunions, des jalons, révisions, des ateliers et bien plus encore. Il peut être utilisé pour visualiser l'affectation des ressources aux éléments du référentiel, par exemple pour savoir qui analyse un ensemble d'exigences. Il peut également être utilisé pour visualiser les tâches du projet. Un analyste peut facilement cliquer sur les éléments de la fenêtre Navigateur ou sur les tâches du projet.

### Options pour le calendrier

Le calendrier propose un certain nombre d'options, notamment la possibilité de créer des événements récurrents. Une icône de la barre d'outils d'options permet de configurer certains aspects de l'apparence du calendrier.





**Apprenez Plus sur le  
Calendrier**

[The Model Calendar](#)

# Interrompre lorsqu'une Variable Change de Valeur

## Connaître l' Interrompre lorsqu'une Variable Change de Valeur

### Introducing the Interrompre lorsqu'une Variable Change de Valeur

Le diagramme de définition Bloc est l'un des diagrammes structurels du Systems Modeling Language (SysML) qui peut être utilisé pour modéliser une grande variété d'objets. Il s'agit d'un diagramme à usage général pour modélisation d'entités dans les domaines commerciaux et techniques, y compris les termes et concepts, Règles Métier et les capacités dans les schémas XML et de base de données.

### Où trouver l' Interrompre lorsqu'une Variable Change de Valeur

Ruban : Conception > Diagramme > Ajouter Diagramme : Type = Ingénierie des Systèmes > SysML Select From = SysML 1.5 Types Diagramme = Définition Bloc > <type requis>

Fenêtre Navigateur Barre d'outils : Icône Nouveau Diagramme : Type = Ingénierie des Systèmes > SysML Select From = SysML 1.5 Types Diagramme = Définition Bloc > <type requis>

Fenêtre Navigateur Menu Contexte | Ajouter Diagramme : Type = Ingénierie des Systèmes > SysML Select From = SysML 1.5 Types Diagramme = Définition Bloc > <type requis>

### Utilisation de l' Interrompre lorsqu'une Variable Change de Valeur

Le diagramme de définition Bloc peut être utilisé chaque fois qu'une représentation logique ou structurelle d'un système est requise. Il est applicable à modélisation de concepts commerciaux et d'ingénierie. Il s'agit du diagramme fondamental pour modélisation de la structure d'un système ou d'un sous-système ou de l'un de ses composants.

### Options pour l' Interrompre lorsqu'une Variable Change de Valeur

La définition Bloc (comme tout diagramme ) peut être considérée comme une liste d'éléments, ce qui facilite le travail avec les propriétés de l'élément.

Filtres de Diagramme peuvent également être utilisés lors de la présentation des diagrammes pour attirer l'attention sur certaines parties des diagrammes et les diagrammes peuvent être présentés comme dessinés à la main ou dans un style tableau blanc en modifiant les propriétés du diagramme .

### Apprenez Plus sur l' Interrompre lorsqu'une Variable Change de Valeur

[Using Blocks to Model Structure and Constraints](#)

# Diagramme Interne de Bloc

## Connaître le Diagramme Interne de Bloc

### Présentation du Diagramme Interne de Bloc

Le diagramme Bloc interne est utilisé en conjonction avec l' Interrompre lorsqu'une Variable Change de Valeur , mais il est généralement utilisé pour montrer la structure interne d'un Bloc y compris ses parties et comment elles fonctionnent ensemble pour fournir les comportements spécifiés par le bloc ou qui lui ont été alloué .

### Où trouver le Diagramme Interne de Bloc

Ruban : Conception > Diagramme > Ajouter Diagramme : Type = Ingénierie des Systèmes > SysML Select From = SysML 1.5 Types Diagramme = Bloc interne > <type requis>

Fenêtre Navigateur Barre d'outils : Icône Nouveau Diagramme : Type = Ingénierie des Systèmes > SysML Select From = SysML 1.5 Types Diagramme = Bloc interne > <type requis>

Fenêtre Navigateur Menu Contexte | Ajouter Diagramme : Type = Ingénierie des Systèmes > SysML Select From = SysML 1.5 Types Diagramme = Bloc interne > <type requis>

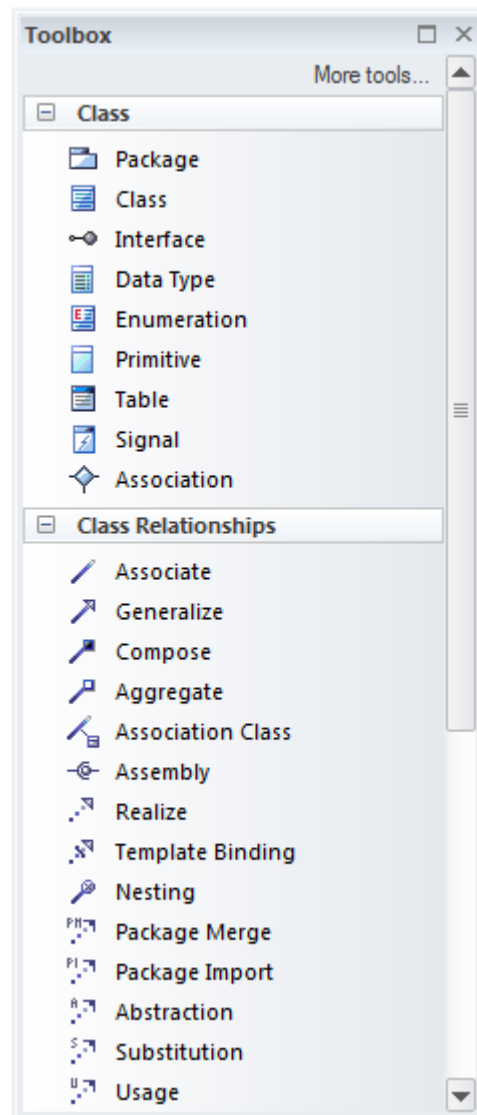
### Utilisation du Diagramme Interne de Bloc

Le diagramme Bloc interne est utilisé pour modéliser la structure interne d'un bloc, y compris ses parties et la relation entre ces parties.

### Options pour le Diagramme Interne de Bloc

Le diagramme Bloc internes (comme tout diagramme ) peut être considéré comme une liste d'éléments, ce qui facilite le travail avec les propriétés de l'élément.

Filtres de Diagramme peuvent également être utilisés lors de la présentation des diagrammes pour attirer l'attention sur certaines parties des diagrammes et les diagrammes peuvent être présentés comme dessinés à la main ou dans un style tableau blanc en modifiant les propriétés du diagramme .

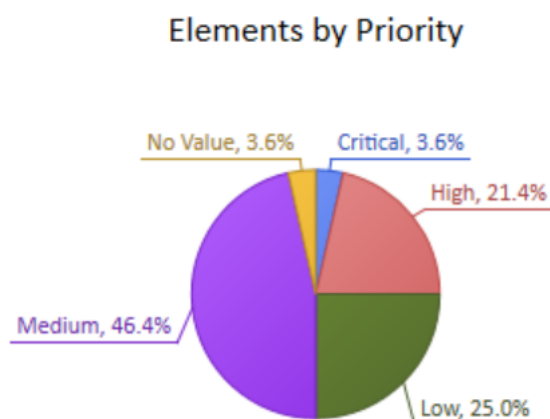


**Apprenez Plus sur le  
Diagramme Interne de  
Bloc**

[Using Properties and Parts to Model Block Usage](#)

## Diagrammes du tableau de bord

Diagrammes de tableau de bord vous permettent de créer des graphiques et des diagrammes de haute qualité pour afficher les informations du référentiel de manière visuellement attrayante. Ce diagramme est un exemple de création d'un **diagramme** de tableau de bord dans Sparx Systems Enterprise Architect ; il illustre le ratio des priorités des exigences dans un graphique à secteurs.




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.

Enterprise Architect fournit une page de boîte à outils contenant des graphiques et des diagrammes préconfigurés, mais vous êtes libre de créer et d'enregistrer autant de graphiques que vous le souhaitez, en puisant des données n'importe où dans le référentiel. Les graphiques et les diagrammes fournissent des informations récapitulatives précieuses qui aident à la gestion des Exigences. Les rapports de haut niveau et l'état du projet peuvent être facilement suivis et documentés à l'aide des nombreux éléments de graphiques et de rapports disponibles, qui sont étroitement liés au contenu et à l'état du modèle.

### Découvrir Diagrammes du tableau de bord

#### Où trouver Diagrammes du tableau de bord

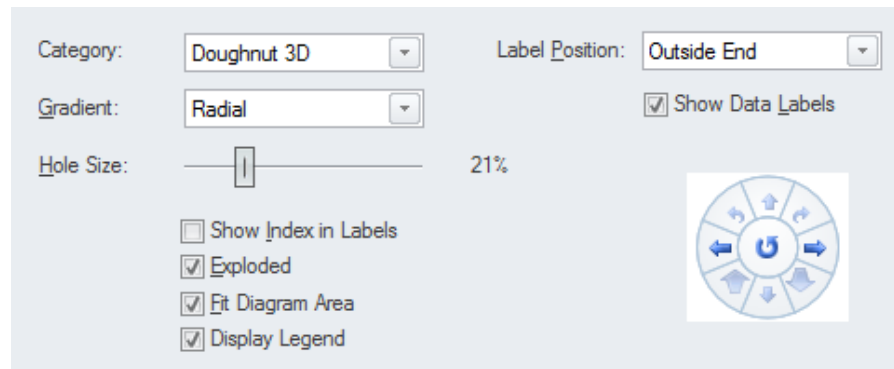
Fenêtre Navigateur | Cliquez-droit sur Paquetage | Ajouter Diagramme :  | Gérer | Afficher toutes Perspectives | Étendu | Tableau de bord

#### Utilisation des Diagrammes de tableau de bord

Diagrammes du tableau de bord présentent des vues d'informations riches mais faciles à comprendre (telles que le statut des Exigences dans une version particulière du système) qui peuvent être ouvertes dans le modèle ou copiées directement dans les présentations de la direction ou de l'équipe de projet. Ils sont utiles pour planifier une itération telle qu'un sprint Agile afin de voir dans quelle mesure les Exigences sont prêtes pour l'équipe de mise en œuvre ; par exemple, pour voir quel pourcentage des Exigences ont été approuvées et sont hautement prioritaires.

