



ENTERPRISE ARCHITECT

Série de Guides d'Utilisateur

Decision Model and Notation (DMN)

Author: Sparx Systems

Date: 23/11/2023

Version: 16.1

CRÉÉ AVEC  **ENTERPRISE
ARCHITECT**

Table des Matières

Decision Model and Notation (DMN)	4
Modélisation avec DMN	7
Décision	8
Modèle de connaissances Métier	10
Paramètres BKM	12
Valeurs des paramètres d'entrée pour Simulation	14
Exemple Simulation Tableau de Décision	16
Exemple Simulation d'expression littérale	18
Des données d'entrée	20
Expression DMN de données d'entrée	21
Définition de l'élément	23
Barre d'outils de définition Item	25
Définitions d'éléments et Ensembles de données	26
Types de composants	29
Énumérations de valeurs autorisées	31
Ensembles de données	33
Échanger Ensembles de données à l'aide de DataObjects	36
Service Décision	40
Simuler un service Décision	43
Démarrage	45
Plus d'information	47
Créer un Modèle Décision	48
Intégrer dans BPSim pour Simulation	55
Exemple : Intégrer le service Décision DMN dans Object de données BPSim et Paramètres Propriété	60
Exemple : Intégrer DMN Métier Knowledge Modèle dans BPSim Paramètres Propriété	61
Importation de XML DMN	62
Diagrammes Exigences Décision	64
Éditeur d'expression Décision	66
Relation	68
Liste encadrée	71
Modifier Dialogue d'expression DMN	74
Expression littérale	77
Barre d'outils pour l'éditeur d'expression littérale	80
Exemple - Remboursement du prêt	81
Invocation	83
Barre d'outils pour l'éditeur d'appels	86
Exemple 1 - Lier les données d'entrée au Modèle de connaissances Métier	88
Exemple 2 - Lier des variables d'entrée de contexte à Métier Knowledge Modèle	90
Achèvement automatique des expressions DMN	91
Validation des expressions DMN	95
Contexte encadré	97
Barre d'outils pour l'éditeur de contexte encadré	100
Exemple - Calcul des mensualités du prêt	101
Tableau de Décision	106
Barre d'outils pour l'éditeur Tableau de Décision	114
Tableau de Décision Hit Policy	116
Validation Tableau de Décision	119

Intégrer dans un élément de classe UML	122
Exemple Diagramme	129
Module de génération de code et Test	131

Decision Model and Notation (DMN)

Créer et simuler des modèles détaillés de Décisions d'entreprise

Les organisations sont confrontées à des environnements opérationnels de plus en plus difficiles, caractérisés par une concurrence féroce et souvent imprévisible de la part des acteurs du marché existants et nouveaux, des changements dans les réglementations gouvernementales et industrielles et des bouleversements dans le tissu social de leur clientèle. Les décisions qu'une organisation prend dans ce contexte sont essentielles à son succès et à sa capacité à se frayer un chemin en toute sécurité dans ces eaux inexplorées. Grâce fonctionnalités Decision Model and Notation (DMN) d'Enterprise Architects, vous pouvez non seulement modéliser les décisions prises par votre organisation, mais vous pouvez également exécuter des simulations à partir de ces modèles pour prédire les résultats en fonction d'exemples d'ensembles de données. La puissance du langage réside dans le fait que les hommes d'affaires peuvent facilement comprendre et travailler avec diagrammes Exigences Décision simples mais expressifs qui détaillent quelles sont les décisions, quelles sont les entrées dans les décisions et quels sont les résultats attendus. Les règles peuvent être documentées de plusieurs manières, notamment tableaux de décision faciles à définir. Une fois terminés, ces diagrammes accompagnés d'exemples de données d'entrée peuvent être simulés pour montrer les résultats des décisions.

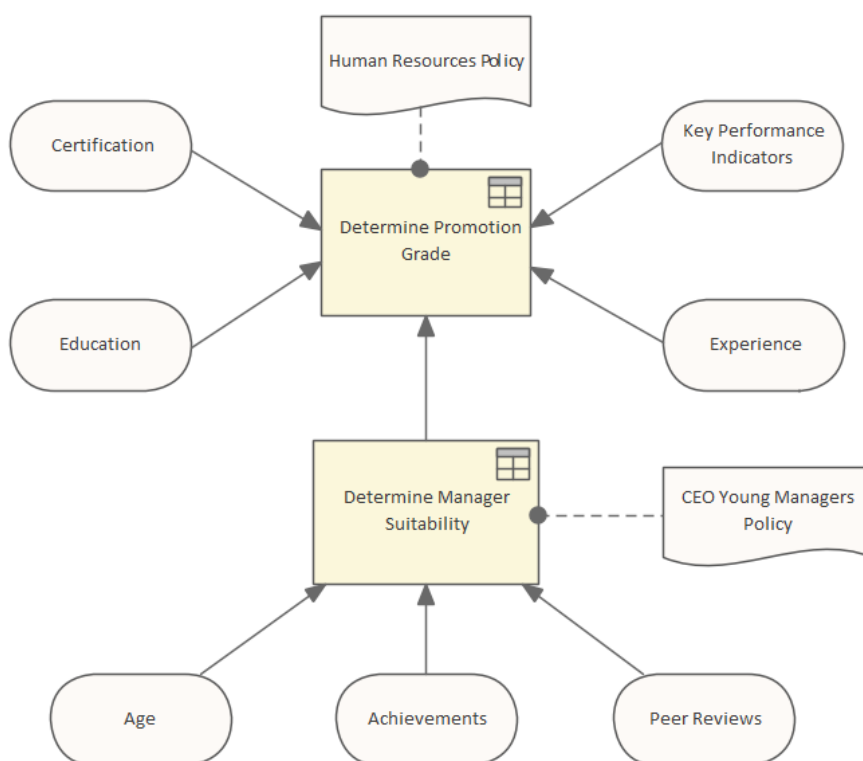


diagramme Décision Exigences montrant une Décision avec un Modèle de Connaissance Métier et un certain nombre d'entrées dont une autre Décision .

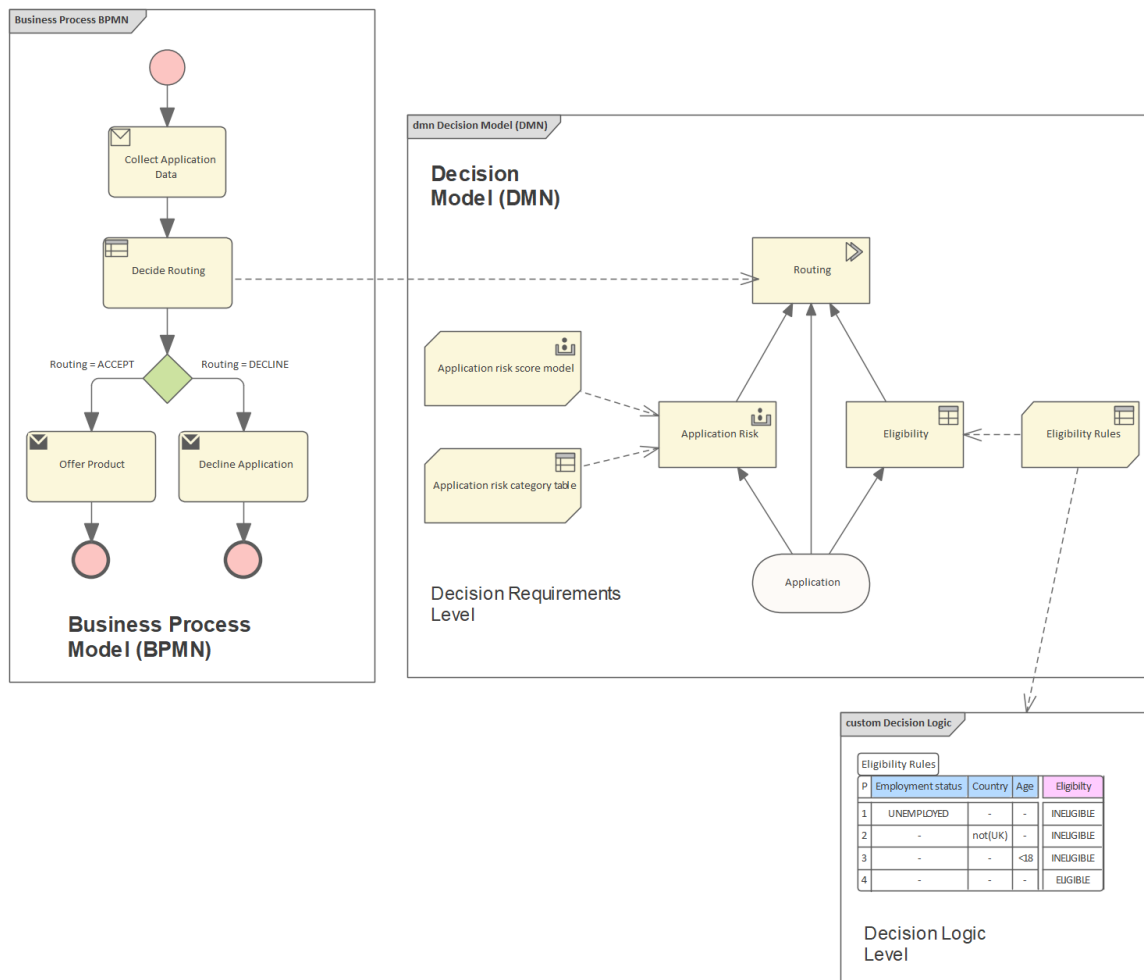
Une fois que ces modèles ont été définis, simulés et testés par l'entreprise, les technologues et les ingénieurs peuvent ensuite affiner ces modèles et générer automatiquement des artefacts logiciels, notamment du code de programmation, directement à partir des modèles, réduisant ainsi la possibilité d'erreurs d'interprétation et réduisant le temps de mise en œuvre.

Qu'est-ce que le DMN ?

DMN est destiné à fournir un pont entre les modèles de processus métier et les modèles logiques de décision :

- Les modèles de processus Métier définiront les tâches au sein des processus métier où la prise de décision doit avoir lieu

- Diagrammes Exigences Décision définiront les décisions à prendre dans ces tâches, leurs interrelations et leurs exigences en matière de logique décisionnelle.
- La logique Décision définira les décisions requises de manière suffisamment détaillée pour permettre la validation et/ou l'automatisation.



Pris ensemble, diagrammes Décision Exigences et la logique de décision vous permettent de construire un Modèle Décision complet qui complète un modèle de processus métier en spécifiant - en détail - la prise de décision effectuée dans les tâches de processus.

DMN fournit des constructions couvrant à la fois les exigences de décision et modélisation de la logique de décision.

- Pour modélisation des exigences de décision, il définit le concept de Graphe d' Exigences Décision (DRG) comprenant un ensemble d'éléments et leurs règles de connexion, et une notation correspondante : le Diagramme Exigences Décision (DRD).
- Pour modélisation de la logique de décision, il fournit un langage appelé FEEL pour définir et assembler Tableaux Décision , des calculs, une logique if/then/else, des structures de données simples et une logique définie en externe à partir de Java et PMML dans des expressions exécutables avec une sémantique formellement définie.

Avantages de l'utilisation de DMN dans Enterprise Architect

Modélisation des processus décisionnels à l'aide de DMN vous permet d'enregistrer, de spécifier et d'analyser des processus décisionnels complexes en tant que système de décisions, de règles métier, d'ensembles de données et de sources de connaissances interdépendants. Ce faisant, vous pouvez décomposer un processus décisionnel très complexe en un réseau de décisions de soutien et de données d'entrée. Cela facilite une compréhension plus facile du processus global, supporte la refactorisation des processus et simplifie la tâche de validation du processus, en vous permettant de valider facilement les étapes individuelles qui composent le processus global.

Lorsque vous créez un Décision Modèle dans Enterprise Architect à l'aide de DMN, vous pouvez exécuter des simulations du modèle pour vérifier l'exactitude du modèle. Après avoir vérifié votre modèle, vous pouvez générer un module DMN en Java, JavaScript, C++ ou C#. Le module DMN généré peut être utilisé avec Enterprise Architect BPSim Moteur d'Exécution, Statemachine Exécutable ou dans un système logiciel distinct que vous implémentez.

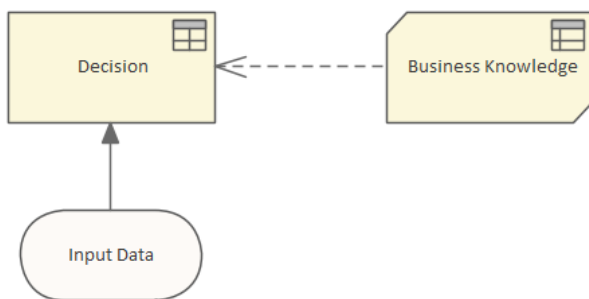
Enterprise Architect fournit également un « module Test » facilité, qui est un pré-processus pour l'intégration de DMN avec BPMN. L'objectif est de produire des éléments BPMN2.0::DataObject, puis de les utiliser pour vérifier qu'une décision cible spécifiée est correctement évaluée avec le module DMN. Vous configurez ensuite BPSim en chargeant DataObjects et en attribuant des décisions de module DMN à BPSim Propriétés.

Cette fonctionnalité est disponible dans les éditions Unified et Ultimate d' Enterprise Architect, à partir de la version 15.0.

Graphiques Exigences Décision

Le modèle d'exigence de décision DMN consiste en un graphique Exigences Décision (DRG) représenté dans un ou plusieurs Diagrammes Exigences Décision (DRD). Les éléments modélisés sont les décisions, les domaines de connaissances métiers, les sources de connaissances métiers, les données d'entrée et les services de décision.

Un DRG est un graphique composé d'éléments reliés par des exigences, et est autonome dans le sens où toutes les exigences modélisées pour toute Décision dans le DRG (ses sources immédiates d'information, de connaissances et d'autorité) sont présentes dans le même DRG. Il est important de distinguer cette définition complète du DRG d'un DRD présentant une vue particulière de celui-ci, qui peut être un affichage partiel ou filtré.



Modélisation avec DMN

Cette rubrique vous présente les éléments les plus importants dont vous avez besoin pour créer des modèles Décision . Comme indiqué dans les sujets précédents, un modèle de décision comporte deux parties fondamentales, à savoir le diagramme Exigences Décision et la logique de décision. La création d'un diagramme de décision est simple et la partie la plus lourde de l'exercice sera probablement de découvrir la manière dont une organisation prend des décisions et quelles sont les contributions à ces décisions. Les diagrammes contiennent généralement des décisions enchaînées décrivant le fait qu'une décision peut contribuer à une autre décision et ainsi de suite.

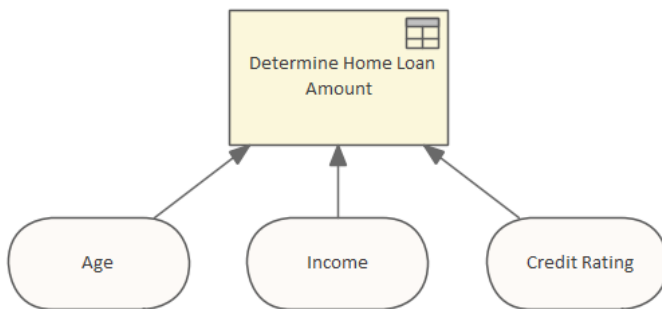


diagramme Exigences Décision simple montrant trois entrées dans une Décision à l'aide d'un Tableau de Décision .

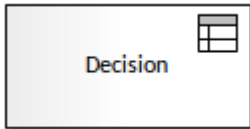
La logique de la décision est décrite à l'aide d'un certain nombre de dispositifs mais la forme la plus couramment utilisée et accessible est le Tableau de Décision . Le Tableau de Décision contient des lignes et des colonnes comme une feuille de calcul et couvre toutes les combinaisons possibles d'entrées pour produire un certain nombre de sorties. Par exemple, si un demandeur a plus de 21 ans et moins de 65 ans et gagne 60 000 \$ par an avec une bonne cote de crédit, la banque lui prêtera 300 000 \$ pour un prêt immobilier.

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

Un Tableau de Décision montrant trois entrées et une sortie, des lignes seraient ajoutées pour définir les règles.

Décision

Un élément Décision est utilisé pour évaluer une sortie basée sur une ou plusieurs entrées. La logique qui détermine le résultat est soit définie dans cet élément Décision, soit elle invoque la logique de décision contenue dans un Métier Knowledge Modèle qui est connecté à la Décision.



Contributions

Une Décision peut avoir n'importe quel nombre d'entrées, y compris la possibilité de définir les valeurs d'entrée dans l'élément. L'entrée la plus courante consiste à utiliser un élément de données d'entrée.

Sortir

Une Décision peut avoir zéro ou un résultat. Le résultat peut être un ensemble de données complexe.

Expressions de valeur

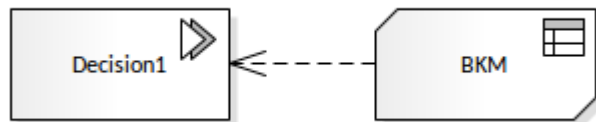
La sortie d'un élément Décision est déterminée à l'aide d'une expression de valeur. L'expression de valeur contient la logique de décision de l'élément et peut prendre l'une des quatre formes suivantes : Tableau de Décision, Expression Littérale, Invocation ou Contexte Encadré. Les expressions de valeur sont définies et modifiées à l'aide de l'éditeur d'expression DMN, qui affiche l'un des quatre formats en fonction du type d'expression utilisé.

Lorsqu'il est affiché sur un diagramme, l'élément Décision affiche une icône dans le coin supérieur droit qui indique le type d'expression de valeur qu'il utilise.

Type	Description
	Un Tableau de Décision est une représentation tabulaire d'un ensemble d'expressions d'entrée et de sortie associées, organisées en règles indiquant quelle entrée de sortie s'applique à un ensemble spécifique d'entrées d'entrée.
	Une expression littérale est la forme la plus simple d'expression DMN. Il est communément défini comme une instruction sur une seule ligne ou un bloc conditionnel if-else.
	Une Invocation Décision nécessite qu'un élément Métier Knowledge Modèle soit référencé à l'aide d'un connecteur Knowledge Requirement. L'élément Décision contient simplement les paramètres qui fournissent le contexte d'évaluation du Métier Knowledge Modèle (BKM). Une partie ou la totalité du résultat renvoyé par le BKM peut être configurée pour être transmise comme sortie de Décision.
	Un contexte encadré est une collection d'entrées de contexte. Chaque entrée de contexte se compose d'une variable et d'une expression. Le Contexte a également une valeur de résultat.

Modèle de connaissances Métier

Un élément Métier Knowledge Modèle (BKM) représente un élément de logique de décision réutilisable. Typiquement, il est connecté à un élément Décision qui invoque le BKM et transmet un ensemble d'entrées. Le BKM, en utilisant sa logique interne, évalue une sortie qui est renvoyée à la Décision.



À moins qu'un BKM ne travaille sur des valeurs fixes, il nécessite généralement de définir un ensemble de paramètres d'entrée, ainsi que la définition d'une sortie. Les paramètres et la logique de décision sont définis à l'aide de la fenêtre DMN Expression.

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

Entrées et sorties

Lorsqu'il est utilisé dans un modèle de décision, un BKM doit être connecté via un KnowledgeRequirement à une Décision ou à un autre BKM, à travers lequel il reçoit ses entrées. Les paramètres d'entrée sont définis à l'aide de l'icône . Ceux-ci peuvent être définis comme un type simple ou un type complexe défini à l'aide d'un ItemDefinition. La dénomination des paramètres d'entrée influence la dénomination dans l'expression de valeur.

Sortir

Une sortie BKM se fait via une KnowledgeRequirement qui doit être une entrée dans une Décision ou dans un autre BKM. La sortie est définie à l'aide de :

- L'icône pour une expression littérale
- Colonne(s) de sortie dans le tableau DMN Expression pour un Tableau de Décision, Boxed Content et Invocation.

Une sortie peut être un type simple ou un type complexe défini à l'aide d'un ItemDefinition.

Expressions de valeur


Pour définir un moyen d'évaluation d'un résultat, basé sur la logique de décision, un élément Métier Knowledge Modèle (BKM) contient une expression de valeur. Ceci est défini et modifié à l'aide de la fenêtre Expression DMN, qui comporte quatre formats, le format étant déterminé par le type d'expression de valeur que vous souhaitez utiliser.


L'élément BKM peut être défini avec ces structures pour l'expression de valeur. Chacun est représenté dans le modèle avec une icône.

Type	Description
	Un Tableau de Décision est une représentation tabulaire d'un ensemble d'expressions d'entrée et de sortie associées, organisées en règles indiquant quelle entrée de sortie s'applique à un ensemble spécifique d'entrées d'entrée.
	Une expression littérale est la forme la plus simple d'expression DMN. Il est communément défini comme une instruction sur une seule ligne ou un bloc conditionnel if-else.
	Une invocation Décision nécessite qu'un élément du modèle de connaissances Métier soit référencé à l'aide d'un connecteur d'exigence de connaissances. Il contient simplement les paramètres qui fournissent le contexte pour l'évaluation d'un modèle de connaissances métier.
	Un contexte encadré est une collection d'entrées de contexte. Chaque entrée de contexte se compose d'une variable et d'une expression. Le Contexte a également une valeur de résultat.

Validation et Tester

Pour garantir qu'un élément BKM est capable de produire une sortie correcte, il peut être validé à l'aide de l'icône

Validation . Un BKM peut également être testé en tant qu'unité pour garantir son fonctionnement à l'aide du bouton

Simulation . Pour plus de détails, consultez la rubrique d'aide *Valeurs des paramètres d'entrée pour Simulation*.


Paramètres BKM

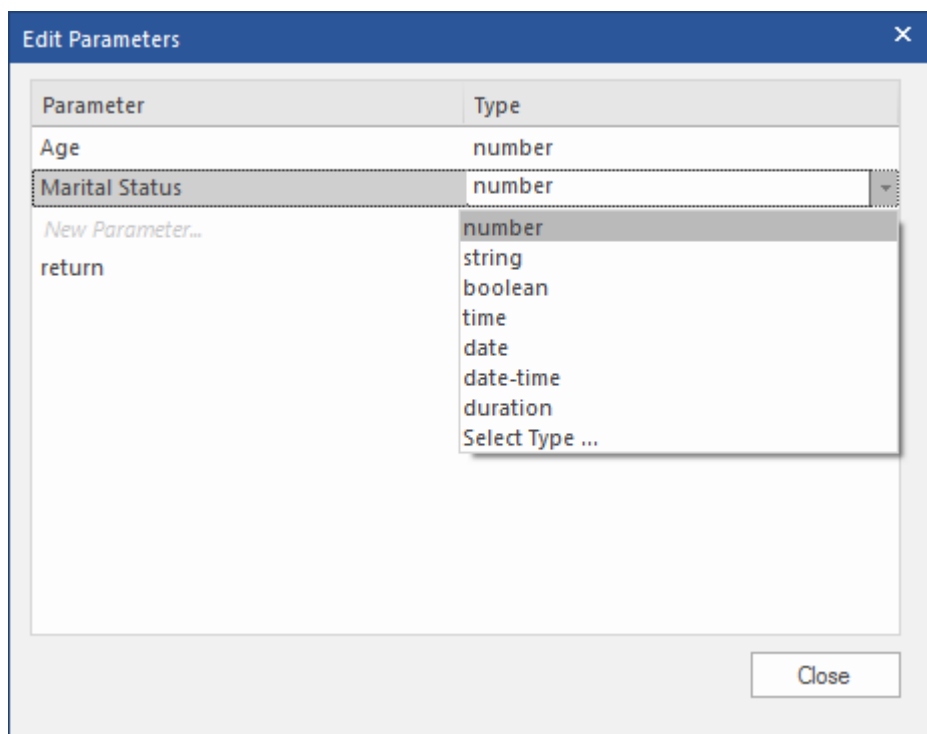
Un Métier Knowledge Modèle (BKM) est implémenté sous la forme d'une définition de fonction, avec des paramètres et une expression DMN comme corps (telle que Tableau de Décision , Boxed Context ou Literal Expressions).

Comme un BKM est destiné à fonctionner de manière autonome et à être appelé par des décisions ou d'autres BKM, il est nécessaire de définir tous les paramètres d'entrée. De plus, pour les expressions littérales, vous devez définir le paramètre de sortie.

Lors de la définition de paramètres d'entrée, vous pouvez les définir avec des valeurs par défaut pour les tests. Après avoir créé un BKM, pour vérifier qu'il fonctionne correctement, vous pouvez exécuter une simulation basée sur ces valeurs par défaut.

Paramètres d'un Modèle de connaissances Métier

Pour ouvrir la dialogue 'Modifier les paramètres', dans la fenêtre Expression DMN, cliquez sur le bouton Modifier les paramètres  :



Note : il s'agit d'un exemple d'expression littérale qui inclut un type *de retour* .

Modifier les paramètres

Vous pouvez effectuer ces actions sur les paramètres :


Action	Description
	Ajoutez un nouveau paramètre en tapant dans la ligne "Nouveau paramètre..."
	Modifiez le nom du paramètre existant en éditant sur place dans la cellule.

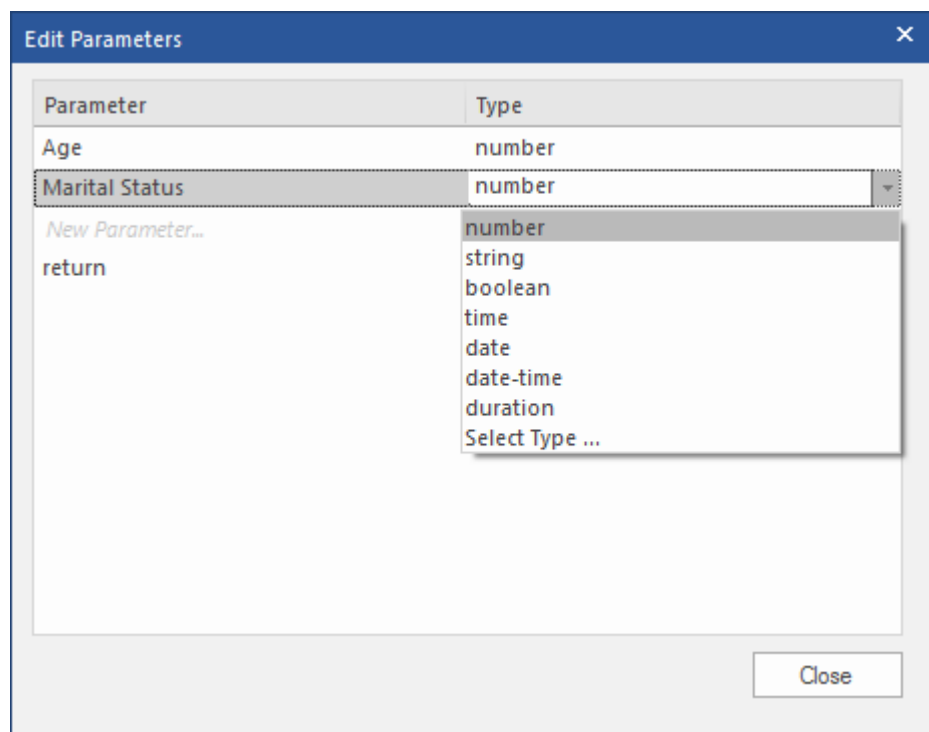
	Supprimez un paramètre existant à l'aide du menu contextuel.
	<p>Cliquez sur le Type pour activer une liste déroulante. Sélectionnez un type pour le paramètre dans la liste déroulante.</p> <p>Définir un Type de définition Item</p> <p>Lors de la modification du type de paramètre, il existe une option permettant de sélectionner un type prédéfini dans une ItemDefinition. L'option pour cela est « Sélectionner Type ... ». Lorsque cette option est sélectionnée, elle ouvrira un dialogue pour sélectionner une ItemDefinition.</p>

Valeurs des paramètres d'entrée pour Simulation

Un Métier Knowledge Modèle étant autonome, il est possible d'effectuer un « Test unitaire » de simulation en fournissant un ensemble de valeurs par défaut en entrée pour ses paramètres. Ces valeurs peuvent être définies dans l'onglet *Valeurs des paramètres d'entrée pour Simulation* de la fenêtre Expression DMN.


Paramètres d'un Métier Knowledge Modèle (BKM)

Les paramètres d'un BKM sont accessibles à partir de la fenêtre Expression DMN, à l'aide du bouton Modifier les paramètres  dans la barre d'outils :



Un ensemble de valeurs par défaut pour ces paramètres, qui peuvent être utilisés dans une simulation du BKM, sont définis dans l'onglet « Valeurs des paramètres d'entrée pour Simulation ». dans la fenêtre Expression DMN :

Application risk score model		Input Parameter Values for Simulation
Application risk score model . Partial score		
Age : number		35
Marital Status : Marital Status		"M"
Employment Status : Employment St.		"EMPLOYED"

Une fois ces paramètres définis, le BKM peut être testé à l'aide du bouton  Simulation .

Exemples Simulation

Voici deux exemples d'utilisation des *valeurs des paramètres d'entrée pour Simulation* .

Type	Description
------	-------------

Tableau de Décision	Un exemple de simulation d'un élément Tableau de Décision BKM basé sur les valeurs définies dans l'onglet <i>Valeurs des paramètres d'entrée pour Simulation</i> .
Expression littérale	Un exemple de simulation d'un élément d'expression littérale BKM basé sur les valeurs définies dans l'onglet <i>Valeurs des paramètres d'entrée pour Simulation</i> .

Exemple Simulation Tableau de Décision

L'exemple Métier Knowledge Modèle (BKM) décrit dans cette section est disponible depuis l'Assistant de Modèle (Ctrl+Shift+M). Sélectionnez un Paquetage hôte dans votre modèle, invoquez l'Assistant de Modèle et - depuis le menu déroulant Perspectives - sélectionnez 'Exigences | Décision Modélisation'.

Pour accéder à l'exemple utilisé dans cette section :

- Créer un motif pour 'DMN Décision | Un exemple complet'
- Naviguez dans la fenêtre Navigateur jusqu'à « Un exemple complet | Modèles de connaissances Métier

Il est également disponible dans le modèle Enterprise Architect Sample (EAExample) :

- Naviguez dans la fenêtre Navigateur jusqu'à 'Analyse et Modélisation Métier > Exemples DMN > Un exemple complet > Métier Knowledge Models'


Double-cliquez sur l'élément 'Règles d'éligibilité' pour ouvrir le BKM dans la fenêtre Expression DMN

Lorsqu'un Tableau de Décision est créé pour un Métier Knowledge Modèle, nous pouvons tester ce BKM en liant certaines valeurs :

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

Nous pouvons fournir des valeurs de test telles que celles-ci :

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

Cliquez sur le bouton Simulation  de la barre d'outils pour obtenir ce résultat :

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

- La valeur du paramètre d'exécution remplacera les « Valeurs autorisées » en mode simulation.
- Les règles valides sont mises en évidence
- Puisque la politique Hit de ce Tableau de Décision est P (Priorité), le résultat final est déterminé par l'ordre des valeurs de sortie ; puisque 'INELIGIBLE' et 'ELIGIBLE' sont les valeurs de sortie et que 'INELIGIBLE' précède 'ELIGIBLE', la règle n°3 donnera le résultat final et ce candidat est 'INELIGIBLE'.

Exemple Simulation d'expression littérale

Le Métier Knowledge Modèle (BKM) décrit dans cette section est disponible depuis l' Assistant de Modèle (Ctrl+Shift+M). Sélectionnez un Paquetage hôte dans votre modèle, invoquez l' Assistant de Modèle et - depuis le menu déroulant Perspectives - sélectionnez ' Exigences | Décision Modélisation '.

Pour accéder à l'exemple utilisé dans cette section :

- Créer un motif pour 'DMN Métier Knowledge Modèle > Métier Knowledge Modèle Literal Expression'
- Naviguez dans la fenêtre Navigateur jusqu'à « Métier Knowledge Modèle Literal Expression > Payment »

Il est également disponible dans le modèle Enterprise Architect Sample (EAExample) :

- Naviguez dans la fenêtre Navigateur jusqu'à ' Simulation de Modèle > DMN Models > Métier Knowledge Modèle > Métier Knowledge Modèle Literal Expression'

Double-cliquez sur l'élément 'Paiement' pour ouvrir le BKM dans la fenêtre DMN Expression.

Semblable à un Tableau de Décision , le Modèle de Connaissances Métier implémenté sous forme d'Expression Encadrée peut également être testé.

Prenons l'exemple de l'élément « Paiement ». Ce BKM calculera le remboursement mensuel en fonction du taux d'intérêt, du nombre de termes et du montant du principal.

Nous pourrions fournir des valeurs de test telles que celles-ci :

Parameter	Value
Payment	
Rate	0.04 / 12
Term	30 * 12
Principle	300000

Cliquez sur le bouton Simulation  dans la barre d'outils ; ce résultat est obtenu :

Le paramètre d'exécution et les valeurs de retour seront affichés avec un signe égal '=' suivi de la valeur d'exécution. Cette valeur est également affichée sous forme d'étiquette par rapport à l'élément sur son diagramme parent.

Dans cet exemple, étant donné un taux annuel de 4 % pendant 30 ans et un capital de 300 000 \$, le remboursement

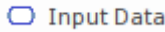
mensuel est de 1 432,25 \$.

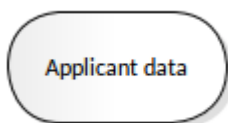
Note : La DMN Bibliothèque dispose déjà d'une fonction PMT définie ; cet exemple montre principalement comment fonctionne l'expression littérale et comment la tester avec un ensemble d'arguments.

Des données d'entrée

Un élément InputData est utilisé pour saisir dans Decisions un ensemble de valeurs provenant de l'extérieur du modèle. Cet ensemble de valeurs est utilisé pour évaluer les décisions. Il dérive son type et un ensemble de valeurs d'un ItemDefinition.

Aperçu


Les éléments InputData sont créés en faisant glisser une icône  de la boîte à outils sur un diagramme DMN.



Le nom de l'élément InputData doit être unique et ne pas dupliquer le nom d'un autre Décision , InputData, Métier Knowledge Modèle , Décision Service ou Import dans le modèle de décision.

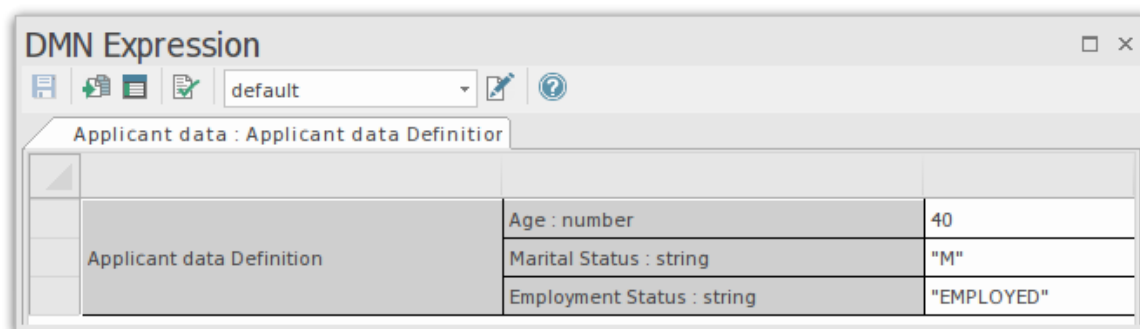
Référencer une ItemDefinition

La structure des données, ainsi que les ensembles de valeurs pour un élément InputData, sont définis dans un élément ItemDefinition. Un élément DMN InputData doit être référencé (tapé) par un ItemDefinition soit :

- En cliquant sur l'icône  dans la fenêtre DMN Expression de l'élément InputData ou
- En sélectionnant l'élément InputData et en appuyant sur Ctrl+L pour sélectionner la ItemDefinition dans le dialogue

Propriétés des données d'entrée

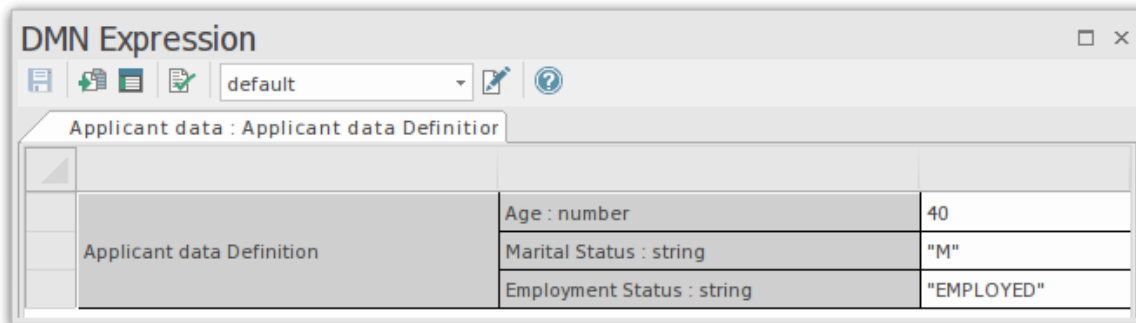
Les propriétés d'un élément InputData sont accessibles via la fenêtre DMN Expression. Double-cliquez sur l'élément InputData pour ouvrir cette fenêtre.



La fenêtre Expression DMN fournit une vue de la structure des données ainsi qu'un accès aux Ensembles de données pouvant être utilisés dans les simulations.

Expression DMN de données d'entrée






La fenêtre DMN Expression fournit une vue de la structure de données d'un InputData, des options pour modifier la valeur de Items et un accès aux Ensembles de données qui peuvent être utilisés dans les simulations.



Accéder

Ruban	Simuler > Décision Analysis > DMN > DMN Expression, puis sélectionner/créer un InputData
Autre	Double-cliquez sur un élément DMN InputData

Options barre d'outils

Option	Description
	Enregistre la configuration dans l'élément InputData actuel.
	Ensembles le type de InputData en sélectionnant une référence à un ItemDefinition.
	Ouvre l'élément ItemDefinition référencé par ce InputData en tant que définition de type.
	Exécute une validation de InputData. Enterprise Architect effectuera une série de validations pour vous aider à identifier les erreurs dans InputData.
	Option permettant de sélectionner un ensemble de données tel que défini dans ItemDefinition qui fait référence à ce InputData.
	Ouvre le dialogue pour éditer les ensembles de données pour ces données d'entrée. Chaque InputData peut définir plusieurs ensembles de données. Grâce à cette fonctionnalité, DMN Simulation peut tester rapidement les résultats d'une décision en choisissant différents ensembles de données.