#### Options pour Diagrammes de tableau de bord

Les graphiques et diagrammes standards disponibles dans la Boîte à outils peuvent être configurés de plusieurs manières, notamment en modifiant la source, en appliquant des filtres ou en modifiant l'apparence du graphique comme indiqué dans ce diagramme, disponible dans la fenêtre Propriétés du graphique à l'aide de la section « Apparence ».



**Apprenez Plus sur  
Diagrammes de tableaux  
de bord**

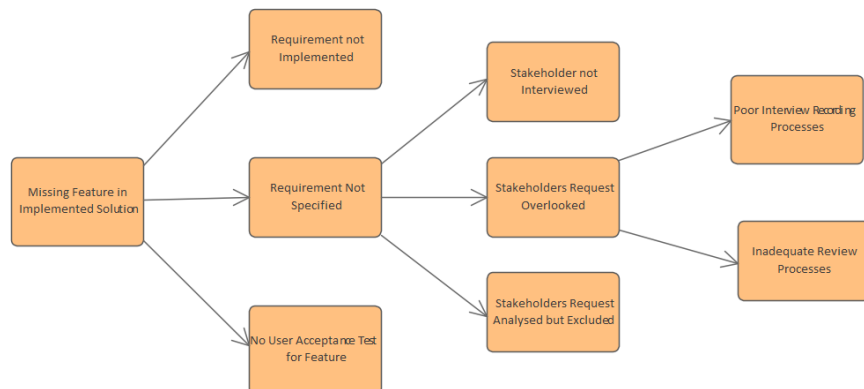
- [Standard Charts](#)

# Diagramme d'arbre de Décision

## Connaître l' Diagramme Décision

### Présentation du Diagramme d'arbre Décision

Les arbres Décision constituent un moyen efficace de représenter graphiquement un certain nombre d'options et fournissent un mécanisme permettant d'étudier les résultats et les avantages possibles du choix de ces options. Ils peuvent également aider l'analyste à se faire une idée équilibrée des risques et des avantages associés à chaque plan d'action possible. Ils sont un proche cousin du Tableau de Décision mais ont l'avantage d'être graphiques. Enterprise Architect dispose d'un diagramme spécialement conçu permettant de modéliser et d'afficher des décisions complexes, y compris les probabilités et l'incertitude.



### Où trouver l' Diagramme Décision

Ruban : Conception > Diagramme > Ajouter Diagramme > Modélisation Stratégique > Arbre Décision

Navigateur Barre d'outils : Icône Nouveau Diagramme > Modélisation Stratégique > Arbre Décision

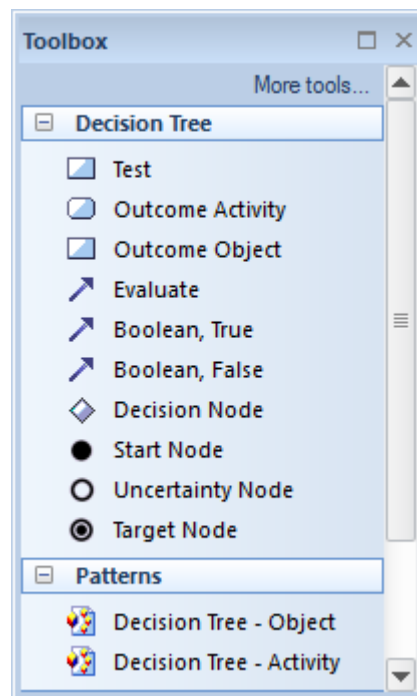
Menu contextuel de la fenêtre Navigateur | Ajouter Diagramme ... > Modélisation Stratégique > Arbre Décision

### Utilisation du Diagramme d'arbre Décision

Les arbres Décision peuvent être utilisés pour aider aux processus de prise de décision, en particulier lorsque la décision implique un ensemble complexe de conditions qui ont des probabilités d'occurrence différentes. Ils peuvent être utilisés pour l'analyse de décision stratégique ou opérationnelle et peuvent aider à formaliser la base de la prise de décision, en particulier lorsqu'il est impératif que les mesures prises soient basées sur une analyse formelle ou qu'elles aient des conséquences coûteuses. Un arbre Décision peut être utilisé pour présenter une image graphique d'un Tableau de Décision pour les parties prenantes qui préfèrent visualiser diagrammes plutôt que tableaux et des documents.

### Options pour le Diagramme d'arbre Décision

Les arbres Décision peuvent être dessinés avec différents niveaux de formalité, depuis des arbres simples avec une série de décisions aboutissant à des résultats jusqu'à des arbres plus formels impliquant une incertitude avec des valeurs de probabilité attribuées ou des expressions formelles avec des paramètres d'entrée. La page de la boîte à outils « Arbre Décision » contient une gamme d'éléments qui peuvent être utilisés et deux Motifs qui peuvent être utilisés pour créer un diagramme donnant à l'analyste un point de départ.



Filtres de Diagramme peuvent également être utilisés lors de la présentation des diagrammes pour attirer l'attention sur certaines parties des diagrammes et les diagrammes peuvent être présentés comme dessinés à la main ou dans un style tableau blanc en modifiant les propriétés du diagramme .

**Apprenez Plus sur le  
Diagramme d'Arbre  
Décision**

[Decision Tree](#)



# Diagramme Statemachine

## Apprendre à connaître le Diagramme Statemachine

### Présentation du Diagramme Statemachine

Le diagramme Statemachine est l'un des diagrammes de comportement du Systems Modeling Language (SysML) qui peut être utilisé pour modéliser une large gamme d'objets. Il s'agit d'un diagramme à usage général pour modélisation d'entités dans les domaines commerciaux et techniques, y compris les termes et concepts, Règles Métier et les capacités dans les schémas XML et de base de données.

### Où trouver le Diagramme Statemachine

Ruban : Conception > Diagramme > Ajouter Diagramme : Type = Ingénierie des Systèmes > SysML, Sélectionner depuis = SysML 1.5, Types Diagramme = Statemachine > <type requis>

Barre d'outils de la fenêtre Navigateur : Type = Ingénierie des Systèmes > SysML, Select From = SysML 1.5, Diagramme Types = Statemachine > <type requis>

Fenêtre Navigateur Menu Contexte | Ajouter Diagramme : Type = Ingénierie des Systèmes > SysML, Select From = SysML 1.5, Types Diagramme = Statemachine > <type requis>

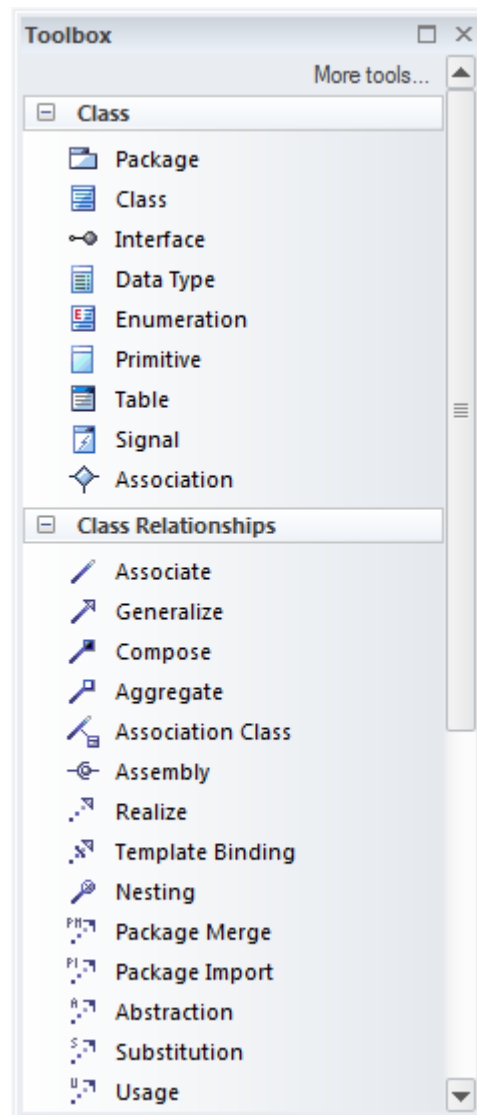
### Utilisation du Diagramme Statemachine

Le diagramme de classes peut être utilisé chaque fois qu'une représentation logique ou structurelle d'un système est requise. Il est applicable à modélisation de concepts commerciaux et techniques et peut être utilisé pour modéliser des informations et des structures telles que des schémas XML et de base de données.

### Options pour le Diagramme Statemachine

Le diagramme de classe (comme tout diagramme ) peut être considéré comme une liste d'éléments, ce qui facilite le travail avec les propriétés de l'élément.

Filtres de Diagramme peuvent également être utilisés lors de la présentation des diagrammes pour attirer l'attention sur certaines parties des diagrammes et les diagrammes peuvent être présentés comme dessinés à la main ou dans un style tableau blanc en modifiant les propriétés du diagramme .



**Apprenez Plus sur le  
Diagramme Statemachine**

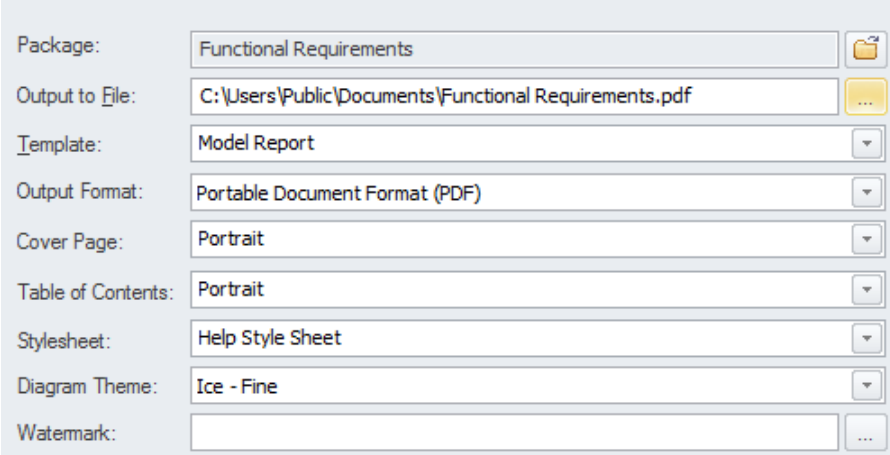
[StateMachines](#)

# Documentation

## Découvrir la documentation

### Présentation de la documentation

Les fonctionnalités de documentation permettent de générer automatiquement une large gamme de documentation directement à partir des modèles. Celles-ci peuvent être basées sur des documents tels que des formats PDF et Docx ou HTML. gabarits flexibles peuvent être utilisés pour personnaliser complètement les documents générés, y compris les logos d'entreprise, tableaux de contenu, tableaux d'informations sur les éléments et diagrammes . Des rapports ad hoc peuvent également être créés à partir d'un certain nombre d'outils tels que le glossaire et la fenêtre de recherche.



The screenshot shows a settings dialog for generating documentation. It includes the following fields:

- Package: Functional Requirements
- Output to File: C:\Users\Public\Documents\Functional Requirements.pdf
- Template: Model Report
- Output Format: Portable Document Format (PDF)
- Cover Page: Portrait
- Table of Contents: Portrait
- Stylesheet: Help Style Sheet
- Diagram Theme: Ice - Fine
- Watermark: (empty)

### Où trouver la documentation

Ruban : Publier > Rapports de Modèle > Générateur de Rapports

### Utilisation de la documentation

Les modélisateurs, analystes, architectes, Gestionnaires de Projet et autres peuvent utiliser cette facilité pour produire une large gamme de publications et de rapports basés sur des documents, tels qu'une Spécification Exigences du système, Rapport de cas d'utilisation, Dictionnaire de données, une description Architecture de la solution, etc. Il peut également être utilisé pour créer des rapports ad hoc, par exemple une liste des exigences les plus volatiles. La documentation HTML peut également être publiée pour permettre aux parties prenantes qui n'ont pas accès à Enterprise Architect de visualiser les modèles à partir d'un site intranet qui peut simplement être placé sur un système de fichiers sans avoir besoin d'un serveur Web.

### Options de documentation

Plusieurs options peuvent être définies pour personnaliser les informations incluses dans un document généré, notamment l'ordre des éléments et diagrammes et le masquage de certains éléments. Des filtres, des substitutions de mots et d'autres options peuvent également être appliqués.

The screenshot displays the 'Order' and 'Filter' sections of a software interface. The 'Order' section is divided into three rows: 'Packages by:', 'Elements by:', and 'Diagrams by:'. Each row contains two dropdown menus, both set to 'Tree Order' and 'Ascending'. Below this is an 'Options' section with a grid of checkboxes. The 'Hide Diagram Borders' checkbox is checked, while others are unchecked. At the bottom of the 'Options' section are 'Diagram Format:' (set to 'Metafile') and 'Adjust Heading Levels:' (set to 'Heading 9'). A 'Switch generator' button is located at the very bottom. The 'Filter' section on the right includes 'Only include objects:' (Created After 15/05/2018), 'Where Package Phase:' (set to '>' and '3'), 'With element status:' (empty), 'Connector Direction:' (set to 'Both'), and 'Except where Query excludes' (radio buttons for 'Custom SQL' and 'Custom Script').

**Apprenez Plus sur la  
Documentation**

[Model Publishing](#)

# Matrice d'Analyse des Écarts

## Connaître la Matrice d'Analyse des Écarts

### Présentation de la Matrice d'Analyse des Écarts

La Matrice d'Analyse des Écarts est une Matrice de relations spécialisée qui permet d'enregistrer les écarts qui existent entre deux versions d'une partie d'une entreprise. Les écarts entre deux versions différentes d'une architecture peuvent être enregistrés, ou les écarts entre deux versions de Capacités ou de Compétences du Personnel, ou deux versions d'Informations ou de Données. L'outil est structuré de manière similaire à une feuille de calcul avec des colonnes et des lignes. Les éléments qui constituent la ligne de base (point de départ) sont répertoriés sous forme de lignes et les éléments qui constituent la cible (point final) sont répertoriés sous forme de colonnes. Il existe une colonne pour enregistrer les éléments manquants ou éliminés et une ligne pour enregistrer les nouveaux éléments. À l'intersection d'un élément de base et d'un élément cible, notes peuvent être ajoutées pour décrire les détails de la relation entre les deux éléments.

Target \ Baseline	Video Conferencing Services	Enhanced Telephony Services	Mailing List Services	Missing / Eliminated
Broadcast Services				Retired service : Intentionally eliminated
Video Conferencing Services	Included			
Enhanced Telephony Services		Potential match		
Shared Screen Services				Address Shared Screen Service : Unintentionally eliminated
New		Improve Telephony service : To be enhanced	Mailing List : New-To be produced or developed	

### Où trouver la Matrice d'Analyse des Écarts

Ruban : Conception > Paquetage > Analyse des écarts

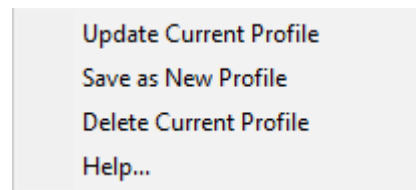
### Utilisation de la Matrice d'Analyse des Écarts

La Matrice d'Analyse des Écarts peut être utilisée à la fois pour l'analyse commerciale et technique. C'est un outil à usage général pour enregistrer les détails d'une comparaison entre différentes versions d'une partie d'une entreprise. Dans l'analyse commerciale, elle peut être utilisée pour analyser les compétences

du personnel, les données et les informations, Processus Métier , les lignes de Métier et plus encore, en les comparant aux états actuels et futurs d'une entreprise. Dans Architecture d'Entreprise la matrice peut être utilisée pour enregistrer les écarts entre les architectures de base, de transition et cibles, en comparant les capacités, Architecture et les composants de la solution, etc.

### Options pour la Matrice d'Analyse des Écarts

La Matrice d'Analyse des Écarts peut être configurée pour afficher différentes parties du référentiel. Une fois que les Paquetages appropriés ont été choisis pour la Cible et la Ligne de base, et que les types d'éléments ont été sélectionnés pour le filtre, le type d'élément Écart peut être sélectionné. L'élément choisi pour l'écart limitera les éléments disponibles pour représenter l'écart pour les éléments « Manquants » ou « Nouveaux » dans les cellules de la matrice. Un certain nombre de choix sont disponibles dans le menu « Options », notamment la possibilité de mettre à jour, de supprimer et d'enregistrer la Matrice d'Analyse des Écarts en tant que profil, en lui donnant un nom afin qu'il puisse être rappelé ultérieurement.



### Apprenez Plus sur la Matrice d'Analyse des Écarts

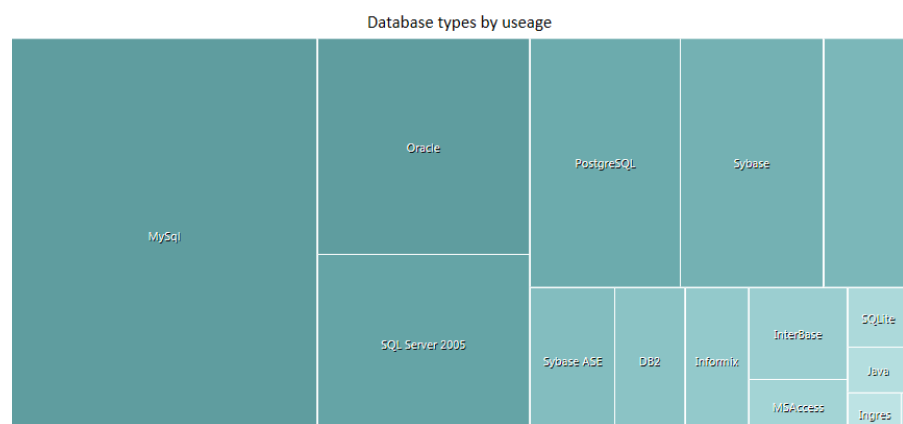
[Gap Analysis Matrix](#)

# Carte de Chaleur

## Découvrir la Carte de Chaleur

### Présentation de la Carte de Chaleur

Une Carte de Chaleur est un type de graphique qui peut être utilisé pour visualiser des données en deux dimensions. Elle utilise la couleur des rectangles pour indiquer une dimension des données et la taille relative des rectangles pour indiquer une autre dimension. Elles sont généralement utilisées pour créer des représentations convaincantes des données pour la prise de décisions stratégiques ou tactiques. Elles peuvent être utilisées à n'importe quel niveau d'un référentiel, de architecture stratégique aux architectures technologiques.



### Où trouver la Carte de Chaleur

Double-cliquez sur l'élément Graphique | Détails du graphique | Source > Paquetage

### Utilisation de la Carte de Chaleur

Cartes des Chaleur sont généralement utilisées pour créer des représentations convaincantes de données pour la prise de décisions stratégiques ou tactiques. Elles peuvent être utilisées avec Exigences pour indiquer les statuts d'un groupe d'exigences et, si les mesures sont disponibles, le coût de mise en œuvre estimé de chaque exigence. Elles peuvent être utilisées avec un inventaire d'applications ou de technologies pour montrer la prévalence des technologies. Par exemple, quelles applications ont été développées dans un langage particulier ou exécuter sur un système d'exploitation particulier.

### Options pour la Carte de Chaleur

En alternative à la spécification des paramètres de la Carte de Chaleur dans les champs de l'onglet ' Paquetage ', vous pouvez sélectionner l'onglet ' SQL personnalisé ' et créer une Carte de Chaleur personnalisée à l'aide de SQL. Vous spécifiez toujours le type de graphique dans le champ ' Type ', mais les autres champs de dialogue sont grisés.