Achèvement automatique


Si les données d'entrée ont un champ avec « Valeur autorisée » défini, le champ peut être rempli en sélectionnant le champ, en appuyant sur la barre d'espace, puis en sélectionnant une option dans la liste déroulante.

The screenshot shows the 'DMN Expression' window. At the top, there is a toolbar with icons for file operations and a dropdown menu set to 'default'. Below the toolbar, the main area is titled 'Applicant Data : tApplicantData'. It contains a table with the following data:

	Age : number	25
	MartitalStatus : string	"M"
	EmploymentStatus : string	e
	ExistingCustomer : boolean	"EMPLOYED"
		"SELF-EMPLOYED"
	Monthly	60000
		20
		3000

A dropdown menu is open for the 'EmploymentStatus : string' field, showing the following options: "EMPLOYED", "SELF-EMPLOYED", "STUDENT", and "UNEMPLOYED".


Ensembles de données

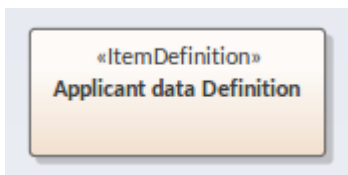
Ensembles de données sont définis dans la ItemDefinition référencée par l'élément InputData. À l'aide de la liste déroulante de la barre d'outils, vous pouvez sélectionner un ensemble de données dans ItemDefinition. Une fois qu'un ensemble est sélectionné, vous pouvez modifier les valeurs des éléments. Vous pouvez également ajouter de nouveaux Ensembles de données en ouvrant la fenêtre Modifier l'ensemble de données à l'aide de l'icône .

Définition de l'élément

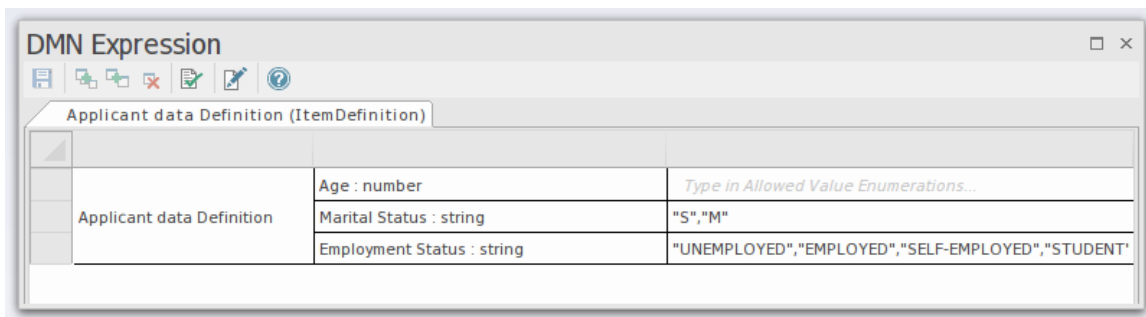
La définition de la structure des éléments de données utilisés dans le modèle est fondamentale pour la création de modèles Décision . Une ItemDefinition est utilisée pour définir la structure des données d'entrée et, éventuellement, pour restreindre la plage de valeurs autorisées des données. Les définitions d'éléments peuvent aller d'un simple type unique à un type structuré complexe.

Aperçu

Les éléments ItemDefinition sont créés en faisant glisser une icône  Item Definition de la page DMN Toolbox vers un diagramme DMN.



Les propriétés principales d'un élément ItemDefinition sont accessibles via la fenêtre DMN Expression.



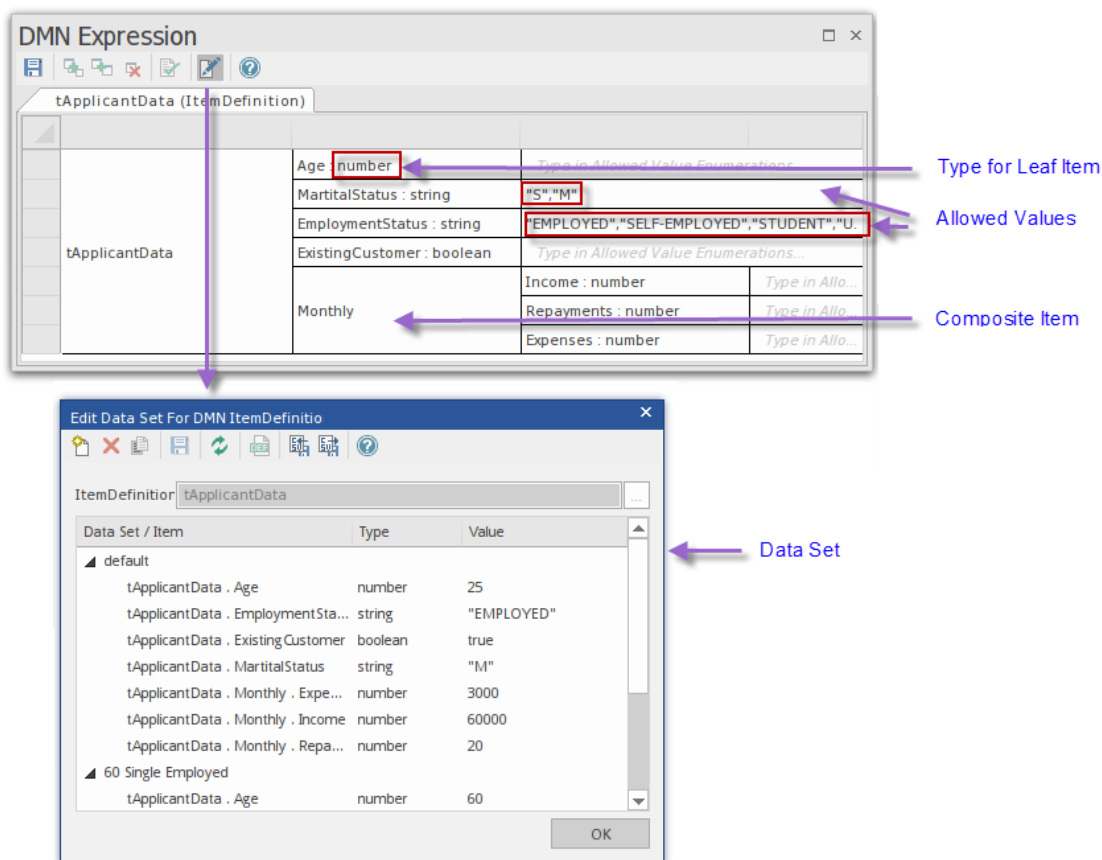
Accéder

Pour ouvrir la fenêtre Expression DMN pour un élément ItemDefinition :

Ruban	Simuler > Analyse Décision > DMN > Expression DMN, puis sélectionnez ou créez une ItemDefinition
Autre	Double-cliquez sur une définition d'élément DMN

Expression DMN et ensemble de données

Cette image est une présentation de la fenêtre Expression DMN, montrant un élément de données complexe et la disposition des champs clés utilisés dans la définition des données. Inclus est une vue d'un ensemble de données défini à l'aide de cette ItemDefinition. Un ensemble de données est une « instance » de données conforme à une ItemDefinition, qui contient un ensemble de valeurs à utiliser dans la simulation DMN.



Comme les ItemDefinitions sont des éléments de base du modèle, il est recommandé de les valider avant de les utiliser dans le modèle. Cela garantira que tous les problèmes seront résolus dès le début du processus de création d'un modèle complexe.







Pour plus de détails sur la configuration de ItemDefinitions, consultez ces rubriques d'aide :

- *Définition Item DMN, ensemble de données et données d'entrée*
- *Types de composants*
- *Valeur autorisée de la définition d'élément*
- *Achèvement automatique des expressions DMN*
- *Validation des expressions DMN*

Barre d'outils de définition Item

Ce tableau fournit des descriptions des fonctionnalités accessibles dans la fenêtre DMN Expression lorsqu'un élément ItemDefinition est sélectionné.

Options barre d'outils

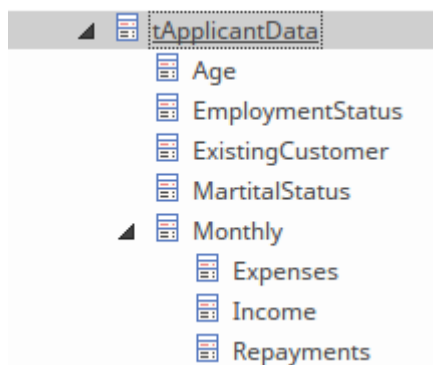
Option	Description
	Enregistre la configuration de la ItemDefinition actuelle.
	Crée un nouveau composant de données en tant qu'enfant du composant sélectionné.
	Crée un nouveau composant de données en tant que frère du composant sélectionné.
	Supprime le composant de données sélectionné.
	Valide la ItemDefinition ; Enterprise Architect effectuera une série de validations pour vous aider à identifier les erreurs dans ItemDefinition.
	Ouvre la dialogue « Modifier l'ensemble de données », dans laquelle vous pouvez créer et modifier des « instances » de ItemDefinition à utiliser par les éléments InputData.

Définitions d'éléments et Ensembles de données

Une ItemDefinition décrit les types et les structures des éléments de données utilisés dans un modèle Décision . Il sert de définition du type de données pour les éléments InputData, les éléments Décision et les paramètres Métier Knowledge Modèle . Un ItemDefinition peut également définir des ensembles de données qui fournissent des ensembles de valeurs à utiliser dans les simulations DMN. Le passage d'un ensemble de données à l'autre offre la possibilité d'effectuer une analyse de type « et si » à l'aide du modèle Décision .


Structure de définition d'élément

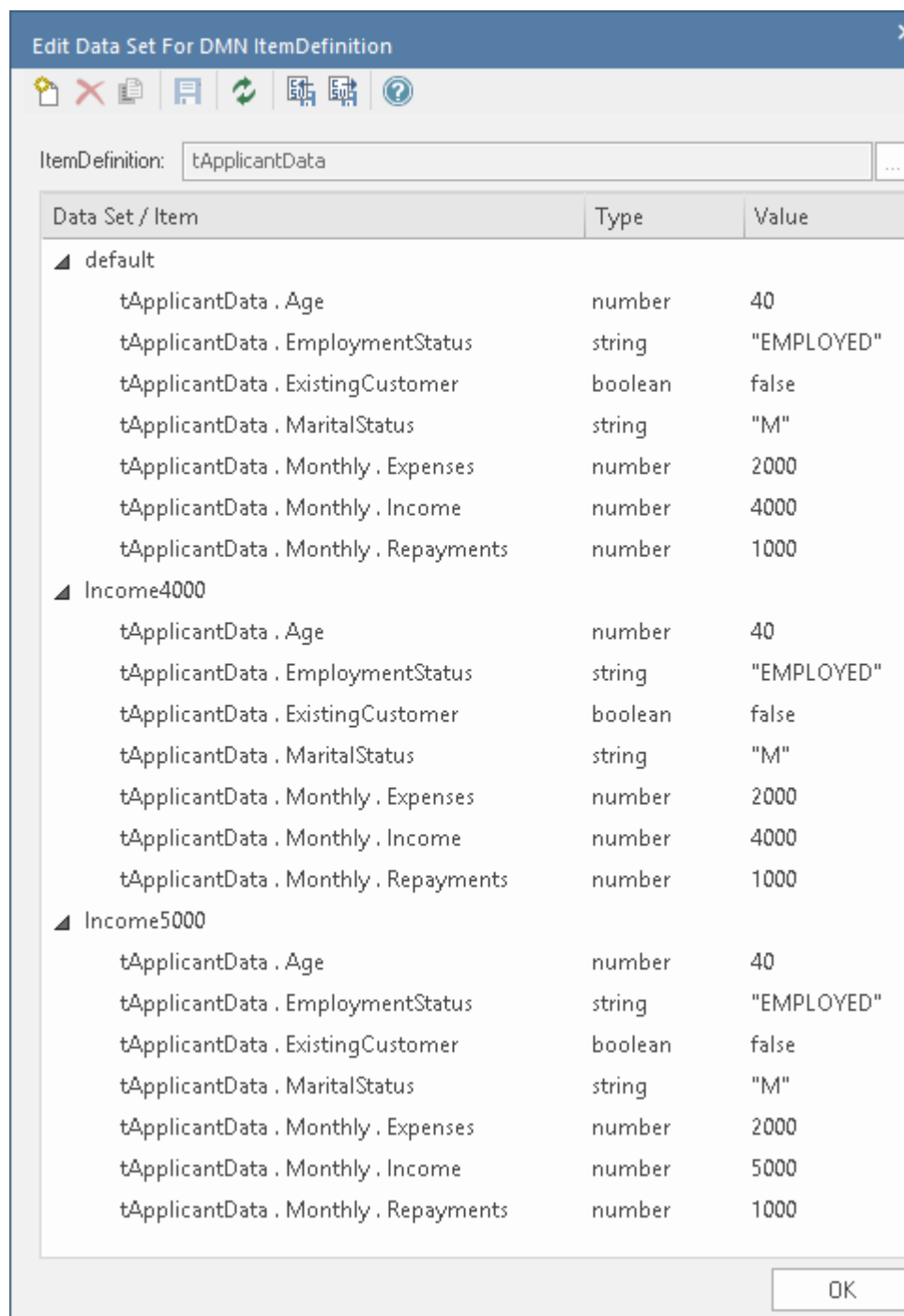
Une ItemDefinition complexe se compose d'éléments imbriqués. Par exemple, *le tApplicantData* est structuré comme suit :



L'exemple du **tApplicantData** ItemDefinition est un type composite de cinq éléments enfants. « Mensuel » est composé de trois enfants (Dépenses, Revenus et Remboursements). Les composants Leaf (non composites), auront un type primitif tel que nombre, string ou Booléen.

Base de données

L'ensemble de données de ItemDefinition peut être affiché et modifié à l'aide de l'icône  dans la barre d'outils. Avec la dialogue « Modifier l'ensemble de données », vous pouvez ajouter, supprimer et dupliquer les ensembles de données. Il existe également support de l'importation et de l'exportation CSV d'ensembles de données.



Comme le montre l'exemple, ItemDefinition pour le *tApplicantData* définit trois ensembles de données :

- défaut
- Revenu4000
- Revenu5000

Chaque ensemble de données peut être visualisé dans un élément InputData saisi dans ItemDefinition. Par exemple, l'élément InputData 'Applicant Data' est saisi dans la ItemDefinition 'tApplicantData'. La fenêtre DMN Expression pour « Données du candidat », illustrée ici, affiche les valeurs des données en fonction de l'ensemble de données sélectionné dans la liste déroulante de la barre d'outils de la fenêtre (*Revenu5000* dans ce cas).

DMN Expression


Income5000

Applicant Data : tApplicantData

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

Définition d'une référence à une ItemDefinition

Un élément DMN InputData est défini pour être référencé (typé) par un ItemDefinition en utilisant soit :

- L'icône  dans la fenêtre Expression DMN de l'élément InputData ou
- En sélectionnant l'élément InputData et en appuyant sur Ctrl+L pour sélectionner la ItemDefinition dans le dialogue

Il existe d'autres cas d'utilisation de ItemDefinitions ; par exemple, lors de la définition du type d'un paramètre d'entrée dans un BKM ou d'un paramètre de sortie dans un Tableau de Décision .

Types de composants








Un élément ItemDefinition peut être défini comme une arborescence de composants composée d'un seul des éléments suivants :

- Un type intégré ou
- Une composition d'éléments ItemDefinition

Dans cette arborescence de composants, si un composant est une « feuille » qui n'a aucun composant enfant, il doit être défini comme type intégré. Si un ItemDefinition a des composants enfants, ce sont ces composants enfants/feuilles qui sont définis comme type intégré.

Par exemple, *les données du candidat* et *le mensuel* sont des compositions, tandis que *l'âge* et *les dépenses* sont des feuilles définies selon un type intégré :

DMN Expression



tApplicantData (ItemDefinition)

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

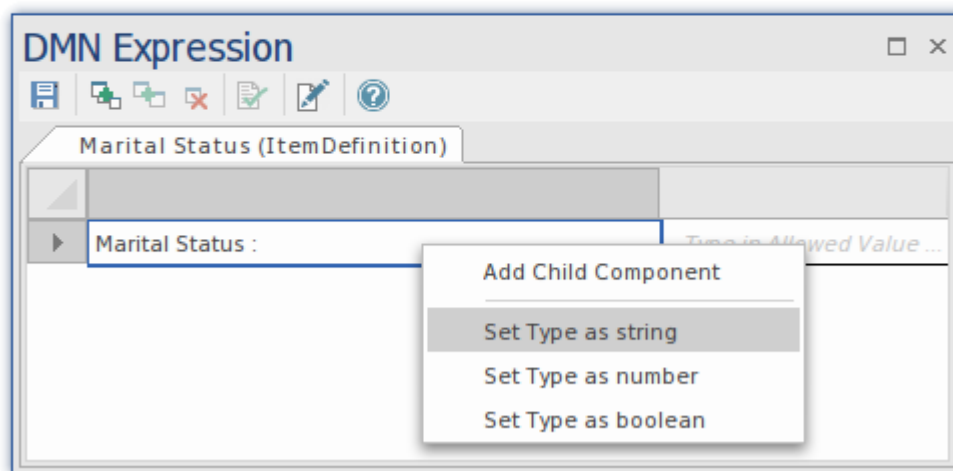
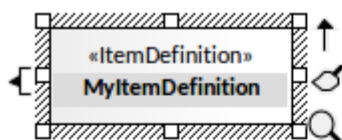
Le langage FEEL a ces types intégrés :

- **nombre**
- **string**
- **booléen**
- jours et durée
- durée en années et en mois
- temps
- date et l'heure

Note : 'number', 'string' et 'boolean' sont pris en charge par Enterprise Architect pour la simulation.