### Apprenez Plus sur la Carte de Chaleur

[Heat Maps](#)

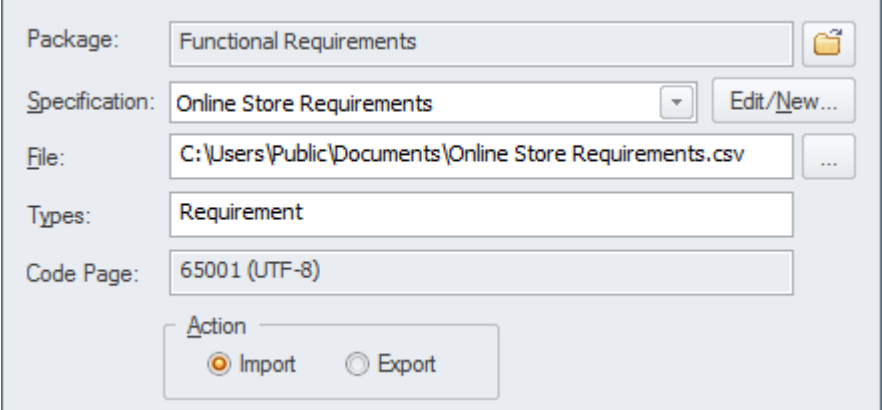
# Importer et exporter des feuilles de calcul

## Importer et exporter des feuilles de calcul

### Présentation des feuilles de calcul d'importation et d'exportation

Cette fonctionnalité est utile pour importer des exigences définies dans une feuille de calcul ou un tableau de traitement de texte dans Enterprise Architect. Une fois dans Enterprise Architect, les exigences peuvent être gérées et tracées vers des éléments tels que les facteurs opérationnels, les scénarios et les composants. Les exigences dans Enterprise Architect peuvent également être exportées vers une feuille de calcul dans le but de les fournir à un tiers ou pour un certain type d'analyse numérique ou statistique. Le mappage entre les champs de la feuille de calcul et les propriétés analogues dans Enterprise Architect est entièrement configurable à l'aide d'une spécification.

Pour un échange d'informations plus détaillé, le lien MDG pour Microsoft Office (disponible auprès de Sparx Systems) fournit des fonctionnalités supplémentaires et des points d'intégration utiles pour gérer des exigences complexes.



The screenshot shows a dialog box for CSV import/export. It includes the following fields and controls:

- Package:** Functional Requirements
- Specification:** Online Store Requirements (with an Edit/New... button)
- File:** C:\Users\Public\Documents\Online Store Requirements.csv (with a browse button)
- Types:** Requirement
- Code Page:** 65001 (UTF-8)
- Action:** Radio buttons for Import (selected) and Export.

### Où trouver des feuilles de calcul d'importation et d'exportation

Ruban : Publier > Échange de Modèles > CSV

### Utilisation des feuilles de calcul d'importation et d'exportation

Cette fonctionnalité peut être utilisée pour importer ou exporter des exigences à partir d'un fichier CSV. Avant l'installation d'un outil tel que Enterprise Architect, les analystes utilisaient peut-être une feuille de calcul ou un tableau dans leur traitement de texte préféré pour enregistrer des exigences ; celles-ci peuvent être facilement importées à l'aide de la fonctionnalité d'importation CSV. Alternativement, des exigences doivent parfois être fournies à un tiers qui spécifiera généralement qu'il les souhaite dans un fichier de feuille de calcul ; cela peut être réalisé à l'aide de la fonctionnalité d'exportation.

### Options d'importation et d'exportation de feuilles de calcul

La fonctionnalité d'importation et d'exportation est entièrement configurable et dispose d'une spécification définie par l'utilisateur pour faciliter le mappage des champs de la feuille de calcul aux propriétés des exigences dans Enterprise Architect. Cette fonctionnalité inclut également la possibilité d'importer et d'exporter des champs dans des valeurs étiquetées de l'exigence.

### Apprenez Plus sur l'importation et l'exportation de feuilles de calcul

[CSV Import and Export](#)



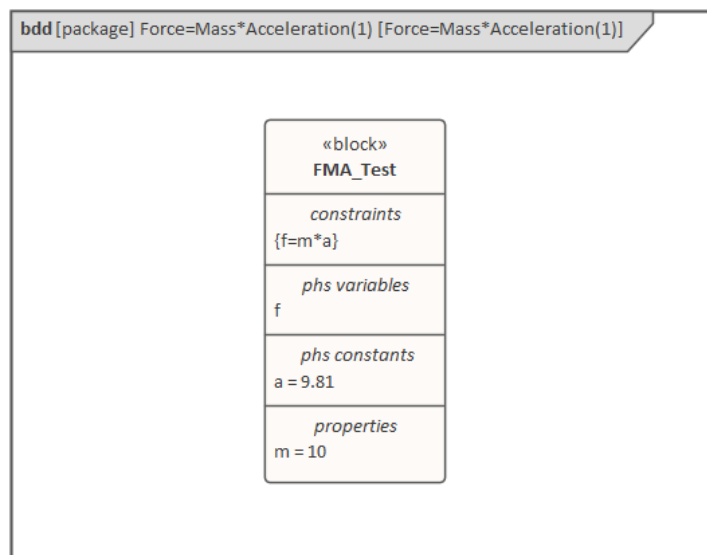


# Diagramme Paramétriques

## Connaître le Diagramme Paramétriques

### Présentation du Diagramme Paramétriques

Le diagramme Paramétriques est l'un des diagrammes structurels du Systems Modeling Language (SysML) qui peut être utilisé pour modéliser des systèmes d'équations mathématiques. Il existe un assistant qui vous aidera à convertir les équations en éléments modélisés, y compris les propriétés. Une fois les équations modélisées, elles peuvent être utilisées pour créer des simulations à l'aide de l'intégration OpenModelica.



### Où trouver le Diagramme Paramétriques

Ruban : Conception > Diagramme > Ajouter Diagramme : Type = Ingénierie des Systèmes > SysML, Select From = SysML 1.5, Diagramme Types = Paramétriques

Barre d'outils de la fenêtre Navigateur : Type = Ingénierie des Systèmes > SysML, Select From = SysML 1.5, Diagramme Types = Paramétriques

Fenêtre Navigateur Menu Contexte | Ajouter Diagramme : Type = Ingénierie des Systèmes > SysML, Select From = SysML 1.5, Diagramme Types = Paramétriques

### Utilisation du Diagramme Paramétriques

Le diagramme Paramétriques peut être utilisé chaque fois que vous avez besoin de visualiser ou de simuler un système d'équations.

### Options pour le Diagramme Paramétriques

Le diagramme Paramétriques offre un certain nombre d'options dont la modélisation d'une équation simple ou d'un système d'équations.

### Apprenez Plus sur le Diagramme Paramétriques

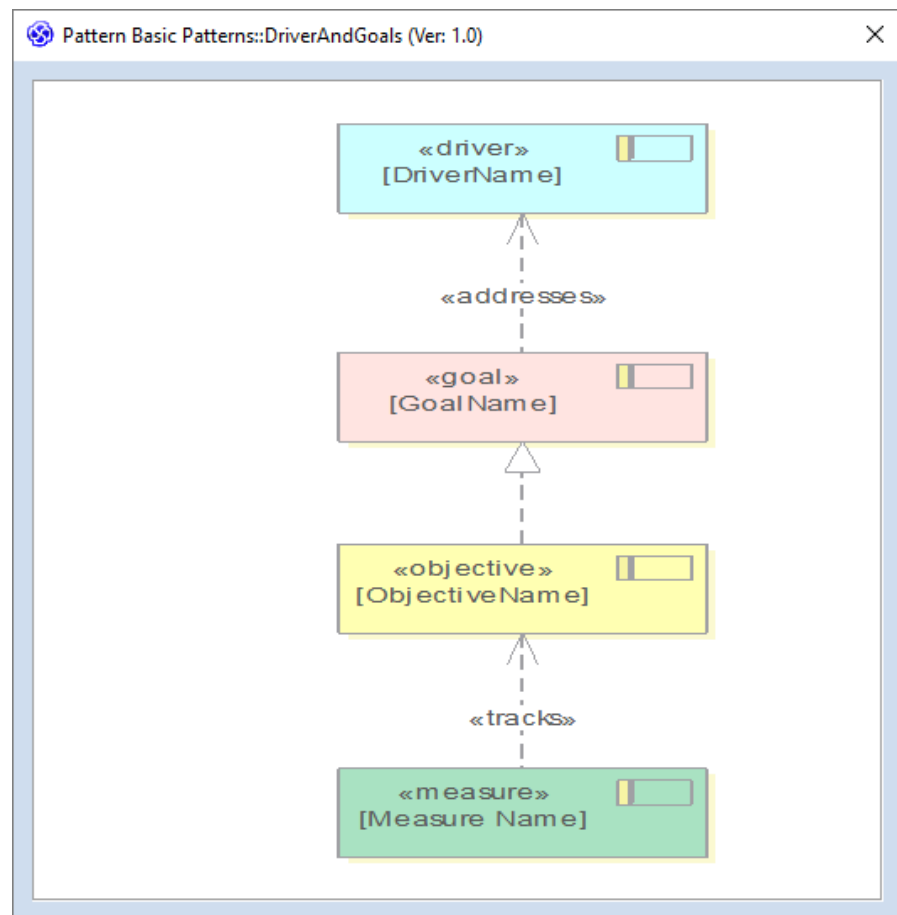
- [Parametric Diagrams](#)
- [Class Diagram](#)

# Motifs

## Découvrir Motifs

### Présentation Motifs

Un Motif est une solution de conception réutilisable générale à un problème courant dans un contexte architectural donné. Motifs ne sont pas des conceptions résolues, mais plutôt gabarits de résolution d'un problème. Le concept est né dans le monde architecture du bâtiment et a été publié pour la première fois dans un livre de Christopher Alexander intitulé *Design Motifs*. Ils ont ensuite été appliqués à l'industrie du logiciel et ont été largement utilisés par le domaine de l'ingénierie logicielle pour résoudre des problèmes d'ingénierie logicielle récurrents, même si à première vue la nature des problèmes semblait assez différente.