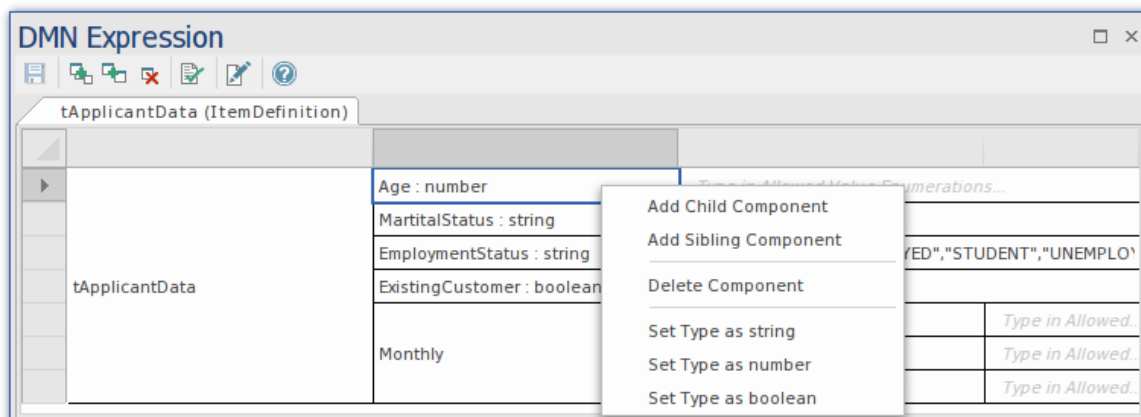
Pour définir un type pour une ItemDefinition « feuille », vous pouvez utiliser l'une des trois méthodes suivantes :

- Sélectionnez l'option de menu contextuel appropriée dans la fenêtre Expression DMN (recommandé)



- Type ' : string ', ' : boolean' ou ' : number' après le nom dans la cellule de la fenêtre DMN Expression
- Type ' string ', 'boolean' ou 'number' comme valeur de la balise ' Type ' dans la fenêtre Propriétés pour ItemDefinition

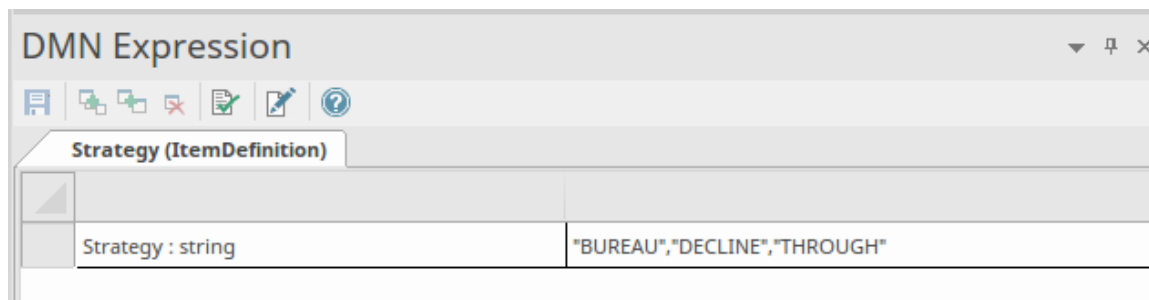
Pour les ItemDefinitions composites, le menu contextuel propose également des options pour créer un composant enfant ou frère, ou pour supprimer l'élément sélectionné :



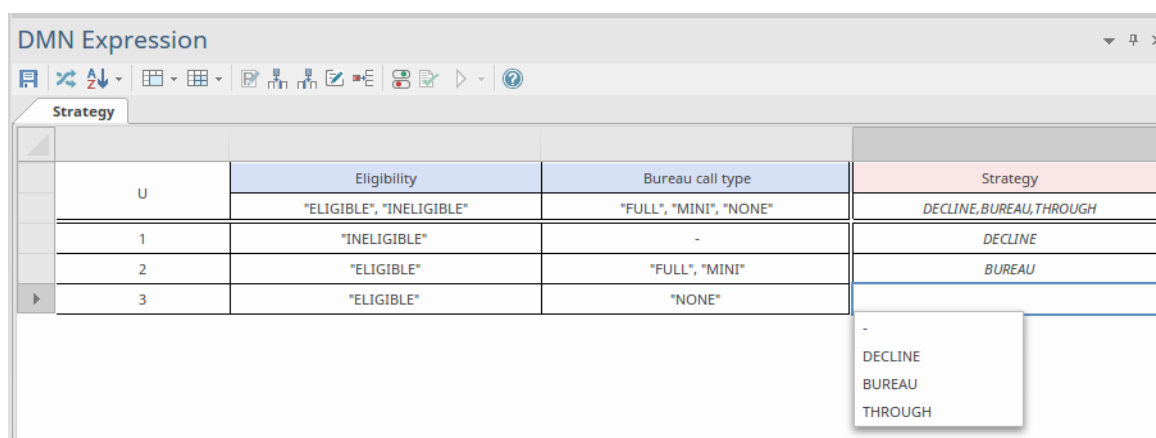
Énumérations de valeurs autorisées

Lors de la définition des entrées de données pour une Décision, il est courant de vouloir restreindre l'ensemble des valeurs autorisées pour une entrée. Par exemple, vous souhaitez peut-être limiter les valeurs autorisées pour l'état civil à seulement deux options, « Célibataire » et « Marié ».

Vous pouvez spécifier les valeurs autorisées pour n'importe quel composant feuille d'un ItemDefinition. Initialement, le champ de données d'un composant feuille contient le texte *Type dans les énumérations de valeurs autorisées*. Vous tapez simplement ce texte avec les valeurs autorisées. Par exemple, la stratégie ItemDefinition a trois valeurs autorisées : BUREAU, DECLINE et THROUGH.



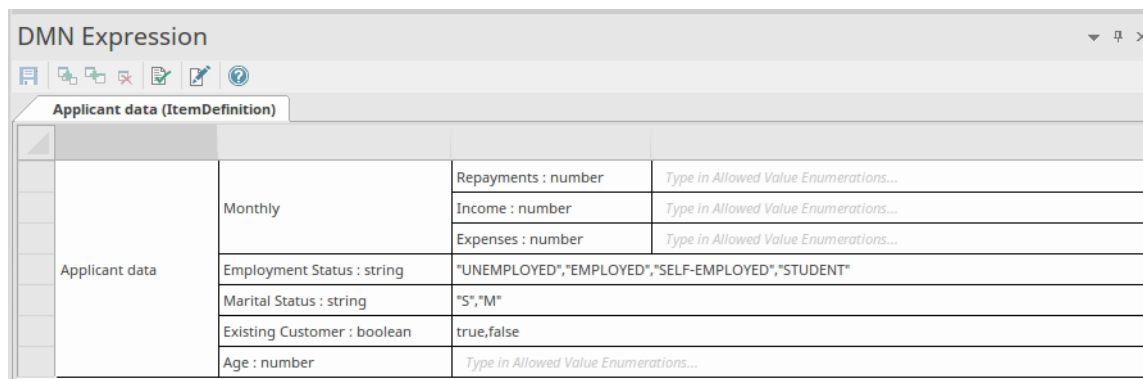
Les énumérations de valeurs autorisées sont également utilisées pour support la saisie semi-automatique. Lors de la spécification de valeurs pour un élément InputData ou un paramètre d'entrée qui fait référence à une ItemDefinition dans laquelle des valeurs autorisées ont été définies, l'utilisateur peut simplement appuyer sur la barre d'espace et choisir une valeur dans la liste.



Vous pouvez également compléter automatiquement en tapant la première lettre de l'option que vous souhaitez saisir.

Les paramètres d'entrée et les clauses de sortie des Tableaux Décision support la spécification des valeurs autorisées. Cela restreint les valeurs qui peuvent être utilisées lors de la définition des règles dans le tableau, mais permet également à l'utilisateur de remplir rapidement les règles en appuyant sur la barre d'espace puis en sélectionnant l'élément requis.

Une ItemDefinition plus complexe peut inclure un certain nombre d'énumérations de valeurs autorisées ; Par exemple:



Ensembles de données

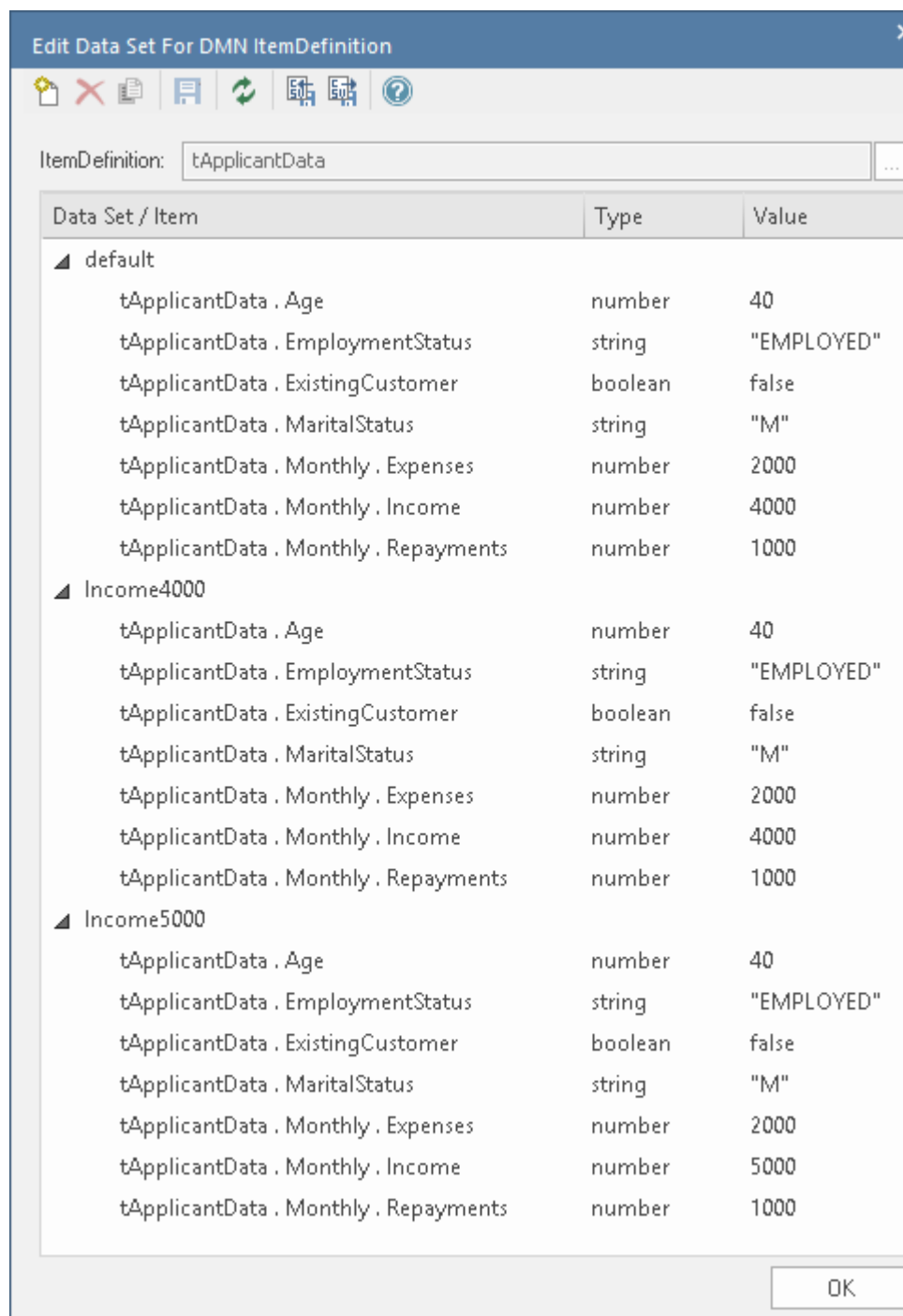
Chaque élément `InputData` typé par un `ItemDefinition` possède un ensemble de composants, et plusieurs ensembles de données peuvent être définis pour fournir différents ensembles de valeurs pour ces composants. Grâce à cette fonctionnalité, un utilisateur effectuant une Simulation DMN peut tester rapidement le résultat d'une décision en choisissant différents ensembles de données. Les ensembles de données sont associés et basés sur `ItemDefinition`, mais vous pouvez également travailler dessus via l'élément `InputData`.

Vous ajoutez ou mettez à jour des ensembles de données à l'aide de la dialogue « Modifier l'ensemble de données », que vous appelez à partir de la fenêtre Expression DMN pour l'élément `ItemDefinition` ou `InputData`. Initialement, la dialogue « Modifier l'ensemble de données » affiche un seul ensemble de composants sans valeurs, sous le nom d'ensemble « par défaut ». Vous pouvez soit laisser cet ensemble sans valeurs, soit fournir des valeurs ; de toute façon, vous pouvez l'utiliser comme gabarit à dupliquer pour de nouveaux ensembles de données. Vous ne pouvez pas supprimer l'ensemble de données « par défaut ».



Lorsque vous accédez à un élément `InputData` dans la fenêtre Expression DMN, les valeurs de l'ensemble de données « par défaut » sont affichées par rapport aux composants de l'élément. Vous pouvez ensuite cliquer sur la flèche déroulante dans la barre d'outils et sélectionner n'importe quel autre ensemble de données dans la liste. Note que si vous laissez l'ensemble de données « par défaut » intact, vous pouvez créer un ensemble de données « par défaut » en double et lui attribuer des valeurs, et cet ensemble « par défaut » fournira les valeurs lorsque vous accéderez initialement à l'élément `InputData`.

Vous pouvez dupliquer et supprimer tout autre ensemble de données que vous créez, exporter les ensembles de données vers un fichier CSV et les importer à partir d'un fichier CSV.








Note que si vous créez un ensemble de données et n'entrez pas de valeurs, il est supprimé lorsque vous fermez le dialogue .



Accéder

Ruban	Simuler > Analyse Décision > DMN > Expression DMN > cliquez sur l'élément InputData : icône 
Autre	Dans un diagramme , double-cliquez sur l'élément DMN InputData : icône  .

Options barre d'outils

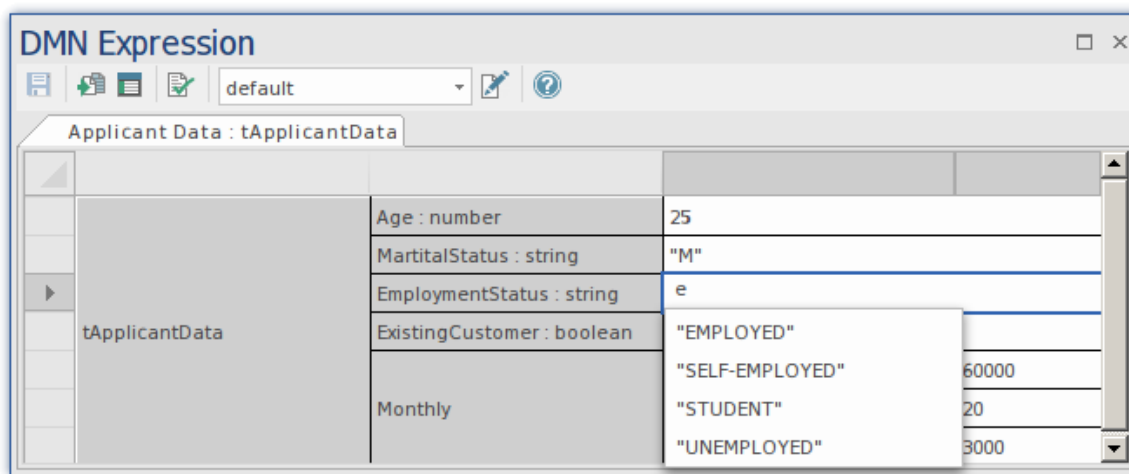
Option	Description
	Cliquez sur ce bouton pour créer un nouvel ensemble de données.
	Cliquez sur ce bouton pour supprimer l'ensemble de données sélectionné.
	Cliquez sur ce bouton pour dupliquer l'ensemble de données sélectionné.
	Cliquez sur ce bouton pour enregistrer les ensembles de données dans InputData.
	Cliquez sur ce bouton pour recharger les ensembles de données pour les InputData.
	Cliquez sur ce bouton pour importer des ensembles de données à partir d'un fichier CSV.
	Cliquez sur ce bouton pour exporter les jeux de données vers un fichier CSV.

Échanger Ensembles de données à l'aide de DataObjects

Lorsque vous testez du code généré à partir d'un modèle DMN ou lors de la simulation de modèles Business Process Model and Notation (BPMN) qui appellent des modèles DMN, vous avez besoin d'un moyen d'échange d'ensembles de données. Par exemple, dans un appel BPMN d'un modèle DMN, un DataObject BPMN est utilisé pour stocker l'ensemble de variables qui seront transmises au modèle DMN qu'il appelle. Ce DataObject doit être rempli avec des données correspondant à la structure de données du DMN InputData, prêtes à être transmises à cet object InputData. Ce même BPMN DataObject est utilisé lors du test du code généré à partir d'un modèle DMN.

Cette rubrique décrit le processus de création d'objets de données BPMN à partir Ensembles de données DMN.

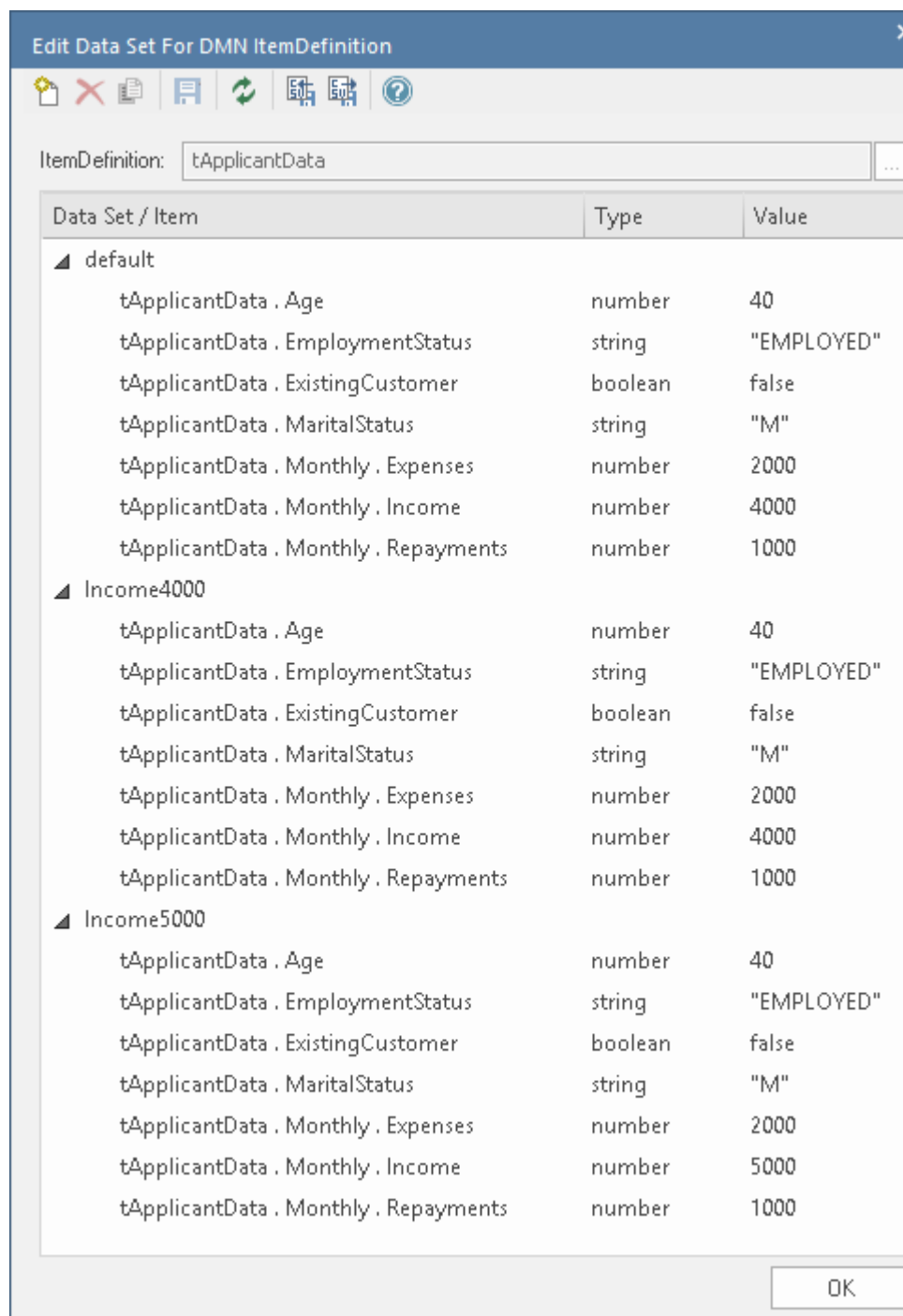
Un ensemble de données est stocké dans un élément DMN InputData et est accessible à l'aide de l'icône  dans la fenêtre Expression DMN.