### Où trouver Motifs

Créer un Motif :

Choisissez l'option du ruban « Spécialiser > Technologies > Publier la technologie > Publier Diagramme en tant que Motif »

Utiliser un Motif :

Dans la fenêtre Navigateur sélectionnez "Ressources > Motifs > < groupe motif > > Cliquez-droit sur le nom Motif > Ajouter Motif au Diagramme "

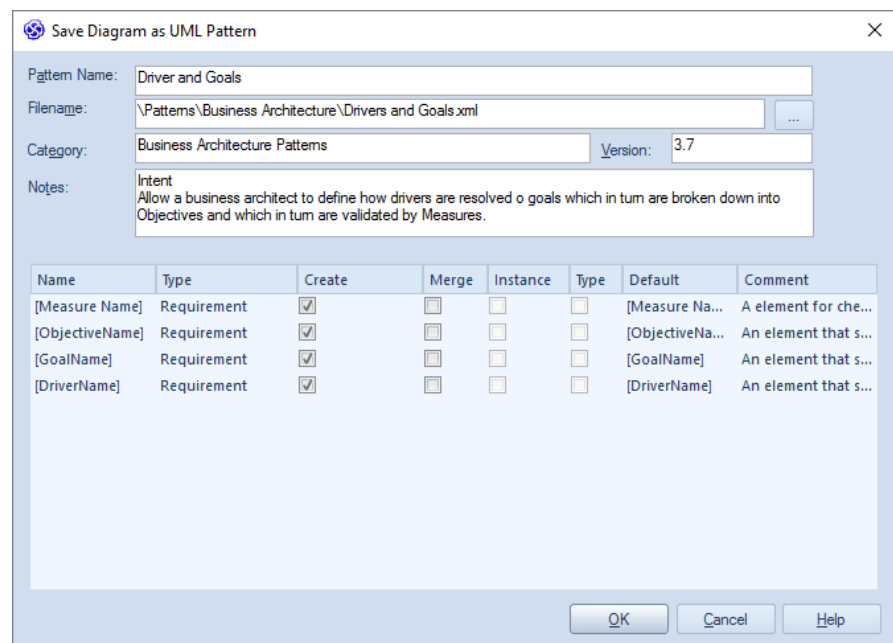
### Utilisation des Motifs

Motifs peuvent être appliqués dans une large gamme de situations, de l'architecture commerciale à architecture technologique, mais sont toujours utilisés pour appliquer une solution commune à un certain nombre de problèmes ou de

contextes qui, à première vue, peuvent sembler très différents. Enterprise Architect a fourni un mécanisme utile pour l'extraction Motifs, ce qui signifie que tout diagramme peut être publié en tant que Motif, puis réutilisé dans le même contexte ou dans un contexte différent. Un exemple de Motif et de son utilisation peut être un Motif articulant la relation entre les pilotes, les buts, les objectifs et les mesures. Un diagramme existant peut être publié en tant que Motif, puis toute architecture commerciale peut réutiliser le Motif en le faisant simplement glisser sur un diagramme vide.

**Options pour Motifs**

Motifs sont généralement disponibles dans l'onglet « Ressources » de la fenêtre Navigateur, mais ils sont également parfois intégrés aux technologies et mis à disposition à partir d'une page de la boîte à outils. Plusieurs options sont disponibles lors de la publication d'un Motif, notamment la possibilité de décrire les détails du Motif dans son ensemble et d'inclure notes pour chacun des éléments qui Motif composent.



# Matrice de relations

## Apprendre à connaître la Matrice relationnelle

### Présentation de la Matrice relationnelle

La Matrice de relations fournit une vue de type matrice convaincante et visuelle pour une analyse pratique de la manière dont Exigences sont liées entre elles et aux autres éléments du modèle. Elle peut être utilisée pour visualiser les relations entre les parties prenantes et leurs Exigences, la manière dont les cas d'utilisation sont liés aux Exigences Métier ou Exigences fonctionnelles, la manière dont les capacités sont liées aux pilotes Métier, les composants qui implémentent un ensemble d' Exigences, et bien plus encore. Un nombre quelconque de matrices peut être défini rapidement, puis enregistré pour être visualisé dans des ateliers, ou inclus dans la documentation générée automatiquement à partir du modèle ou exporté vers un fichier tableur. Lorsqu'une matrice est créée, les connexions peuvent être visualisées en plaçant les Exigences sur un axe de la matrice et les éléments connectés sur l'autre axe, puis les cellules de la matrice indiqueront la direction de la relation.

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
+ Source												
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

### Où trouver la Matrice de relations

Dans la fenêtre Navigateur, cliquez sur un Paquetage et sélectionnez :

- L'onglet 'Ressources' | Profils Matrice | Cliquez-droit sur un profil | Ouvrez le profil Matrice ou
- Le ruban Démarrer > Toutes Windows > Conception > Outils > Matrice Paquetage

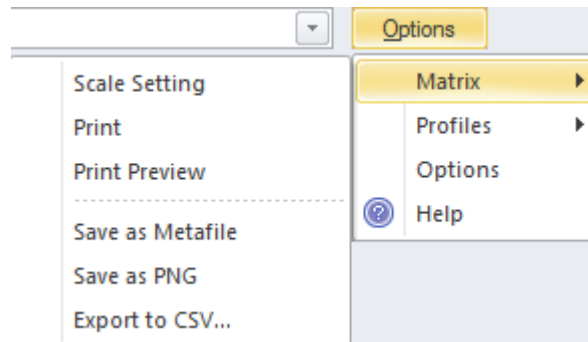
### Utilisation de la Matrice de relations

Pour afficher les relations qui existent entre les éléments (par exemple, quelles Exigences sont réalisées par quels cas d'utilisation) dans deux Paquetages dans une matrice visuellement attrayante. Cela est utile pour analyser les éléments ou les relations manquants ; par exemple, pour déterminer quelles Exigences ne sont réalisées par aucun cas d'utilisation, ou quels composants n'ont pas Exigences ou de cas d'utilisation correspondants. Cela est particulièrement utile dans les ateliers avec les parties prenantes Métier qui ne sont peut-être pas habituées à voir

Exigences dans diagrammes de trace.

### Options pour la Matrice de relations

Il existe une gamme d'options qui peuvent être définies pour la Matrice de relations, notamment l'enregistrement dans l'onglet « Ressources » de la fenêtre Navigateur ou dans un format CSV pour l'ouvrir dans une feuille de calcul. L'apparence de la Matrice de relations peut également être modifiée en triant les éléments, en affichant une vue de numérotation schématique et en supprimant les noms Paquetage . Ces éléments sont disponibles à partir du bouton Options de la Matrice de relations.



Apprenez Plus sur la Matrice Relationnelle

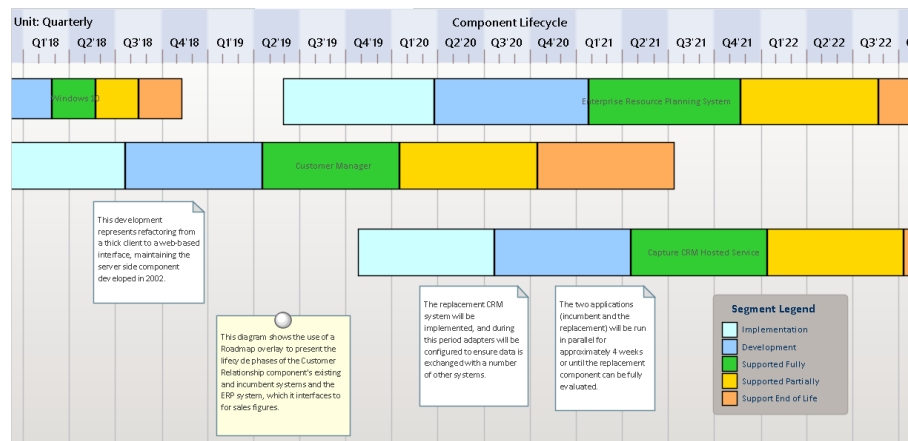
[Relationship Matrix](#)

# Feuille de Route Diagramme

## Connaître le Diagramme Feuille de Route

### Présentation du Diagramme Feuille de Route

Le diagramme Feuille de Route est une superposition qui peut être appliquée à n'importe quel diagramme pour décrire les phases significatives des éléments et la façon dont ils évoluent au fil du temps.



Il n'y a aucune restriction sur le type d'élément qui peut apparaître sur le diagramme, et tout diagramme peut avoir une superposition Feuille de Route définie. Les phases significatives définies par l'utilisateur dans la vie de l'élément sont représentées par des barres de couleur, qui peuvent être définies pour afficher la durée. Les couleurs et les phases peuvent être configurées à l'aide d'une légende Diagramme, qui les applique automatiquement aux éléments du diagramme. Elles sont particulièrement utiles dans diagrammes Architecture d'Entreprise pour décrire les capacités et Feuilles de Route d'application.

### Où trouver la Feuille de Route Diagramme

Ruban : Disposition > Diagramme > Feuille de Route

Diagramme Menu Contexte : Feuille de Route

### Utilisation du Diagramme Feuille de Route

Le diagramme Feuille de Route a une large gamme d'utilisations en Architecture d'Entreprise où il peut être utilisé pour montrer les feuilles de route des applications et des capacités en Ingénierie des Systèmes, où il est utilisé pour montrer le timing dans les composants de bas niveau.

### Options pour le Diagramme Feuille de Route

La superposition Feuille de Route propose une gamme d'options qui déterminent les propriétés de la chronologie, telles que l'échelle des règles de temps, les unités, leurs positions et l'apparence de la chronologie, y compris les polices et les couleurs. La hauteur et la position de la chronologie peuvent également être configurées pour s'adapter au diagramme et à l'affichage.

**Roadmap options**

Enabled

Timeline Properties

Roadmap Title: Product Lifecycle

Units: Quarterly

Tick spacing: [Slider]

Timeline Start: 10/10/2020

Timeline End: 04/11/2020

Scale: [Color swatch] Marker offset: [Color swatch] Ticks: [Color swatch]

Timeline Appearance

Roadmap Position: Top

Timeline Height: [Slider]

Timeline Color: [Color swatch]

Line Color: [Color swatch]  Lines at major intervals

Font: **A**  Center marker labels

Use legend for phase colors

OK Cancel Help

La légende Diagramme peut être configurée pour définir les phases de la vie de l'élément, pour définir la spécification des bandes colorées et bien plus encore. Les segments Feuille de Route peuvent être affichés ou masqués sur des éléments individuels dans les cas où un segment particulier pourrait ne pas s'appliquer à un ou plusieurs éléments du diagramme .

**Apprenez Plus sur le  
Diagramme Feuille de  
Route**

[Roadmap Diagrams](#)



# Gestionnaire de Spécification

## Faire connaissance avec le Gestionnaire de Spécification

### Présentation du Gestionnaire de Spécification

Le Gestionnaire de Spécification est l'outil central pour travailler avec Exigences . Il fournit une interface ressemblant à un traitement de texte ou à un tableur pour saisir, gérer et visualiser Exigences . De nouvelles Exigences peuvent être créées avec des noms et des descriptions détaillées et des propriétés telles que le statut et la priorité peuvent être ajoutées à partir de listes déroulantes. Exigences existantes peuvent être visualisées et gérées dans une vue pratique, et leur modification dans le Gestionnaire de Spécification les modifiera dans tous les autres endroits du référentiel tels que diagrammes et les fenêtres. C'est l'outil parfait pour les analystes plus à l'aise avec le texte plutôt qu'avec diagrammes et qui sont habitués à travailler dans un traitement de texte ou un tableur. Il présente l'avantage supplémentaire que les exigences font partie d'un modèle et peuvent être reliées à d'autres éléments, notamment les moteurs Métier , les parties prenantes et les composants de solution.

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.

### Où trouver le Gestionnaire de Spécification

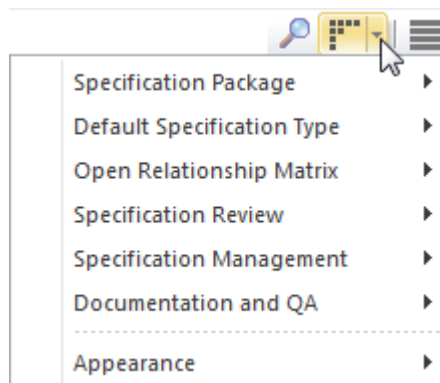
Fenêtre Navigateur | Cliquez-droit sur Paquetage | Gestionnaire de Spécification

### Utilisation du Gestionnaire de Spécification

Pour créer, visualiser et gérer Exigences dans un outil basé sur du texte qui ressemble à un traitement de texte ou à une feuille de calcul. Des détails peuvent être ajoutés aux Exigences et les propriétés des exigences peuvent être ajoutées à partir de listes déroulantes. Lorsque les Exigences sont modifiées dans le Gestionnaire de Spécification les modifications sont facilement reflétées dans la fenêtre Navigateur et dans toutes les autres fenêtres.

**Options pour le  
Gestionnaire de  
Spécification**

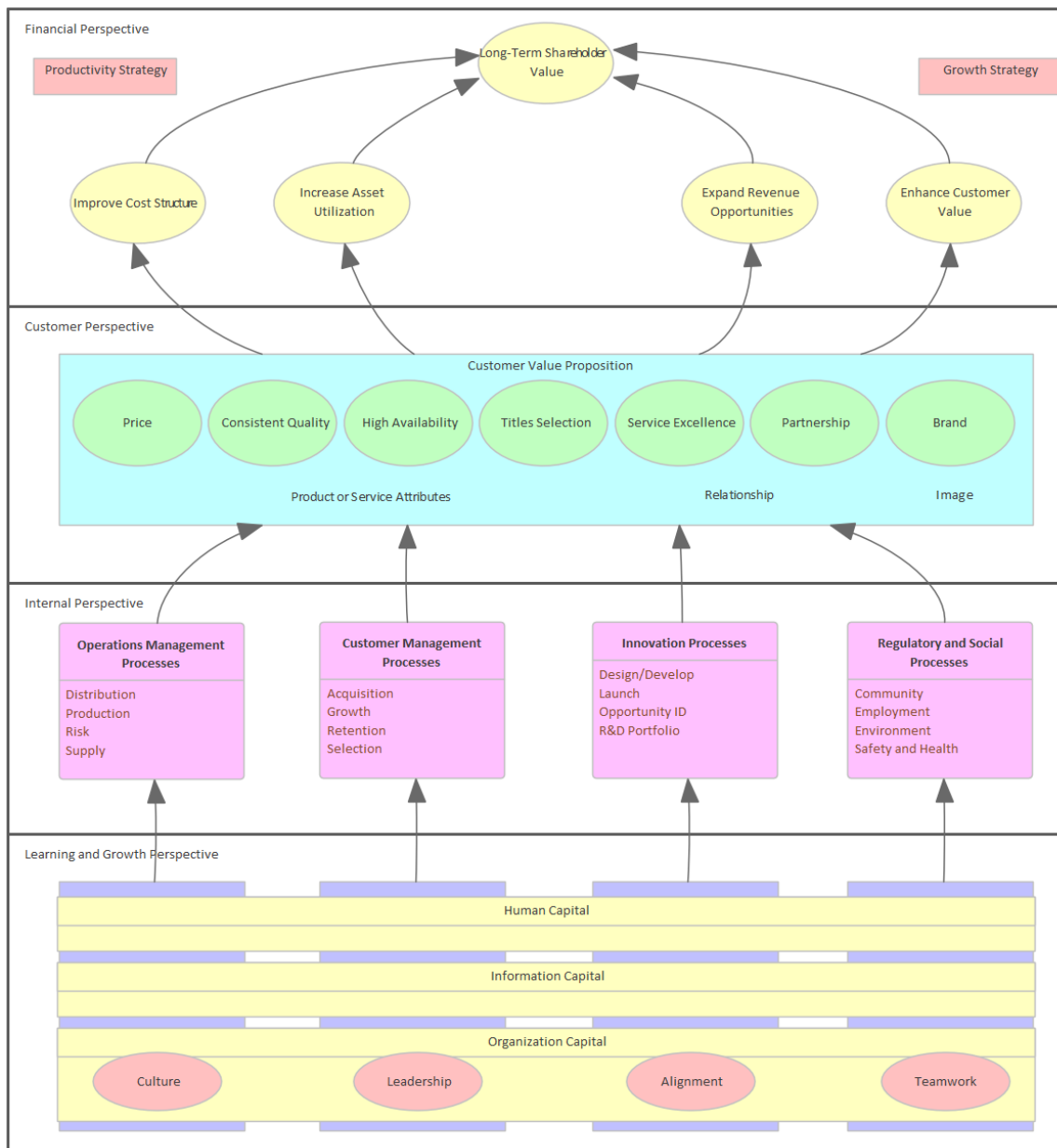
Il existe une large gamme d'options disponibles dans le menu Options, pour personnaliser la façon dont vous utilisez le Gestionnaire de Spécification . Celles-ci incluent la numérotation hiérarchique des niveaux, la dénomination automatique, la vérification orthographique, la documentation, l'importation et l'exportation des Exigences , l'accès à divers outils connexes et bien plus encore.

**Apprenez Plus sur le  
Gestionnaire de  
Spécification**

[The Specification Manager](#)

# Carte de Stratégie

Une Carte de Stratégie est un diagramme utilisé pour décrire les principaux objectifs stratégiques importants pour une organisation ou une équipe commerciale. Le diagramme montre quatre perspectives importantes qui sont les questions significatives qui fournissent la définition d'une stratégie. Les perspectives définies sont : « Financier », « Client », « Processus Métier internes » et « Apprentissage et croissance ». Le diagramme est utilisé comme outil de communication pour garantir une compréhension commune de la stratégie, pour concentrer les efforts de l'organisation et pour aider à l'évaluation des progrès.



## Connaître la Carte de Stratégie

### Où trouver la Carte de Stratégie

Ruban : Conception > Diagramme > Ajouter Diagramme > Modélisation Stratégique > Carte de Stratégie

Navigateur Barre d'outils : Icône Nouveau Diagramme > Modélisation Stratégique

> Carte de Stratégie

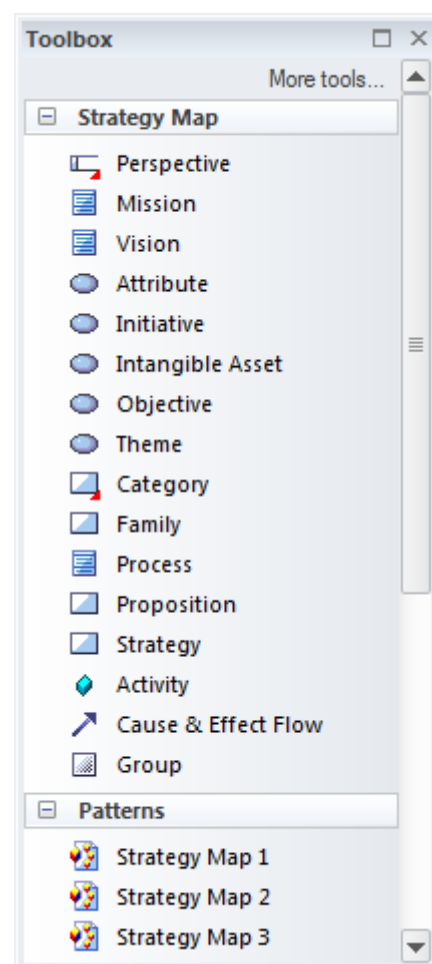
Menu contextuel de la fenêtre Navigateur | Add Diagramme ... > Modélisation Stratégique > Carte de Stratégie

### Utilisation de la Carte de Stratégie

La Carte de Stratégie est utilisée pour modéliser les principaux objectifs stratégiques qu'une organisation ou une équipe de direction souhaite atteindre. Les éléments de chacune des quatre perspectives peuvent être liés à d'autres éléments du référentiel pour montrer comment ils pourraient être mis en œuvre au niveau de l'entreprise, de l'application ou de la technologie.

### Options pour la Carte de Stratégie

Une Carte de Stratégie peut être créée à l'aide de Motifs qui créent automatiquement des éléments et un diagramme pouvant servir de point de départ pour la Carte de Stratégie. Trois Motifs sont disponibles, allant d'une expression très simple avec un seul élément dans chaque perspective à une expression complètement élaborée avec plusieurs éléments dans chaque perspective. Une boîte à outils fournit une gamme d'éléments et de relations supplémentaires pour étendre les cartes de base créées à l'aide des Motifs.



Apprenez Plus sur la Carte de Stratégie

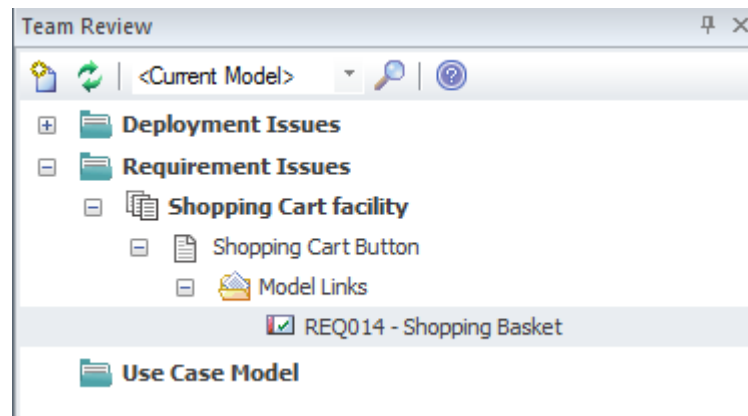
[Strategy Maps](#)

# Bibliothèque

## Découvrir la Bibliothèque

### Présentation de la Bibliothèque

La fenêtre Bibliothèque offre aux développeurs, modélisateurs, clients et parties prenantes la possibilité de commenter et de fournir des commentaires sur les travaux en cours ou à l'issue d'une étape ou d'un projet.