The image shows a screenshot of the 'DMN Expression' window. The title bar says 'DMN Expression'. Below the title bar is a toolbar with icons for file operations and a dropdown menu set to 'default'. The main area is titled 'Applicant Data : tApplicantData'. It contains a table with the following data:

Variable	Type	Value
Age	number	25
MartitalStatus	string	"M"
EmploymentStatus	string	e
ExistingCustomer	boolean	"EMPLOYED"
Monthly		"SELF-EMPLOYED" 60000
		"STUDENT" 20
		"UNEMPLOYED" 3000

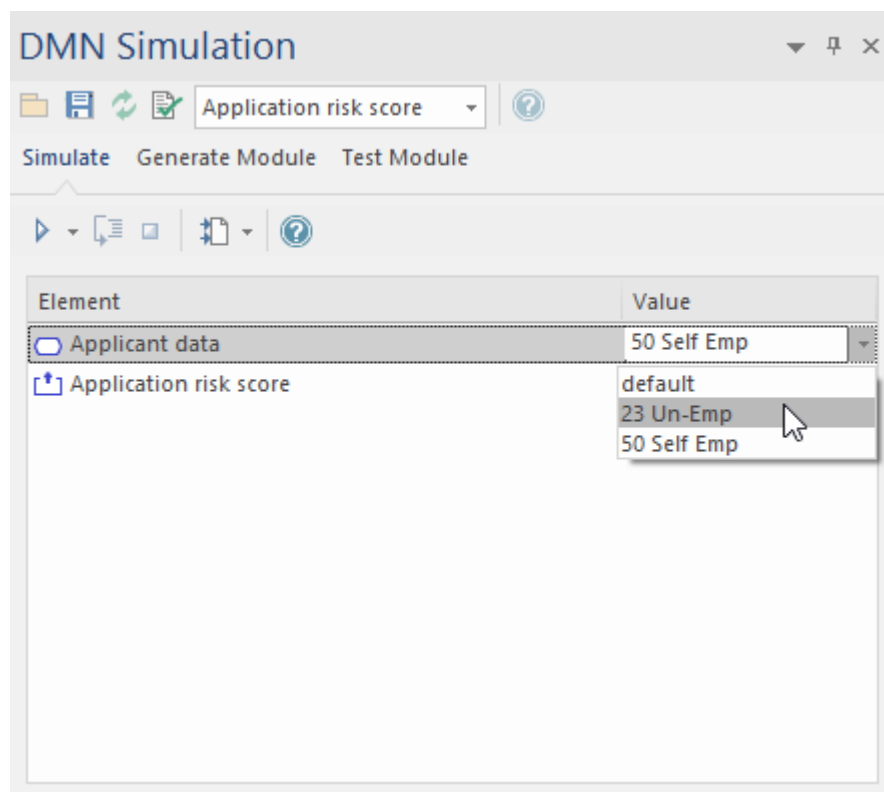
Cela ouvre la dialogue « Modifier l'ensemble de données » d'InputData, qui peut contenir plusieurs ensembles de valeurs :




Il existe deux options pour transférer l'ensemble de données vers un DataObject :

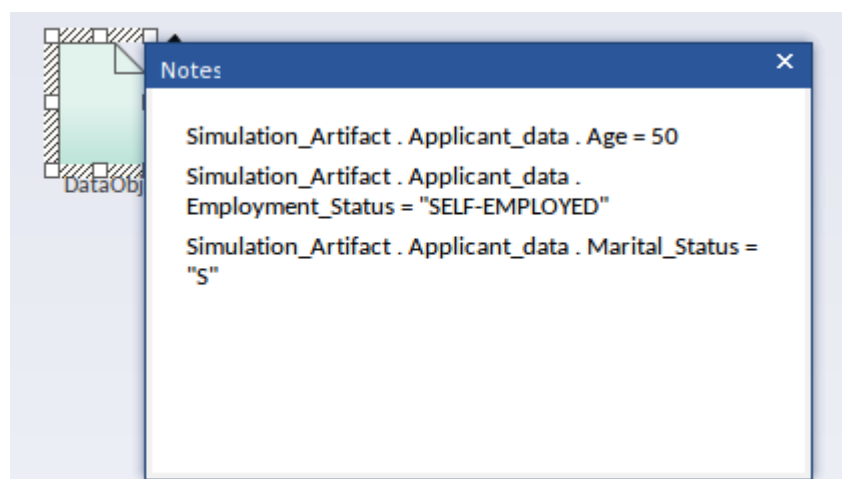
1. Directement

- Créez un DataObject BPMN sous un Paquetage dans la fenêtre Navigateur .
- Ouvrez la fenêtre Simulation DMN





- Sélectionnez un ensemble de données dans la liste déroulante « Valeur »
- Cliquez sur l'icône  dans la fenêtre DMN Simulation ; cela ouvre la dialogue « Sélectionner un élément »
- Sélectionnez l'élément BPMN DataObject
- Cliquez sur le bouton OK

Le Data Set est maintenant visible dans les Notes du DataObject.

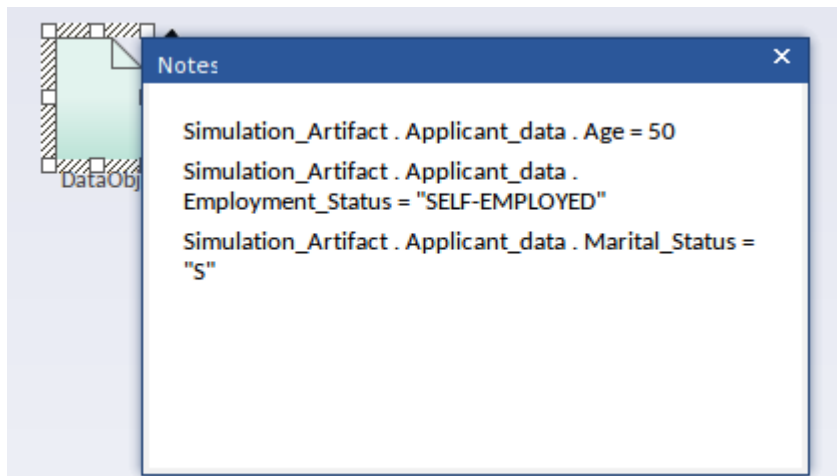


2. Manuel

Pour échanger manuellement cet ensemble de données :

- Ouvrez la fenêtre DMN Expression pour l'élément InputData
- Cliquez sur l'icône Modifier le DataSet  ; cela ouvre la dialogue « Modifier l'ensemble de données »
- Utilisez l'icône  d'exportation CSV pour exporter ces détails vers un fichier

Le texte du fichier CSV peut être ajouté sous forme de texte dans les Notes d'un élément BPMN DataObject.



Service Décision

Des parties de ce sujet ont été utilisées textuellement ou sont librement adaptées du DMN Specification, disponible sur : <https://www.omg.org/spec/DMN>. Ce site contient une description complète du DMN et de ses capacités.

Un service Décision expose une ou plusieurs décisions d'un modèle Décision en tant qu'élément réutilisable, qui peut être invoqué en interne par une autre décision du modèle Décision, ou en externe par une tâche dans un modèle de processus BPMN.

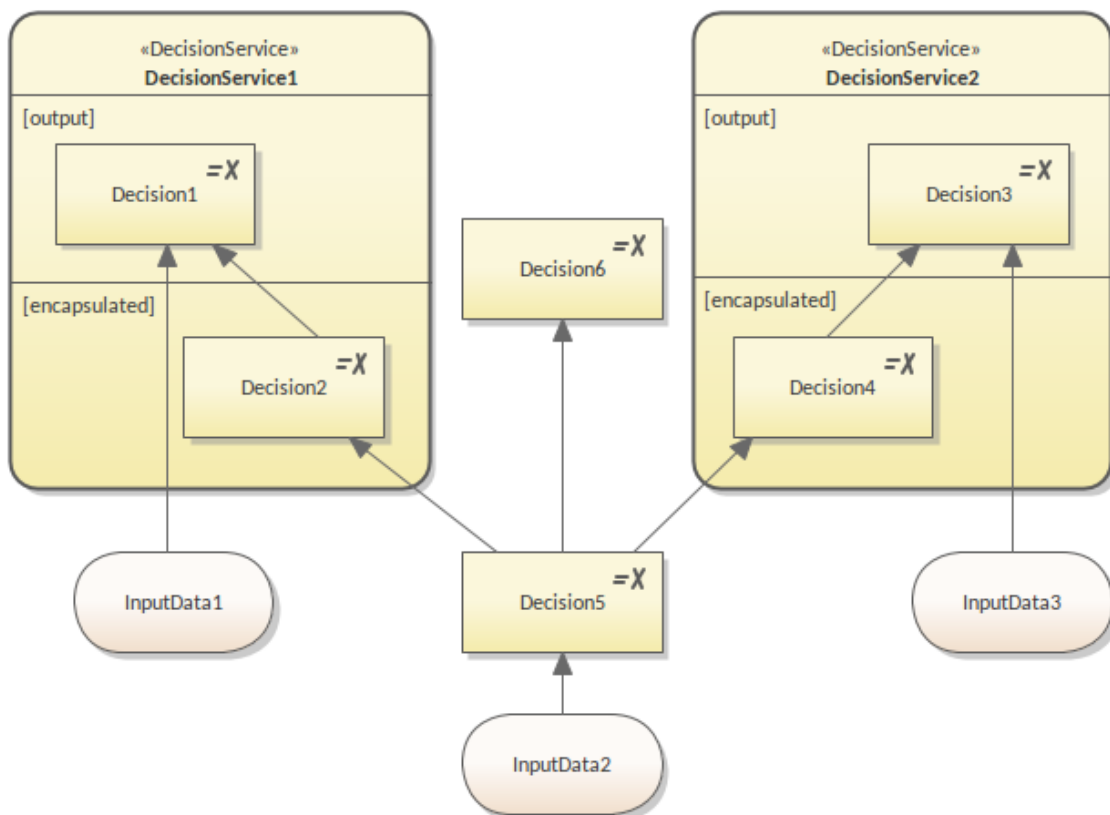
Lorsque le service Décision est appelé avec les données d'entrée et les décisions d'entrée nécessaires, il renvoie les sorties des décisions exposées.

L'interface d'un service Décision

L'interface avec le Service Décision se compose de :

- Données d'entrée - instances de toutes les données d'entrée requises par les décisions encapsulées
- Décisions d'entrée - instances des résultats de toutes les décisions d'entrée
- Décisions de sortie - les résultats de l'évaluation (au moins) de toutes les décisions de sortie, en utilisant les décisions d'entrée et les données d'entrée fournies

Lorsque le service Décision est appelé avec les données d'entrée et les décisions d'entrée nécessaires, il renvoie les sorties des décisions exposées.



Cette figure montre un modèle Décision qui comprend six décisions et trois éléments de données d'entrée.

Pour DecisionService1, :

- La décision de sortie est {Décision1}
- La décision d'entrée est {Décision5}, et
- Les données d'entrée sont {InputData1}

Comme la Décision1 requiert la Décision2, qui n'est pas fournie au service en entrée, le service doit également encapsuler la Décision2 ; par conséquent, les décisions encapsulées sont {Decision1, Decision2}.

Il ressort clairement de la figure que Decision6, Decision3, Decision4 et InputData3 ne sont requis par aucune décision de DecisionService1. Qu'en est-il d'InputData2 ? Bien qu'il soit requis par Decision5, qui est requis par DecisionService1, InputData2 n'est en réalité pas requis par DecisionService1. En effet, Decision5 est défini comme Input Décision . Du point de vue d'un service Décision , nous ignorons toutes les décisions ou données d'entrée requises par une Décision d'entrée.

Pour DecisionService2, :

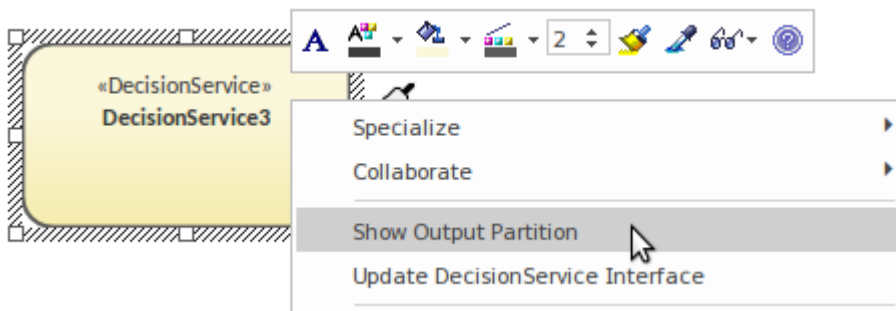
- La décision de sortie est {Décision3}
- La décision d'entrée est {Décision5}, et
- Les données d'entrée sont {InputData3}

Comme la Décision3 nécessite la Décision4, qui n'est pas fournie au service en entrée, le service doit également encapsuler la Décision4 ; par conséquent, les décisions encapsulées sont {Decision3, Decision4}.

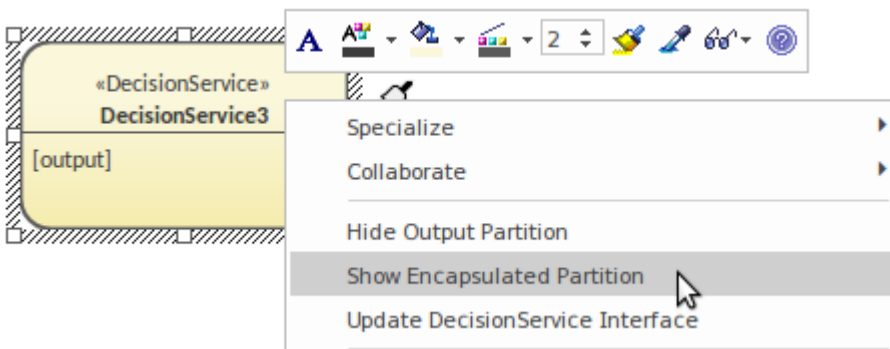
Il est recommandé de créer un diagramme distinct pour chaque service Décision . De cette manière, le diagramme ne contiendra que les éléments d'interface et les décisions encapsulées pour le Service Décision ; les éléments non pertinents n'apparaîtront pas sur le diagramme .

Modélisation d'un service Décision

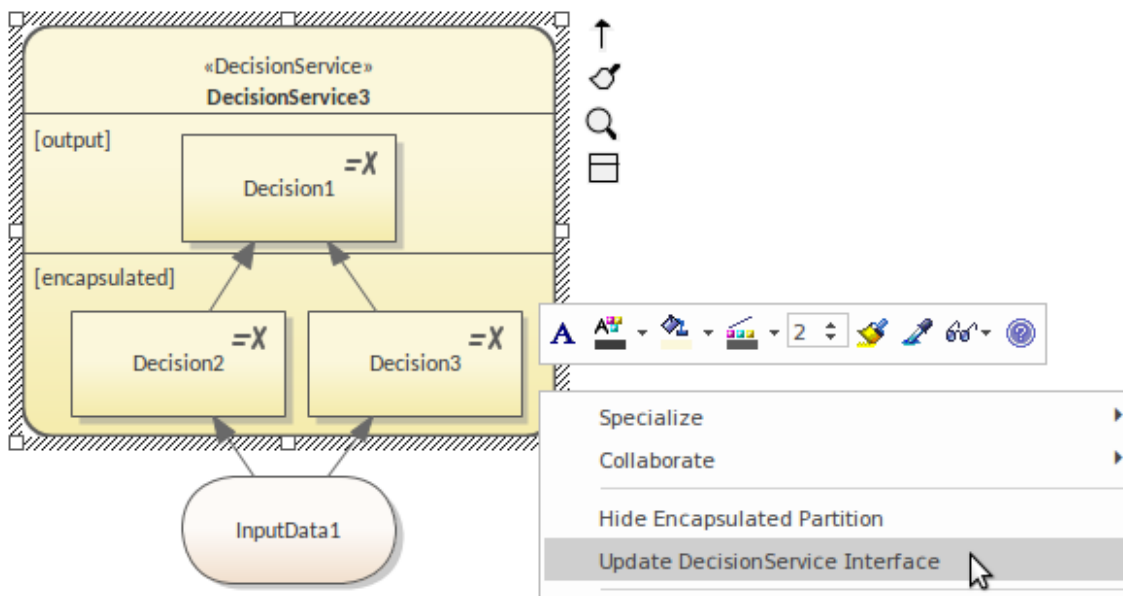
Nous pouvons créer un élément Décision Service à partir des pages DMN de Diagramme Toolbox et basculer les partitions [output] et [encapsulated] depuis le menu contextuel.



Vous ne pouvez afficher une partition [encapsulée] que lorsqu'une partition [de sortie] est affichée.



Une fois les décisions et les données d'entrée placées dans la ou les partitions appropriées, vous devez exécuter la commande 'Update DecisionService Interface' depuis le menu contextuel pour mettre à jour le modèle.