### Utilisation de la Bibliothèque

La fonctionnalité Bibliothèque peut être utilisée pour effectuer révisions de modèles à partir de plusieurs perspectives , y compris des visites guidées, révisions de modèles formelles ou révisions ad hoc.

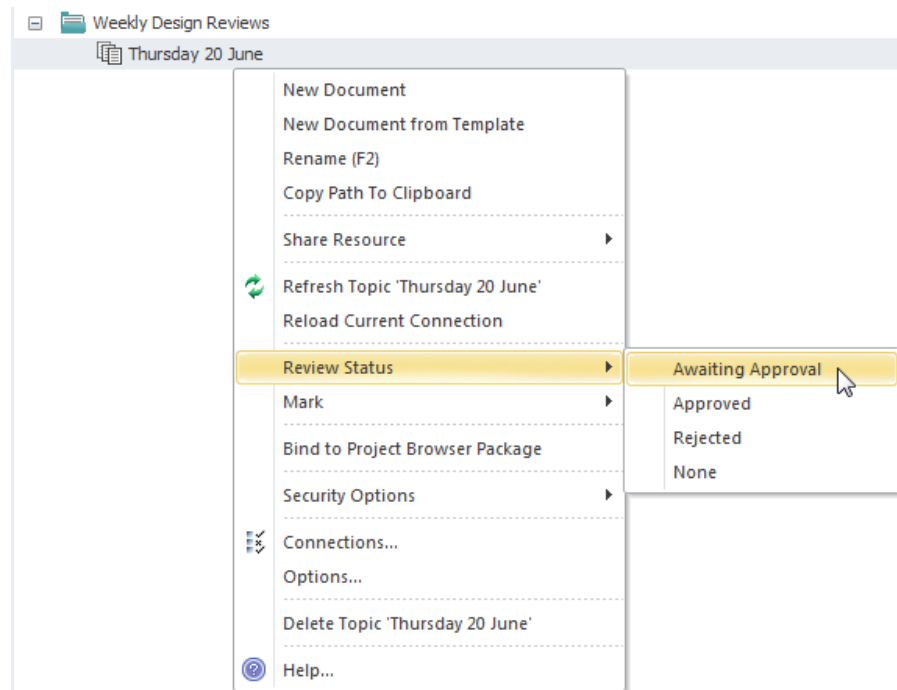
### Où trouver la Bibliothèque

Pour publier ou afficher la discussion d'un élément

Ruban : Démarrer > Collaborer > Modèle Bibliothèque

### Options pour La Bibliothèque

Il existe une large gamme de paramètres disponibles pour configurer la Bibliothèque , disponibles dans les menus contextuels Catégorie et Sujet, et comprenant le réglage du statut de la catégorie ou du sujet et d'autres options. Diagrammes , les éléments et fonctionnalités des éléments peuvent être facilement glissés depuis la fenêtre Navigateur pour créer des liens de modèle qui peuvent être utilisés par les membres de l'équipe pour créer des liens hypertexte directement depuis la fenêtre Bibliothèque vers ces éléments dans la fenêtre Navigateur .



**Apprenez Plus sur  
Bibliothèque d'Équipe**

[The Model Library](#)

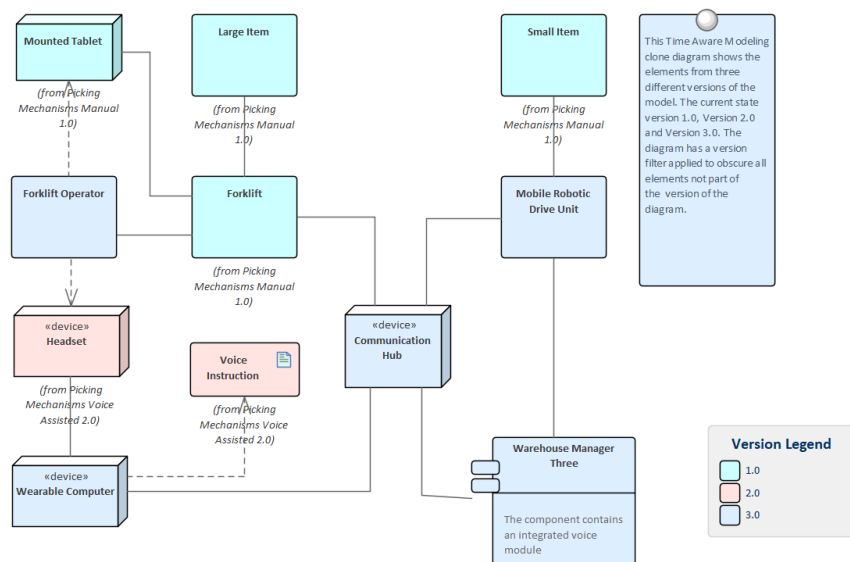
# Modélisation Consciente du Temps

## Connaître Modélisation Consciente du Temps

### Présentation de Modélisation Consciente du Temps

La fonctionnalité Modélisation Consciente du Temps permet aux analystes, architectes et autres de créer des versions incrémentales de leurs modèles en fournissant des outils qui facilitent la migration des éléments, diagrammes et Paquetages à travers la dimension du temps, permettant de créer plusieurs transitions ou versions. Les modèles de base (« tels quels », état actuel) restent inchangés et un nombre quelconque de modèles cibles (« à venir », état futur) peuvent être construits à des fins de visualisation et d'analyse. Elle est particulièrement utile lorsqu'un certain nombre d'options futures doivent être représentées et comparées efficacement, ce qui permet de mener une analyse « et si ». Le temps est l'une des dimensions les plus importantes de l'architecture car il constitue le substrat sur lequel tous les changements se produisent. Les architectes sont conscients de son importance et ont traditionnellement créé des modèles avec des éléments dupliqués ; modélisation consciente du temps permet de cloner les éléments, diagrammes et Paquetages existants.

Integrated Mobile Robotic Drive Units and Voice Assisted Forklifts Version 3.0 Q4 2019



### Où trouver Modélisation Consciente du Temps

Cloner la structure Paquetage en tant que nouvelle version

Ruban : Conception > Paquetage > Gérer > Cloner la structure en tant que nouvelle version

Fenêtre Navigateur Menu Contexte : Cloner la structure en tant que nouvelle version

Cloner Diagramme en tant que nouvelle version

Ruban : Conception > Diagramme > Gérer > Cloner comme nouvelle version

Cloner l'élément en tant que nouvelle version

Ruban : Conception > Élément > Gérer > Cloner l'élément en tant que nouvelle

version

Menu Contexte Diagramme : Cloner un élément en tant que nouvelle version

### Utilisation de Modélisation Consciente du Temps

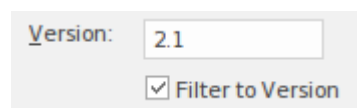
Modélisation Consciente du Temps peut être appliquée dans un large éventail de situations, des modèles commerciaux aux modèles technologiques. Elle montre notamment comment les entités représentées par le modèle évoluent au fil du temps. Les modèles de base (« As-Is », état actuel) peuvent être laissés inchangés tandis que n'importe quel nombre de modèles cibles (« To-Be », état futur) peuvent être construits pour refléter l'évolution possible des modèles de base au fil du temps. Il existe un large éventail de situations dans lesquelles un analyste ou un architecte trouvera cet outil utile, par exemple :

- Les architectes utilisent des modèles pour documenter l'état actuel, puis passent à la tâche plus difficile et plus gratifiante consistant à définir à quoi ressemblera l'état futur, souvent dans une série de transitions.
- Lors de l'acquisition de nouveaux clients, un Analyste Métier est souvent confronté au défi de décrire comment un produit de base doit être configuré pour différents groupes de clients, ce qui donne lieu à différentes versions du même produit.
- Les stratégies Métier prescrivent généralement à quoi ressemblera un modèle de capacité pour l'organisation du futur, ce qui donne lieu à au moins deux versions du modèle de capacité.
- Lors des fusions et acquisitions, Métier Architectes ont pour mission de décrire les états possibles de l'entreprise une fois la reprise réalisée
- Les ingénieurs sont tenus de développer des solutions meilleures ou plus efficaces pour répondre aux défis du futur, il faut donc définir des versions plus récentes et plus performantes de la solution.
- Les testeurs doivent être conscients des différentes versions d'un produit lors de la conception et de l'exécution des cas Test
- Les ingénieurs d'infrastructure doivent définir les environnements futurs en réponse aux problèmes de performances ou de sécurité en créant plusieurs versions de serveurs, de périphériques et même facilités entières.

Toutes ces situations nécessitent que le temps soit intégré dans les modèles afin qu'il puisse être raisonné et explicité. Les fonctionnalités de modélisation Time Aware d'Enterprise Architect peuvent être utilisées dans toutes ces situations pour garantir que le temps est inclus comme un élément de première classe dans les modèles. Le temps n'est pas mesuré ou modélisé en termes absolus ou relatifs, mais en représentant un nombre quelconque d'états futurs ou de différences sous forme de versions.

### Options de Modélisation Consciente du Temps

Les fonctionnalités Modélisation Consciente du Temps permettent à un modélisateur de cloner Paquetages, diagrammes et des éléments. La plupart des modèles ne sont pas triviaux et Enterprise Architect fournit une large gamme d'outils qui aideront à la visualisation des modèles et à la façon dont ils évoluent au fil du temps. La fenêtre de traçabilité sera particulièrement utile pour visualiser la connexion entre les éléments des modèles conscients du temps et d'autres parties du référentiel. Une fonctionnalité très utile est la possibilité d'appliquer un filtre à un diagramme en fonction de la version, masquant ainsi les éléments qui ne font pas partie d'une version particulière.