Important : pour que la simulation DMN fonctionne correctement, veuillez mettre à jour l'interface Décision Service chaque fois que vous :

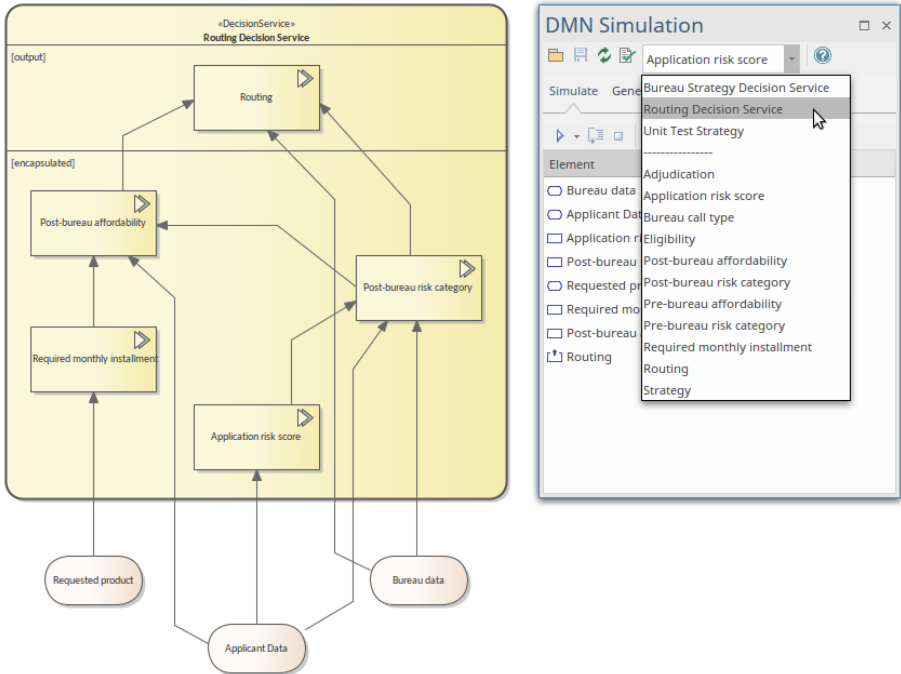
- Afficher/Masquer la ou les partitions du service de décision
- Ajouter une décision au service de décision
- Supprimer une décision du service de décision
- Déplacer une décision entre les partitions
- Ajouter/Supprimer des entrées du service Décision : données d'entrée ou décisions d'entrée


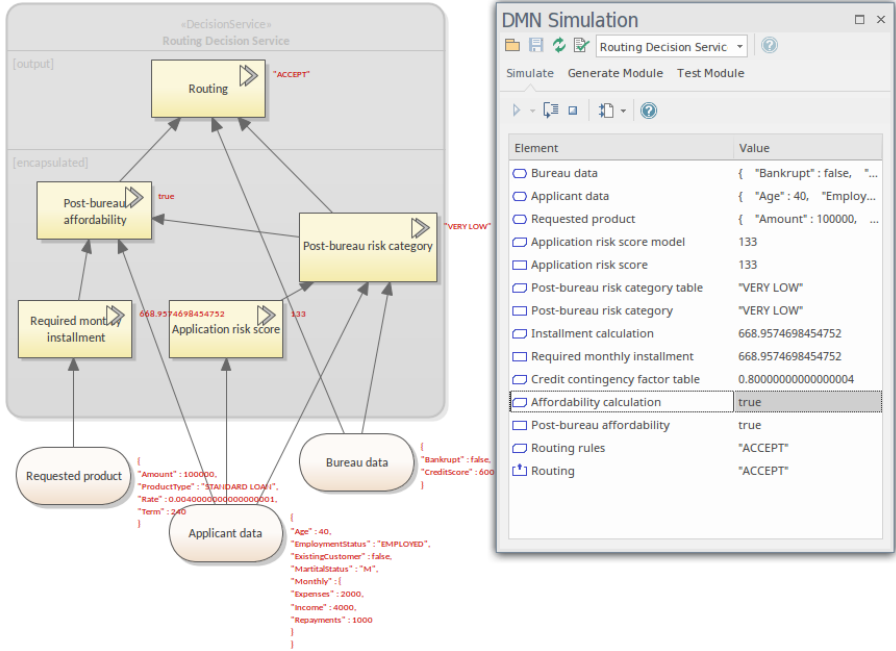
Simuler un service Décision

Il est possible de réaliser une simulation de modèle sur un Service Décision .

Simulation de service Décision

Pour effectuer une simulation de modèle sur le service Décision , procédez comme suit :

Étape	Description
1	<p>Faites glisser un élément d'artefact de configuration Simulation sur un diagramme depuis la page « Composants DMN » de la boîte à outils et double-cliquez dessus pour l'ouvrir dans la fenêtre Simulation DMN.</p>  <p>The diagram shows a Decision Service named 'Routing Decision Service'. It has an output decision 'Routing' and several encapsulated decisions: 'Post-bureau affordability', 'Post-bureau risk category', 'Required monthly installment', and 'Application risk score'. Inputs include 'Requested product', 'Applicant Data', and 'Bureau data'. To the right, the 'DMN Simulation' window is shown with a list of elements for simulation, including 'Routing Decision Service' and various data elements like 'Application risk score' and 'Post-bureau risk category'.</p> <p>Par défaut, tous les éléments Décision Service et chaque décision sont répertoriés pour sélection dans le champ déroulant de le dialogue .</p>
2	<p>Sélectionnez un élément Décision Service sur lequel exécuter la simulation. Dans l'exemple, nous avons choisi 'Routing Décision Service', donc trois éléments de données d'entrée et cinq décisions encapsulées (dont une décision de sortie) sont chargés dans la liste de simulation.</p> <p>Important : Cette liste est issue des données internes du Service Décision ; assurez-vous exécuter la commande 'Update DecisionService Interface' depuis le menu contextuel chaque fois que le diagramme du modèle Décision Service est modifié. Rechargez le Décision Modèle en cliquant sur l'icône 'Rafraîchir' (troisième en partant de la gauche) dans la barre d'outils de la fenêtre DMN Simulation .</p>
3	<p>Les données d'entrée et les décisions sont dans le bon ordre d'exécution. Par exemple, « Score de risque d'application » sera exécuté avant « Catégorie de risque du bureau de poste », « Accessibilité du bureau de poste » et « Routage ». Pour chaque élément de données d'entrée, cliquez sur la flèche déroulante dans le champ « Valeur » et sélectionnez les Ensembles de données pour fournir les valeurs des données d'entrée.</p>

	<p>Validez les données d'entrée et les décisions, et apportez les corrections nécessaires à l'aide de la fenêtre DMN Expression.</p> <p>Dans la fenêtre DMN Simulation , cliquez sur l'icône Enregistrer et sur le bouton  de la barre d'outils.</p>
4	<p>Le résultat de l'exécution du runtime est affiché à la fois dans la liste et sur le diagramme . Vous pouvez également cliquer sur l'icône « Pas à pas » dans la barre d'outils pour déboguer le modèle DMN.</p>  <p>Une bonne pratique consiste à garder la fenêtre DMN Expression ouverte pendant le débogage. Le statut temporel exécuter de l'expression (tel que Tableau de Décision , Boxed Context, Literal Expression ou Invocation) affichera les détails de la logique encapsulée par la Décision ou invoquée Métier Knowledge Modèle .</p>

Démarrage

Decision Model and Notation (DMN) est une norme publiée et gérée par l' Object Management Group (OMG).

Des parties de ce sujet ont été utilisées textuellement ou sont librement adaptées du DMN Spécification , disponible sur la page Web OMG DMN (<https://www.omg.org/spec/DMN>). Une description complète du DMN et de ses capacités est disponible sur le site Web de l'OMG.

Le but de DMN est de fournir les constructions nécessaires pour modéliser les décisions, afin que la prise de décision organisationnelle puisse être facilement représentée dans diagrammes , définis avec précision par analystes métier et (éventuellement) automatisée. Il vise également à faciliter le partage et l'échange de modèles Décision entre les organisations.

Sélection de la perspective

Enterprise Architect partitionne les fonctionnalités étendues de l'outil dans Perspectives , ce qui garantit que vous pouvez vous concentrer sur une tâche spécifique et travailler avec les outils dont vous avez besoin sans être distrait par d'autres fonctionnalités . Pour travailler avec les fonctionnalités Decision Model and Notation vous devez d'abord sélectionner cette perspective :



<nom de la perspective> > Exigences > Décision Modélisation

La définition de la perspective garantit que Decision Model and Notation diagrammes de notation, leurs boîtes à outils et autres fonctionnalités de la perspective seront disponibles par défaut.

Exemple Diagramme

Un exemple diagramme fournit une introduction visuelle au sujet et vous permet de voir certains des éléments et connecteurs importants créés en spécifiant ou en décrivant la manière dont les décisions sont modélisées. Le diagramme Décision Exigences présentera des éléments tels que Tableaux Décision , les sources de connaissances, la date d'entrée et plus encore. Une grande partie de la puissance d' Enterprise Architect repose sur sa capacité à simuler ou à « exécuter » les modèles de décision et à prédire les résultats en fonction de différents ensembles de données. Cette fonctionnalité sera décrite dans des rubriques ultérieures mais commence par la création d'un diagramme Exigences Décision .

Modélisation avec DMN

Cette rubrique vous présente les éléments les plus importants dont vous avez besoin pour créer des modèles Décision . Cela comprend la création d'un diagramme Exigences Décision qui décrit comment les décisions sont liées et quelles entrées chaque décision a potentiellement, y compris d'autres décisions. Vous découvrirez les éléments les plus importants, notamment : Les décisions. Modèles de connaissances Métier , définitions Items de données d'entrée, ensembles de données et services Décision .

Module de génération de code et Test

Cette rubrique vous présente les principaux concepts du langage, notamment sa structure, son architecture ainsi que les éléments et connecteurs utilisés pour créer des modèles Decision Model and Notation (DMN) . Comprendre l'intention et la structure du langage aidera les analystes à créer des modèles de décision significatifs et productifs.

Intégrer dans BPSim pour Simulation

Enterprise Architect permet d'exécuter (simuler) des modèles Décision, ce qui vous permet de visualiser les résultats des décisions. En plus de ce noyau facilité vous pouvez également intégrer les modèles Décision avec BPSim qui est un moteur de simulation pour la simulation de diagrammes BPMN. Dans cette rubrique, vous découvrirez différentes manières d'intégrer DMN à BPSim.

Intégrer dans un élément de classe UML

Dans cette rubrique, vous apprendrez le processus d'intégration d'un Modèle DMN avec un élément de classe UML. Le module DMN peut être intégré à un élément de classe UML, de sorte que le code généré à partir de cet élément de classe peut réutiliser le module DMN et être bien structuré.

Importation de XML DMN

Cette rubrique décrit comment importer un fichier XML DMN à partir d'un autre référentiel Enterprise Architect ou d'un autre outil compatible DMN. L'une des promesses des standards ouverts est la possibilité de partager des modèles entre différents outils. Enterprise Architect devient souvent l'outil de choix pour modélisation en raison de l'étendue des fonctionnalités et des normes qu'il supporte. Cela permet au modèle Décision d'être lié aux éléments de stratégie, Exigences, Processus Métier, de mise en œuvre logicielle et plus encore.

Plus d'information

Cette section fournit des liens utiles vers d'autres sujets et ressources qui pourraient vous être utiles lorsque vous travaillez avec les fonctionnalités des outils Decision Model and Notation.

Plus d'information

Créer un Modèle Décision

Dans le modèle que nous avons décrit dans *Un exemple de Modélisation Décision*, nous avons montré comment une décision peut être modélisée à l'aide d'un Tableau de Décision, dans lequel un résultat de décision est déterminé en trouvant une ligne dans le tableau où les valeurs d'entrée dans le tableau correspondent à l'entrée. valeurs considérées, donnant un résultat de sortie particulier.

Nous allons maintenant voir comment un tel modèle peut être créé dans Enterprise Architect, en suivant le processus de création du modèle de décision pour l'exemple de mise à niveau de la cabine d'une compagnie aérienne.

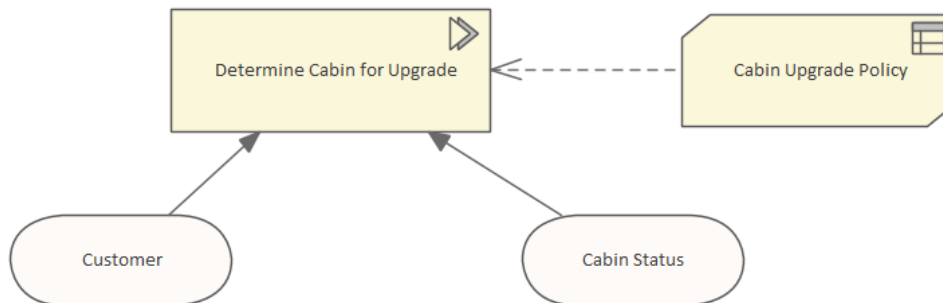
Un certain nombre d'éléments de modèle sont impliqués dans cet exemple, tels que des éléments de données d'entrée, Item définitions d'éléments qui sont utilisés pour décrire les données d'entrée (définissant les types de données), un élément Décision et également un élément Métier Knowledge Modèle qui contient le Tableau de Décision Définition Tableau de Décision.

Créer un Diagramme Exigences Décision

Ces étapes vous guideront tout au long de la création d'un Diagramme Exigences Décision (DRD) simple. Dans cet exemple, nous allons créer le modèle de toutes pièces, plutôt que d'utiliser un motif de l'Assistant de Modèle.

Étape	Description
1	Sélectionnez la perspective 'Exigences Décision Modélisation'. (L'onglet Page Démarrage 'Créer à partir de Motif' (Assistant de Modèle) s'affiche, mais nous ne l'utiliserons pas pour cet exemple.)
2	Créez un nouveau diagramme DMN. Nommez-le « Surclassement de la cabine de la compagnie aérienne ».
3	A l'aide de la boîte à outils diagramme, placez un élément Décision sur le diagramme. Choisissez 'Invocation' comme type - nous utiliserons cet élément pour 'invoquer' une décision à partir d'un élément Métier Knowledge Modèle. Nommez l'élément « Déterminer la cabine à mettre à niveau ».
4	Placez un élément InputData sur le diagramme. Nommez cet élément « Client ».
5	Placez un autre élément InputData sur le diagramme. Nommez cet élément « Statut de la cabine ».
6	Placer un élément Métier Knowledge Modèle sur le diagramme. Choisissez le type 'Tableau de Décision'. Nommez cet élément « Politique de mise à niveau de la cabine ».
7	Dessinez un connecteur « Informations requises » à partir de la décision « Déterminer la cabine pour la mise à niveau » vers les données d'entrée « Client ».
8	Dessinez un connecteur « Informations requises » à partir de la décision « Déterminer la cabine pour la mise à niveau » vers les données d'entrée « État de la cabine ».
9	Dessinez un connecteur « Exigences de connaissances » entre la décision « Déterminer la cabine à mettre à niveau » et la « Politique de mise à niveau de la cabine » du BKM.

A ce stade, nous devrions avoir un simple DRD, qui ressemble à ceci :



Nous pouvons maintenant préciser les détails de chacun des éléments composant ce modèle.

Définir le Tableau de Décision

En double-cliquant sur l'élément Métier Knowledge Modèle 'Cabin Upgrade Policy', la fenêtre 'DMN Expression' s'affiche, affichant un Tableau de Décision vide. C'est ici que nous définirons les règles de notre politique de surclassement en cabine.


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

Par défaut, les nouveaux Tableaux Décision sont créés avec deux colonnes d'entrée et une colonne de sortie, une ligne d'en-tête et trois lignes de règles vides.

La colonne la plus à gauche du tableau affiche la politique Hit et numérote également les règles. Par défaut, la stratégie Hit est « U » pour « Unique ». C'est la politique que nous utiliserons pour notre exemple, vous n'avez donc pas besoin de modifier cet en-tête de colonne.

Pour plus d'informations sur les stratégies Hit , reportez-vous à la rubrique d'aide sur les *stratégies Hit Tableau de Décision* .

Nommer et définir les types pour les entrées et sorties Tableau de Décision

Étape	Description
1	Dans la barre d'outils de la fenêtre 'DMN Expression', cliquez sur le bouton 'Modifier les paramètres',  . La dialogue « Modifier les paramètres » s'affiche.
2	Remplacez le nom du paramètre « Entrée 1 » par « Nombre de personnes surréservées ».

	Si nécessaire, cliquez sur la flèche déroulante ' Type ' et définissez le type de ce paramètre sur ' nombre '.
3	Remplacez le nom du paramètre « Entrée 2 » par « Nombre de vols au cours du mois dernier par passe ». Définissez également le type de ce paramètre sur « numéro ». Fermez la dialogue « Modifier les paramètres ».
4	Modifiez l'expression d'entrée qui sera évaluée pour la colonne 1. Sélectionnez la cellule d'en-tête (contenant le texte « Entrée 1 ») puis cliquez à nouveau ou appuyez sur F2 pour passer en mode « Édition ». Sélectionnez tout le texte de la cellule, puis appuyez sur la barre d'espace. La liste des paramètres d'entrée s'affiche. Cliquez sur « Nombre de personnes surréservées », puis appuyez sur « Entrée ». L' <i>expression</i> de la colonne 1 est définie sur « Nombre de personnes surréservées ». Note : Les expressions d'entrée évaluées pour chaque colonne utilisent généralement simplement le paramètre d'entrée correspondant ; cependant, vous <i>pouvez</i> utiliser une expression complexe.
5	Cliquez-droit sur l'expression de la colonne 1 et vérifiez que son type de données est défini sur 'number'.
6	Modifiez l'expression d'entrée qui sera évaluée pour la colonne 2. Sélectionnez tout le texte, puis appuyez sur la barre d'espace. La liste des paramètres d'entrée s'affiche. Choisissez « Nombre de vols au cours du mois dernier pour le Pass », puis appuyez sur « Entrée ». L' <i>expression</i> de la colonne 2 est définie sur « Nombre de vols au cours du mois dernier pour le Pass ».
7	Cliquez-droit sur l'expression de la colonne 2 et définissez son type de données sur 'number'.
8	Modifiez le nom de la sortie du tableau de décision. Remplacez « Sortie 1 » par « Mettre à niveau la cabine », puis appuyez sur « Entrée ».
9	Définissez le type de données de la sortie de décision. Cliquez-droit sur l'en-tête de la colonne de sortie et choisissez ' string '.
dix	Définissez les valeurs autorisées pour la sortie de décision. Dans la cellule située directement sous l'en-tête de la colonne de sortie (mais au-dessus de la ligne 1), définissez les valeurs autorisées pour la sortie. Entrez « Classe Métier , Première Classe ». Note : Il n'est pas nécessaire de mettre des guillemets autour des valeurs, car le type de données a été spécifié comme ' string '.


Définir les règles du Tableau de Décision


Entrez les valeurs dans les cellules tableau pour correspondre à cette image.

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

Cliquez sur une cellule pour la sélectionner, puis cliquez à nouveau pour la modifier.

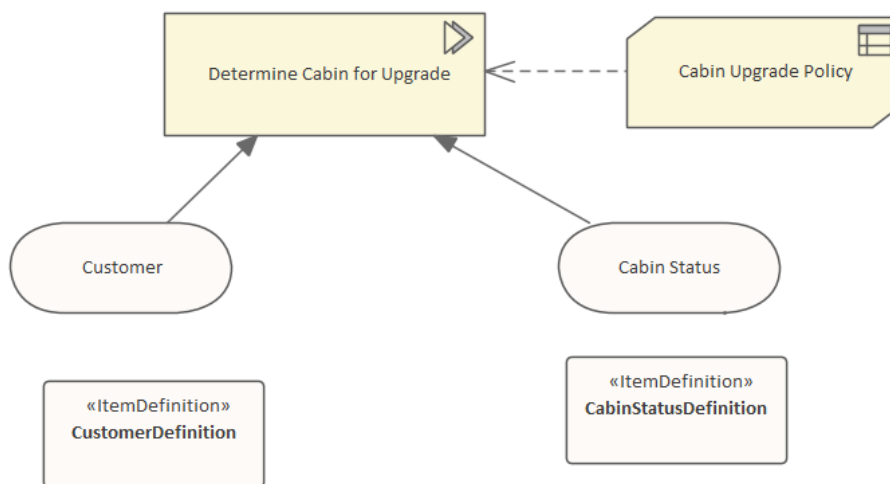
Vous pouvez copier et coller des règles existantes en sélectionnant les lignes à copier (Maj+clic ajoute à la sélection), cliquez-droit et choisissez 'Copier', puis cliquez-droit et choisissez 'Ajouter'.