Apprenez Plus sur  
Modélisation Consciente

[Time Aware Models](#)



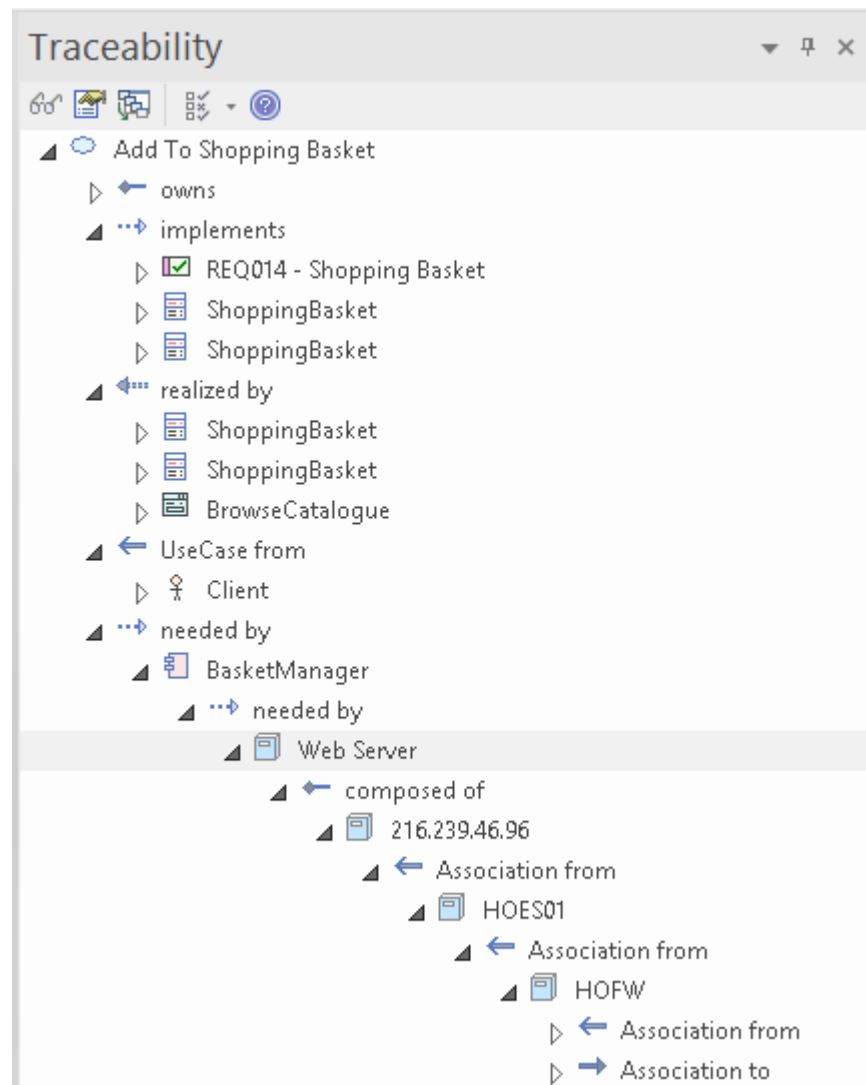
## **du Temps**

# Fenêtre de traçabilité

## Découvrir la fenêtre de traçabilité

### Présentation de la fenêtre de traçabilité

La fenêtre Traçabilité fournit une vue hiérarchique des connexions des éléments, ce qui permet de visualiser et d'interroger la traçabilité au fur et à mesure que les éléments sont parcourus dans le modèle. Cet outil est particulièrement utile car un modélisateur choisira souvent de masquer les relations diagramme, mais en sélectionnant un élément dans le diagramme et en affichant ses connexions dans la fenêtre Traçabilité, toutes ses relations seront révélées.



### Où trouver la fenêtre de traçabilité

Démarrer > Application > Conception > Traçabilité

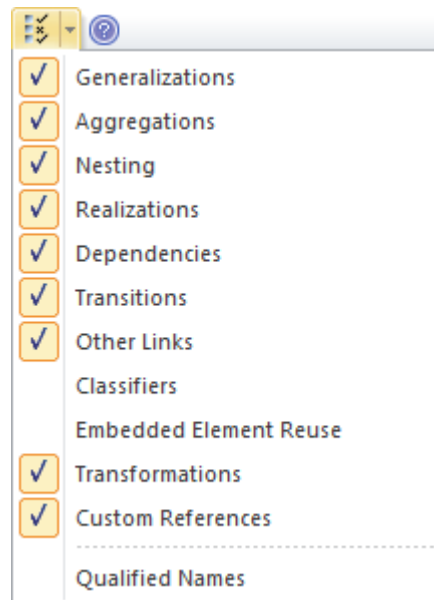
### Utilisation de la fenêtre de traçabilité

La fenêtre Traçabilité fournit une vue hiérarchique de la manière dont un élément est connecté aux autres éléments du référentiel, ainsi que du type de chaque relation. Cette fenêtre fournit une liste complète de toutes les relations qui ne peuvent pas être visualisées en visualisant les éléments dans la fenêtre Navigateur et qui peuvent également ne pas apparaître dans les diagrammes. Elle est très utile pour gérer Exigences et tracer la manière dont une exigence est liée aux éléments

de processus en amont tels que les pilotes Métier et aux éléments de processus en aval tels que les composants. C'est un outil utile, permettant aux nouveaux arrivants dans un modèle d'acquérir une compréhension rapide des éléments importants et bien connectés. Avant de supprimer un élément dans le modèle, vous devez utiliser la fenêtre Traçabilité pour vous assurer que vous comprenez les relations existantes de cet élément.

### Options pour la fenêtre de traçabilité

Il existe une série d'options qui limitent la traçabilité à des types de connecteurs spécifiques ; ces options peuvent être définies pour modifier ce qui est affiché dans la fenêtre. Les options sont disponibles dans la barre d'outils en haut de la fenêtre.



### Apprenez Plus sur la fenêtre de traçabilité

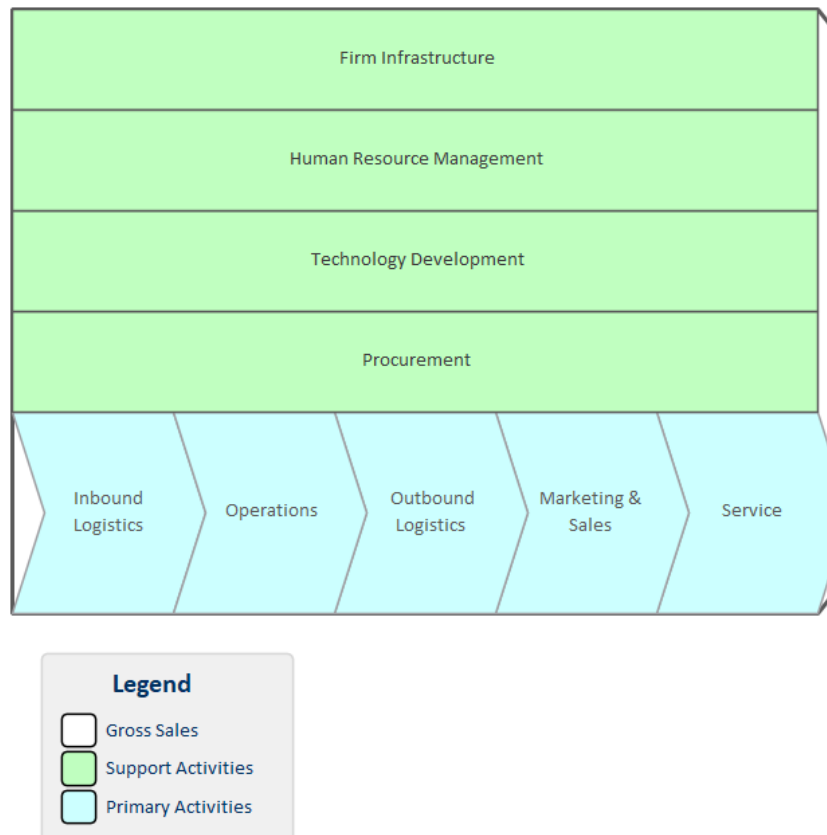
[The Traceability Window](#)

# Chaîne de valeur

## Connaître la chaîne de valeur

### Présentation de la chaîne de valeur

La chaîne de valeur est un diagramme stratégique qui permet de modéliser les activités primaires et secondaires d'une organisation. Le diagramme peut être créé à partir d'un Motif qui additionne et relie les cinq activités primaires d'une chaîne et les quatre activités de soutien qui les sous-tendent.



L'élément Ventés brutes sert de cadre pour les éléments Activité principale et Activité Support .

Un analyste travaillant au niveau d'une unité commerciale stratégique sera souvent amené à modéliser les activités que l'unité commerciale exécute pour fournir valeur à ses clients. La chaîne de valeur est l'outil privilégié pour créer cette représentation stratégique de la séquence d'activités qu'une organisation exécute.

### Où trouver la chaîne de valeur

Ruban : Conception > Diagramme > Ajouter Diagramme > Modélisation Stratégique > Chaîne de valeur

Fenêtre Navigateur Barre d'outils : Icône Nouveau Diagramme > Modélisation Stratégique > Chaîne de valeur

Menu contextuel de la fenêtre Navigateur | Ajouter Diagramme ... > Modélisation Stratégique > Chaîne de valeur

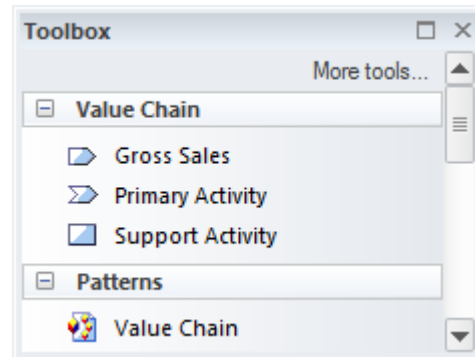
### Utilisation de la chaîne de valeur

La chaîne de valeur est un outil important d'aide à la planification stratégique qui permet de comprendre l'ensemble de la séquence (ou chaîne). Elle permet

également de décomposer la chaîne en ses activités constitutives, ce qui permet d'évaluer les coûts, les ressources et valeur et de les améliorer potentiellement.

### Options pour la chaîne de valeur

Chacune des activités principales et de soutien peut être liée à d'autres éléments du modèle, y compris un document lié et des éléments qui définissent des repères.



Le diagramme de la chaîne de valeur (comme tout diagramme ) peut être considéré comme une liste d'éléments, ce qui facilite le travail avec les propriétés de l'élément.

Filtres de Diagramme peuvent également être utilisés lors de la présentation des diagrammes pour attirer l'attention sur certaines parties des diagrammes .

### Apprenez Plus sur la Chaîne de Valeur

[Value Chains](#)