Une fois que vous avez fini de modifier les règles, cliquez sur le bouton Enregistrer .

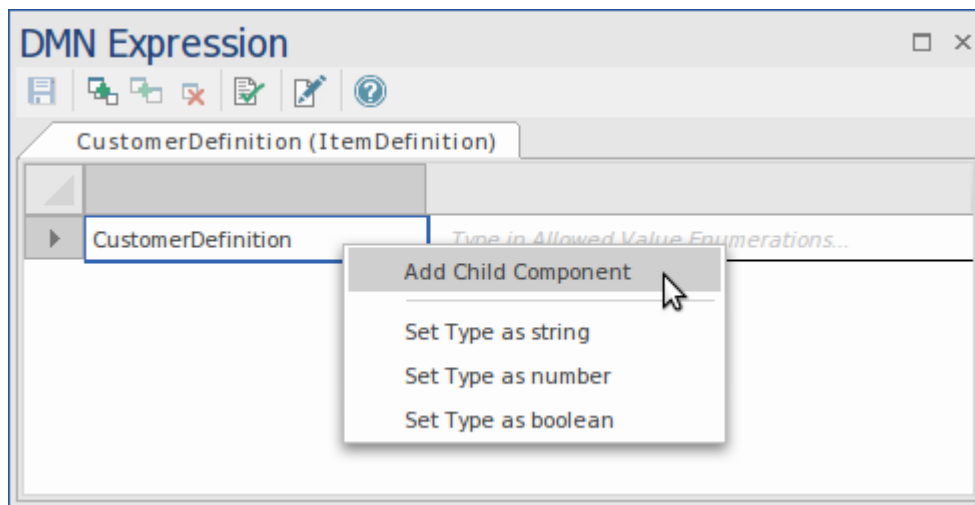
Enfin, cliquez sur le bouton Valider , pour vérifier les erreurs dans le tableau des règles.

Créer des éléments ItemDefinition

Ajoutez deux éléments ItemDefinition au diagramme , un pour chacun des éléments InputData. Nommez un élément « CustomerDefinition » et l'autre « CabinStatusDefinition ».




Double-cliquez sur la ItemDefinition nommée « CustomerDefinition » pour modifier la définition. La fenêtre Expression DMN s'affiche.

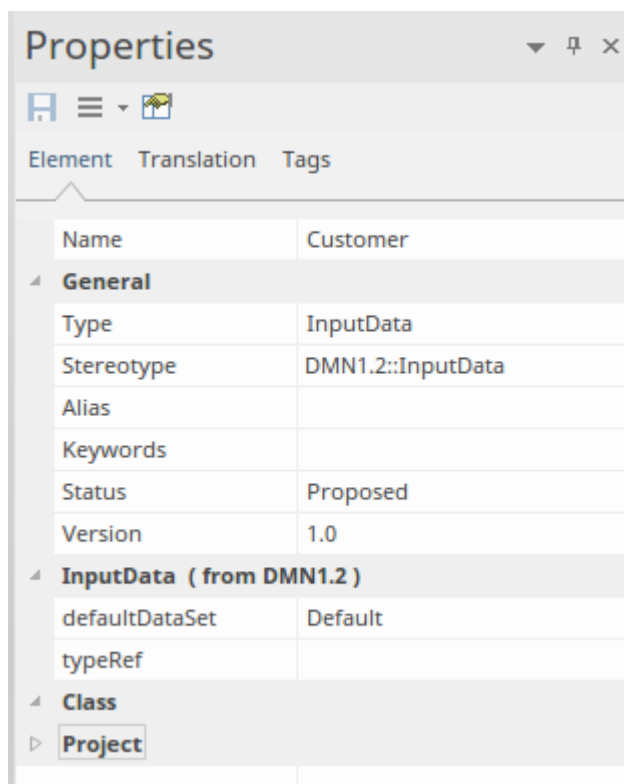


Cliquez-droit sur la cellule 'CustomerDefinition' et choisissez 'Ajouter un composant enfant'. Remplacez le nom du composant enfant par « Nombre de vols au cours du mois dernier » et remplacez son type de données par « numéro ». Cliquez sur le bouton « Enregistrer » pour enregistrer les modifications et fermez la fenêtre.

De même, double-cliquez sur la ItemDefinition nommée « CabinStatusDefinition », ajoutez un composant enfant nommé « Num of Pax Overbooked » et définissez son type de données sur « number ». Enregistrez les modifications et fermez la fenêtre.

Spécifier le Type de données pour chaque élément InputData

Sélectionnez l'élément InputData 'Client'. Dans la fenêtre Propriétés , sélectionnez la propriété 'typeRef' et cliquez sur le bouton .



Sélectionnez ItemDefinition « Définition client » comme type. Cliquez sur 'OK '.

De même, spécifiez « Définition du statut de la cabine » comme type de « Statut de la cabine ».

Spécifier les entrées de l'élément Décision

Double-cliquez sur l'élément de décision « Déterminer la cabine à mettre à niveau »

Dans la fenêtre DMN Expression, localisez la ligne tableau contenant le texte « Num of Pax Overbooked » dans la première colonne. Cliquez dans la cellule de la deuxième colonne de cette ligne et appuyez sur la barre d'espace. Une liste de valeurs d'entrée possibles s'affiche. Choisissez « Statut de la cabine ». Num of Pax Overbooked' et appuyez sur 'Entrée'. La sélection est écrite dans la cellule.

Répétez ce processus pour la deuxième ligne tableau « Nombre de vols au cours du mois dernier », en choisissant « Client ». Nombre de vols au cours du mois dernier.

Cliquez sur le bouton Enregistrer.



Cliquez sur le bouton Valider.

Définir Ensembles de données

L'« exactitude » de votre modèle de décision peut être testée en exécutant des simulations à l'aide d'une gamme d'ensembles de données représentatifs pour vérifier que le modèle produit le résultat correct dans toutes les situations.

Vous pouvez créer de nombreux Ensembles de données portant différents noms, en utilisant une plage de valeurs de données. Vous pouvez définir l'un des ensembles de données comme *valeur par défaut*.

Nous allons maintenant créer un ensemble de données pour chacun de nos éléments InputData.


Étape	Description
1	Double-cliquez sur l'élément InputData 'Client'. La fenêtre Expression DMN s'affiche.
2	Dans la fenêtre DMN Expression, cliquez sur le bouton « Modifier l'ensemble de données »  . La fenêtre « Modifier l'ensemble de données » s'affiche.
3	Cliquez sur le bouton  . Un nouvel ensemble de données est créé.
4	Écrasez le nom de l'ensemble de données si vous le souhaitez. Laissez le Type sur « numéro ». Entrez une valeur de, par exemple, 3. Cliquez sur l'icône Enregistrer et sur le bouton OK.
5	Répétez l'opération pour le « Statut de la cabine » des données d'entrée. Entrez une valeur de, par exemple, 4.

Ajouter un artefact DMNSimConfiguration

Localisez l'artefact DMN « Configuration Simulation » dans la boîte à outils Diagramme. Déposez-en également un sur le diagramme.

Double-cliquez dessus pour ouvrir la fenêtre DMN Simulation dans l'onglet « Simuler ».

Depuis la fenêtre DMN Simulation, vous pouvez exécuter des simulations du Décision Modèle complété. Vous pouvez également effectuer une validation, générer du code et générer des modules de test.

Étape	Description
1	Localisez le champ d'édition dans la barre d'outils de cette fenêtre.
2	Cliquez sur la flèche déroulante dans ce champ. Une liste s'affiche, affichant tous les services Décision et les éléments Décision du Paquetage associé à l'artefact de configuration DMNSim. Dans ce cas, « Déterminer la cabine à mettre à niveau » est le seul élément de la liste.
3	Cliquez sur « Déterminer la cabine à mettre à niveau ».
4	Le corps de la fenêtre affiche maintenant les éléments InputData et les résultats de décision disponibles en tant qu'entrées pour la décision sélectionnée. Cliquez sur le bouton Enregistrer.
5	Utilisez la colonne 'Value' pour sélectionner l'un des DataSets prédéfinis pour les InputValues, puis vous pouvez cliquer sur le bouton ' Exécuter '  dans la barre d'outils inférieure pour exécuter une simulation, en utilisant les ensembles de données sélectionnés.

Intégrer dans BPSim pour Simulation

La force de DMN réside dans sa capacité à décrire les exigences métier à travers le diagramme des exigences Décision et à encapsuler la logique complexe dans des expressions polyvalentes telles que le Tableau de Décision et le Boxed Context.

De même, la force de BPMN réside dans sa capacité à décrire des processus métiers avec une Flux séquence de tâches et d'événements, ou à décrire des collaborations de processus avec des flux de messages.

Le diagramme Décision Exigences forme un pont entre les modèles Processus Métier et les modèles logiques de décision :

- Les modèles Processus Métier définissent les tâches au sein des processus métier, où une prise de décision est requise
- diagrammes Exigences Décision définissent les décisions à prendre dans ces tâches, leurs interrelations et leurs exigences en matière de logique décisionnelle.
- La logique Décision définit les décisions requises de manière suffisamment détaillée pour permettre la validation et/ou l'automatisation.

DMN fournit un modèle Décision complet qui complète un modèle Processus Métier en spécifiant en détail les prises de décision effectuées dans les tâches de processus.

Les deux exemples présentés dans cette rubrique sont accessibles à partir de :

- EAExample Modèle | Simulation de Modèle | Modèles BPSim
- Point de vue | Modélisation Métier | BPSim | Études de cas BPSim

Les expressions BPSim utilisent un modèle DMN de deux manières :

- Le service Décision du DMN - démontré par le processus de demande de prêt
- BusinessKnowledgeModel de DMN - démontré par le calcul des coûts de livraison

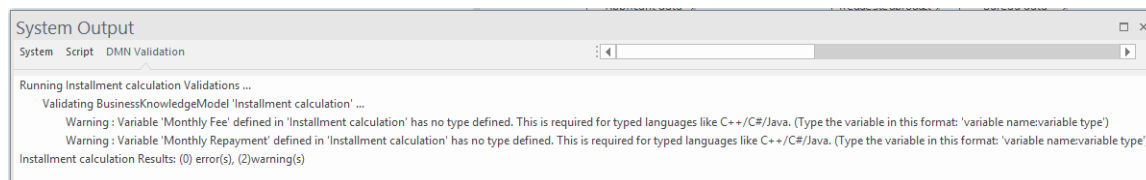
Le processus d'intégration d'un modèle DMN avec un modèle BPSim comprend :

- DMN Modèle Validation, Simulation , Génération de Code et Tester sur le module généré
- Configurer une dépendance d'utilisation de l'artefact BPSim vers l'artefact DMN
- Générer ou mettre à jour le BPMN DataObject à partir du DMN DataSet
- Créer des paramètres de propriété dans BPSim à utiliser sur les tâches et les flux Séquence sortant des passerelles
- Liez l'interface DMN aux paramètres de propriété BPSim

Validation Modèle DMN pour les langages compilés tels que Java

Lorsque vous créez un modèle DMN et que vous le simulez dans Enterprise Architect , le code qui pilote la simulation est JavaScript ; cela signifie que les variables n'ont pas besoin d'être explicitement typées (le type de variable est déduit de la valeur qui lui est assignée).

Cependant, pour les langages tels que C++, C# et Java, le compilateur signalera une erreur indiquant qu'une variable n'a pas de type.



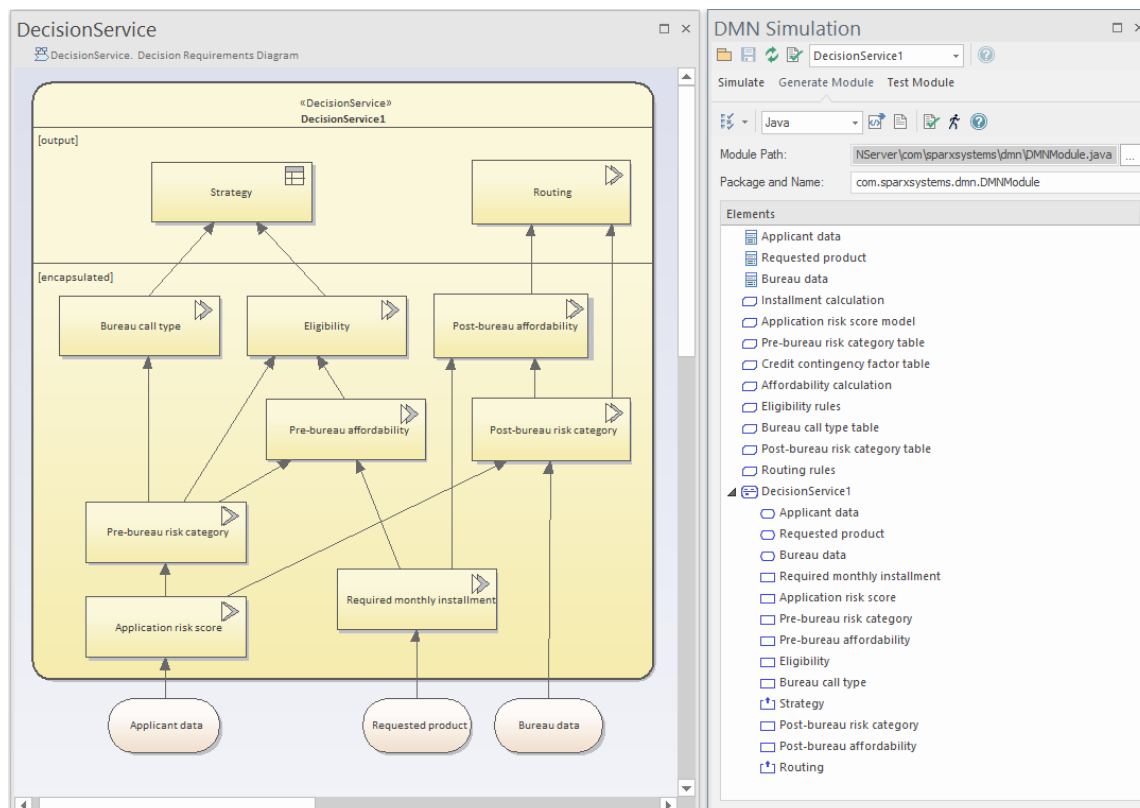
Pour la génération dans ces langages, vous devez exécuter la validation sur le modèle et utiliser les résultats pour trouver les variables qui nécessitent leur type défini. Par exemple:

- Paramètre Métier Knowledge Modèle - sélectionnez l'élément BKM à visualiser dans la fenêtre Expression DMN, cliquez sur le deuxième bouton pour ouvrir la dialogue 'Paramètre', précisez un type pour le paramètre


- Type Décision - sélectionnez l'élément Décision , ouvrez la fenêtre Propriétés , pour la propriété 'variableType' sélectionnez dans le champ 'Valeur'
- Clauses d'entrée/sortie Tableau de Décision - sur la clause d'entrée/sortie Tableau de Décision , cliquez-droit pour afficher le menu contextuel et choisir le type
- Variables de contexte encadré - reportez-vous à la rubrique d'aide [Boxed Context](#)

Génération de code DMN en Java

Après avoir utilisé la validation pour résoudre les problèmes de type de variable, nous pouvons passer à la page ' Générer Module' dans la fenêtre DMN Simulation .



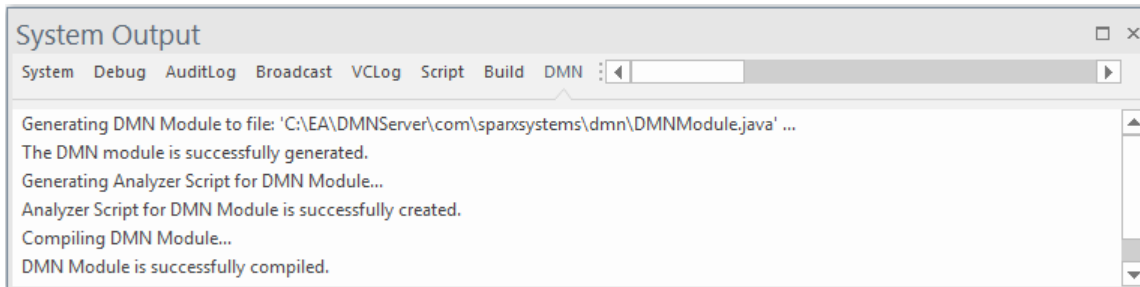
- Sélectionnez *DecisionService1* dans le champ de saisie de données de la barre d'outils supérieure ; tous les éléments impliqués dans *DecisionService1* seront maintenant inclus dans la liste
- La Définition Item et Modèle de Connaissance Métier sont des éléments globaux
- Les données d'entrée et les décisions sont encapsulées dans l'élément *DecisionService*
- Les langages pris en charge sont C++, C# , Java et JavaScript ; note que pour JavaScript le fichier .js généré est le même que le script de simulation (onglet 'Simulate' | Menu déroulant du bouton Exécuter | Générer un nouveau script (Scriptant Window)) sauf que les codes liés à la simulation sont omis
- Pour Java, la valeur 'Module Path' doit correspondre à la structure Paquetage ; dans cet exemple, le *DMNModule.java* doit être généré dans un répertoire pour former un chemin de fichier qui se termine par '*\com\sparxsystems\dmn\DMNModule.java*' - vous devez créer manuellement les structures de répertoires pour maintenant



Cliquez sur le bouton Générer du Code () dans la barre d'outils. Cet exemple utilisera Java ; cependant, C++ et C# sont identiques. Ces actions sont effectuées :

- Le fichier .java est généré vers le chemin spécifié
- Un script d'analyseur (script de construction) pour cet artefact est créé

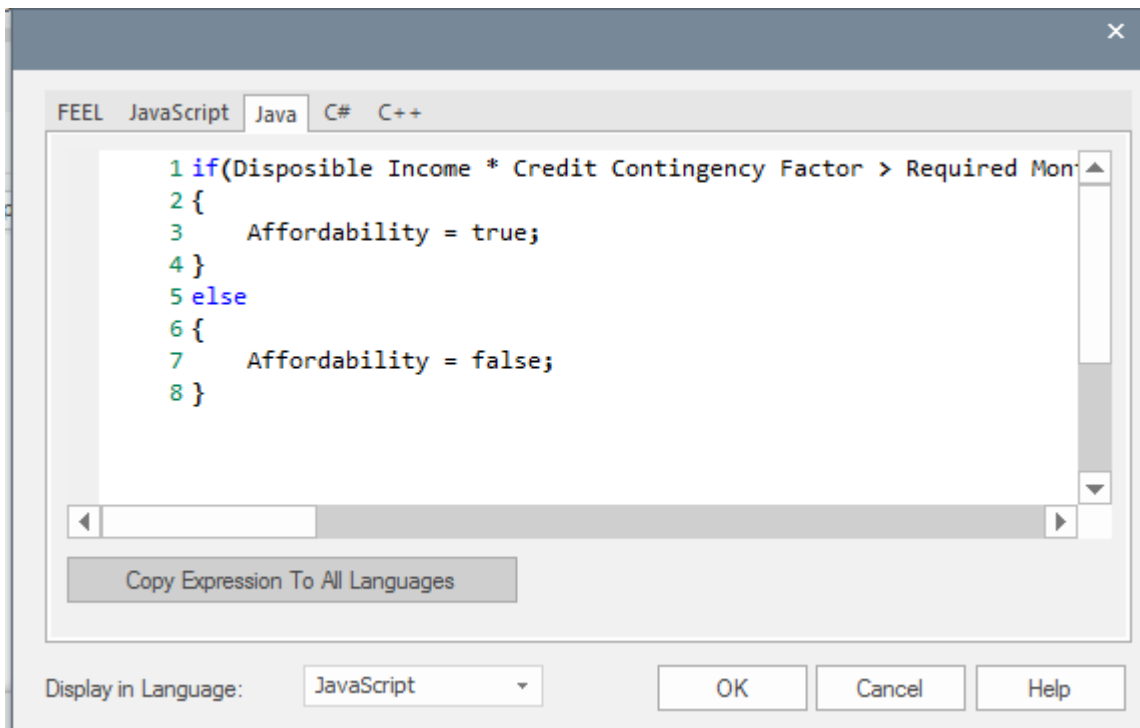
- Le script de construction de ce script d'analyseur est exécuté
- Les messages de progression sont signalés dans la fenêtre Sortie système

Si le modèle est valide, ce processus renverra le message :



S'il y a des erreurs de compilation, vous pouvez ouvrir le fichier .java généré en cliquant sur le bouton  à côté du bouton  dans la barre d'outils, résoudre manuellement le problème et compiler avec le script généré jusqu'à ce que vous réussissiez.

Une raison courante d'échec de compilation est que les langues peuvent avoir des grammaires différentes pour une expression. Vous devrez peut-être fournir une valeur pour une langue afin d'écraser la valeur par défaut (cliquez-droit sur une expression littérale DMN | Modifier l'expression).

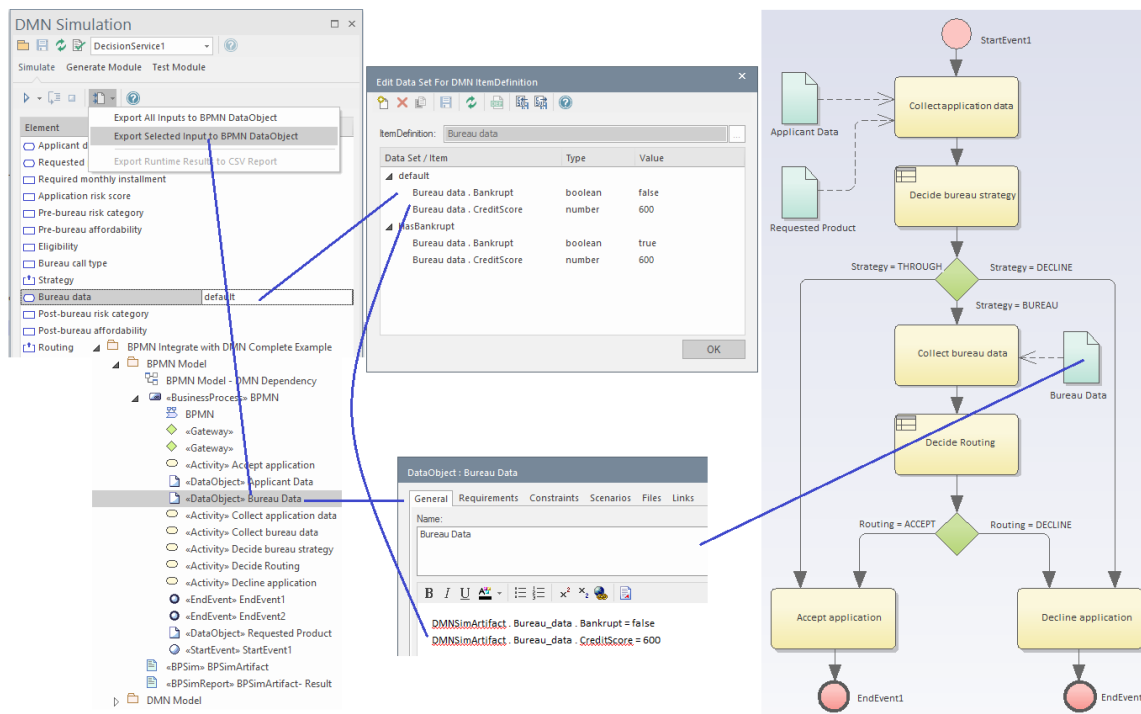


Tester les modules DMN avant utilisation externe


Après avoir généré le modèle en code Java et l'avoir compilé avec succès, nous souhaitons maintenant :

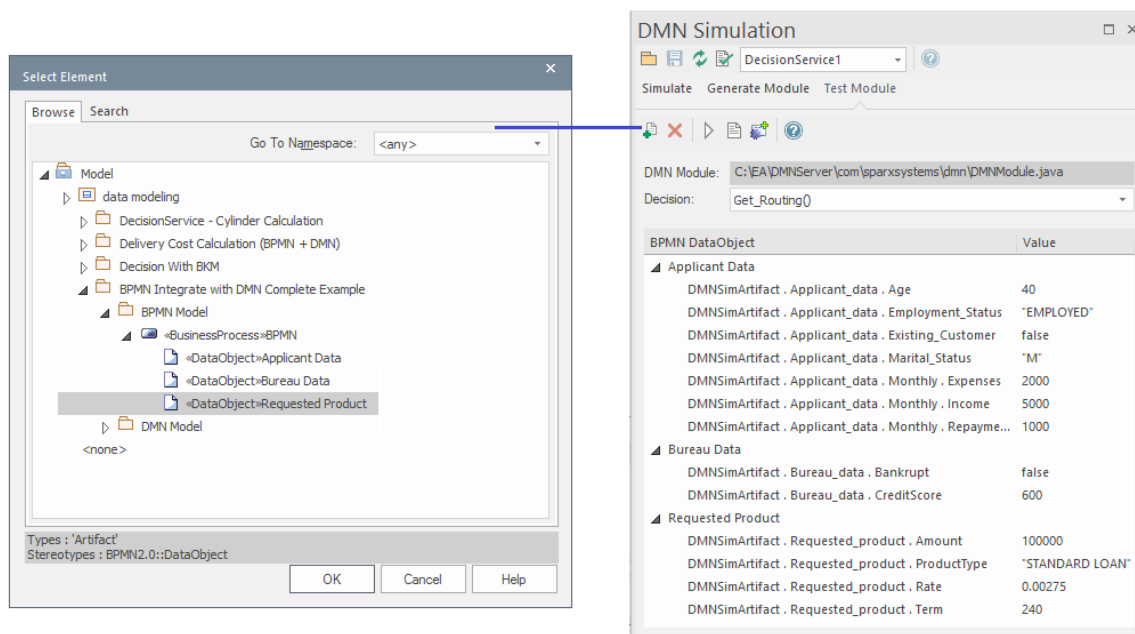
- Test l'exactitude de ce module
- Fournissez-lui des entrées
- Obtenez les valeurs Décision de sortie


Générer BPMN DataObject

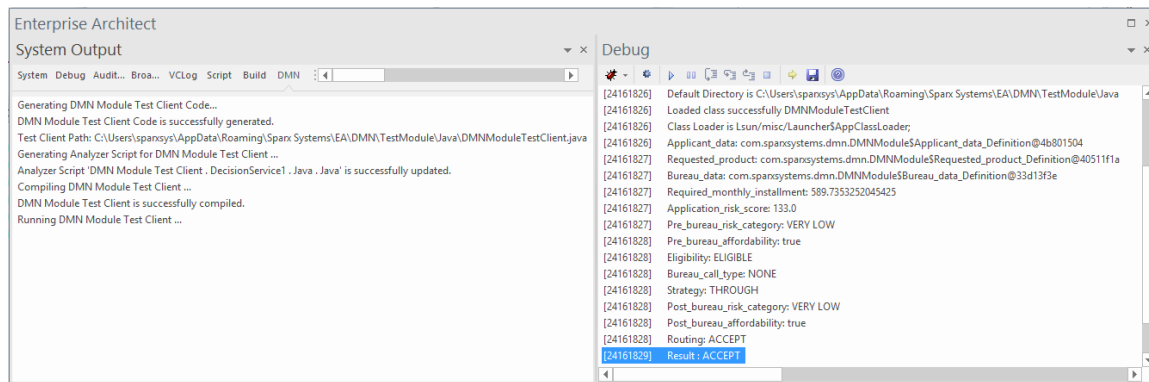


Les données transportées par l'ensemble de données sélectionné seront générées dans le champ « Notes » du BPMN DataObject.

- Cliquez sur le bouton  (2ème à droite sur la barre d'outils de l'onglet 'Générer Module') pour ouvrir l'onglet 'Test Module'



- Cliquez sur l'  dans la barre d'outils pour sélectionner les éléments BPMN DataObject d'entrée.
- Sélectionnez les sorties disponibles dans la combo 'Décision', telles que Get_Routing(), et cliquez sur le bouton Exécuter de la barre d'outils



Le résultat de l'exécution sera affiché dans la fenêtre Débugger. Vous pouvez également ouvrir le fichier du module de test, définir un point d'arrêt sur la ligne et déboguer dans le module DMN pour effectuer un débogage au niveau de la ligne.

Nous vous recommandons fortement de tester votre module DMN avec cette fenêtre pour garantir que le module DMN est fonctionnel avec les entrées données (des objets de données BPMN) et qu'il calculera avec succès le résultat de la sortie.

Note : Le chemin du module DMN est enregistré dans la propriété 'Filepath' de l'artefact DMNSimConfiguration.

Il est maintenant temps d'intégrer le module DMN au modèle BPSim.

La première étape consiste à configurer la dépendance d'utilisation entre l'artefact BPSim et l'artefact Simulation DMN.



Note : Un Artefact BPSim peut utiliser plusieurs modules DMN si nécessaire. Ceci est pris en charge en plaçant simplement tous les artefacts DMN sur ce diagramme et en dessinant un connecteur de dépendance de l'artefact BPSim vers chaque artefact Simulation DMN.

Ces rubriques d'aide fournissent deux exemples d'utilisation de ces méthodes. Voir:

- Exemple : Intégrer le service Décision DMN dans Object de données BPSim et Paramètres Propriété
- Exemple : Intégrer DMN Métier Knowledge Modèle dans BPSim Paramètres Propriété

Exemple : Intégrer le service Décision DMN dans Object de données BPSim et Paramètres Propriété

Un exemple d'intégration d'un service DMN Décision dans le modèle BPSim est fourni dans l'Assistant de Modèle pour BPSim.

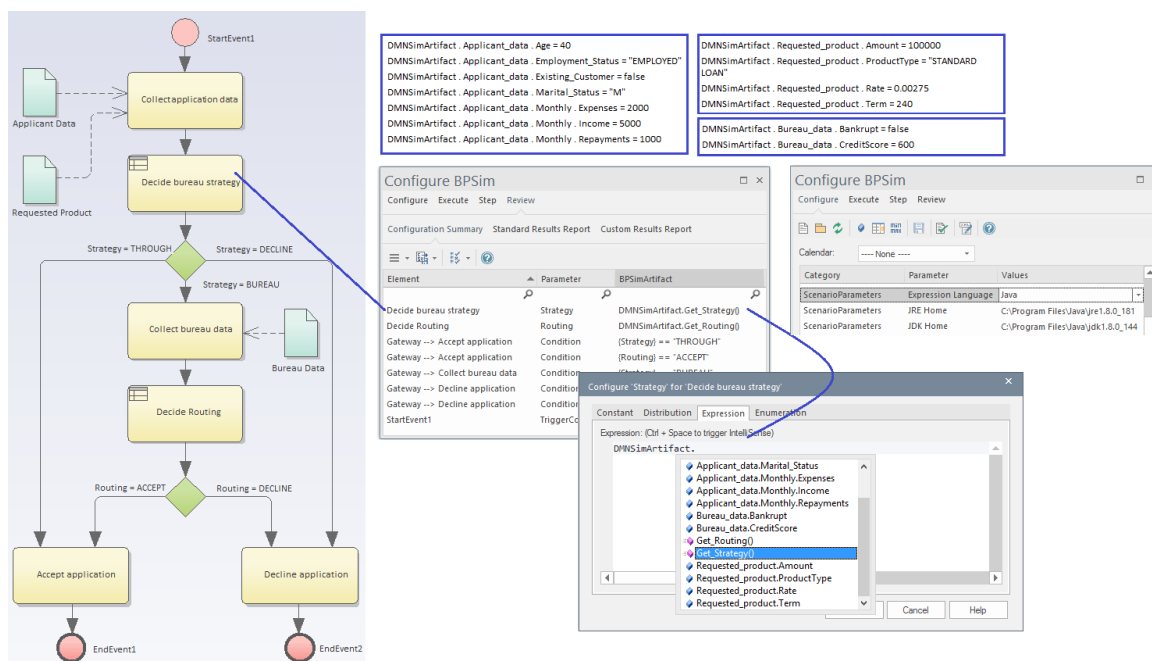
Pour y accéder :

- Poser la *perspective* de la *Modélisation Métier* > *BPSim*
- Ouvrez l'Assistant de Modèle (onglet Page Démarrage 'Créer à partir de Motif')
- Dans le groupe Études de cas BPSim, sélectionnez *BPMN Intégrer à l'exemple complet DMN*.
- Cliquez sur le bouton Créer un ou plusieurs Modèle .

Cela créera des modèles BPMN et DMN configurés pour simuler un appel à un modèle DMN à partir du modèle BPMN.

Note : Afin d'intégrer le module DMN, le langage d'expression doit utiliser Java et le JRE et le JDK doivent être configurés correctement (la version minimale de Java est 1.7). Voir *Installer le Moteur d'Exécution BPSim* dans la rubrique d'aide [BPSim Business Simulations](#) .

Dans ce diagramme BPMN, il y a trois DataObjects (aqua) connectés aux activités BPMN. Ces éléments DataObject transportent des données d'entrée, générées à partir de la fenêtre DMN Simulation .



- Lorsque la simulation est en cours d'exécution, elle chargera automatiquement tous les DataObjects se connectant à la tâche lors du passage du jeton de simulation.
- La deuxième tâche Règles Métier 'Decide bureau strategy' est configurée pour définir la propriété 'Strategy' à la valeur 'DMNSimArtifact.Get_Strategy()' ; vous n'avez pas besoin de taper ceci - appuyez sur Ctrl+Espace pour vous aider à modifier l'expression

Une fois ceux-ci définis, cliquez sur l'onglet « Exécuter » et simulez le modèle. Vous pouvez ensuite visualiser le rapport ou accéder à la page « Étape » pour effectuer le débogage étape par étape du modèle BPSim.

Exemple : Intégrer DMN Métier Knowledge Modèle dans BPSim Paramètres Propriété

Dans certains cas, vous souhaitez peut-être simplement concevoir un Tableau de Décision à utiliser dans un modèle BPMN. Si tel est le cas, il n'est pas nécessaire de passer par les processus de création d'un Service Décision, Décision, d'une Données d'Entrée ou encore d'une Définition Item, puisqu'un Modèle de Connaissance Métier (BKM) peut être directement interfacé.

Un exemple d'intégration d'un DMN BKM dans le modèle BPSim est fourni dans l'Assistant de Modèle pour BPSim.

Pour y accéder :

- Poser la *perspective* de la *Modélisation Métier* > *BPSim*
 - Ouvrez l'Assistant de Modèle (onglet Page Démarrage 'Créer à partir de Motif')
 - Dans le groupe d'études de cas BPSim, sélectionnez *BPMN Integrate with DMN - Delivery Cost Calculation*
 - Cliquez sur le bouton Créer un ou plusieurs Modèle
1. Créez un Modèle de connaissances Métier simple sous forme de Tableau de Décision (vous pouvez également créer d'autres expressions telles que le contexte encadré ou les expressions littérales) avec des paramètres, puis modélisez la logique (clause d'entrée, clause de sortie, règles) et testez-la (le « paramètre d'entrée » Onglet Valeurs pour Simulation dans la fenêtre Expression DMN).

The screenshot displays three windows from the BPSim Assistant:

- DMN Expression:** Shows a decision table for 'Compute Delivery Cost' with columns for 'U', 'Amount', 'Total Weight', and 'Delivery Cost'. The table contains 11 rows of data.
- DMN Simulation:** Shows a BPMN process flow with a task 'Compute Delivery Cost' connected to a 'DummyDecision'.
- Configure BPSim:** Shows the configuration for the 'Compute Delivery Cost' task, including parameters like 'DeliveryCost', 'Weight', 'Price', and 'TriggerCount'.

2. Connectez le BKM à une Décision avec un connecteur d'exigence de connaissances. Cette Décision sert de nom de groupe à un certain nombre de fonctions du BKM ; vous pouvez simplement saisir un nombre tel que « 10 » dans l'expression. Par exemple, si vous souhaitez générer du code Java avec seulement cinq BKM (en considérant que votre modèle peut contenir plus d'une centaine de BKM), vous pouvez connecter ces cinq BKM à une Décision et sélectionner cette Décision dans la fenêtre Simulation DMN, puis les cinq BKM seront être inclus automatiquement.
3. Générer du code Java et (en supposant que tout soit correct) la compilation réussira.
4. Dans la configuration BPSim, nous utilisons simplement Intelli-sense pour construire l'expression de la tâche « Calculer le coût de livraison ».

Dans cet exemple, la tâche 'Générer le prix et le poids du meuble' générera des valeurs aléatoires aux propriétés 'Poids' et 'Prix', puis la tâche 'Calculer le coût de livraison' transmettra la valeur au Métier Knowledge Modèle et le résultat sera reporté à la propriété 'DeliveryCost'.

Vous pouvez maintenant exécuter la simulation et suivre le processus de débogage pour observer, par exemple, les changements valeur d'attribut.

Importation de XML DMN


Enterprise Architect supporte l'importation d'un fichier XML DMN 1.1 ou 1.2 dans un projet, avec à la fois la sémantique du modèle et les informations d'échange de diagrammes.

Accéder

Dans la fenêtre Navigateur, sélectionnez le Paquetage dans lequel importer le fichier XML. Utilisez ensuite l'une des méthodes décrites ici pour ouvrir la boîte dialogue « Importer Paquetage à partir de DMN 1.1 XML ».

Ruban	Publier > Échange de Modèles > Importer > DMN 1.1
Raccourcis Clavier	Ctrl+Alt+I : Autres formats XML > DMN 1.1

Importer du XML DMN 1.1

Pas, pas	Action, Action
1	Dans le champ « Nom de fichier », saisissez le chemin et le nom du fichier source, ou cliquez sur l'icône  pour localiser et sélectionner le fichier.
2	Cliquez sur le bouton Importer pour importer le fichier dans le Paquetage.

Importer l'exemple depuis OMG

1. Téléchargez le fichier zip sur [this link](#) et extrayez-le dans votre gestionnaire de fichiers.
2. Recherchez le dossier *exemples/Chapitre 11/*.
3. Cliquez sur le fichier *Chapitre 11 Exemple.dmn* et importez-le au format DMN 1.1.

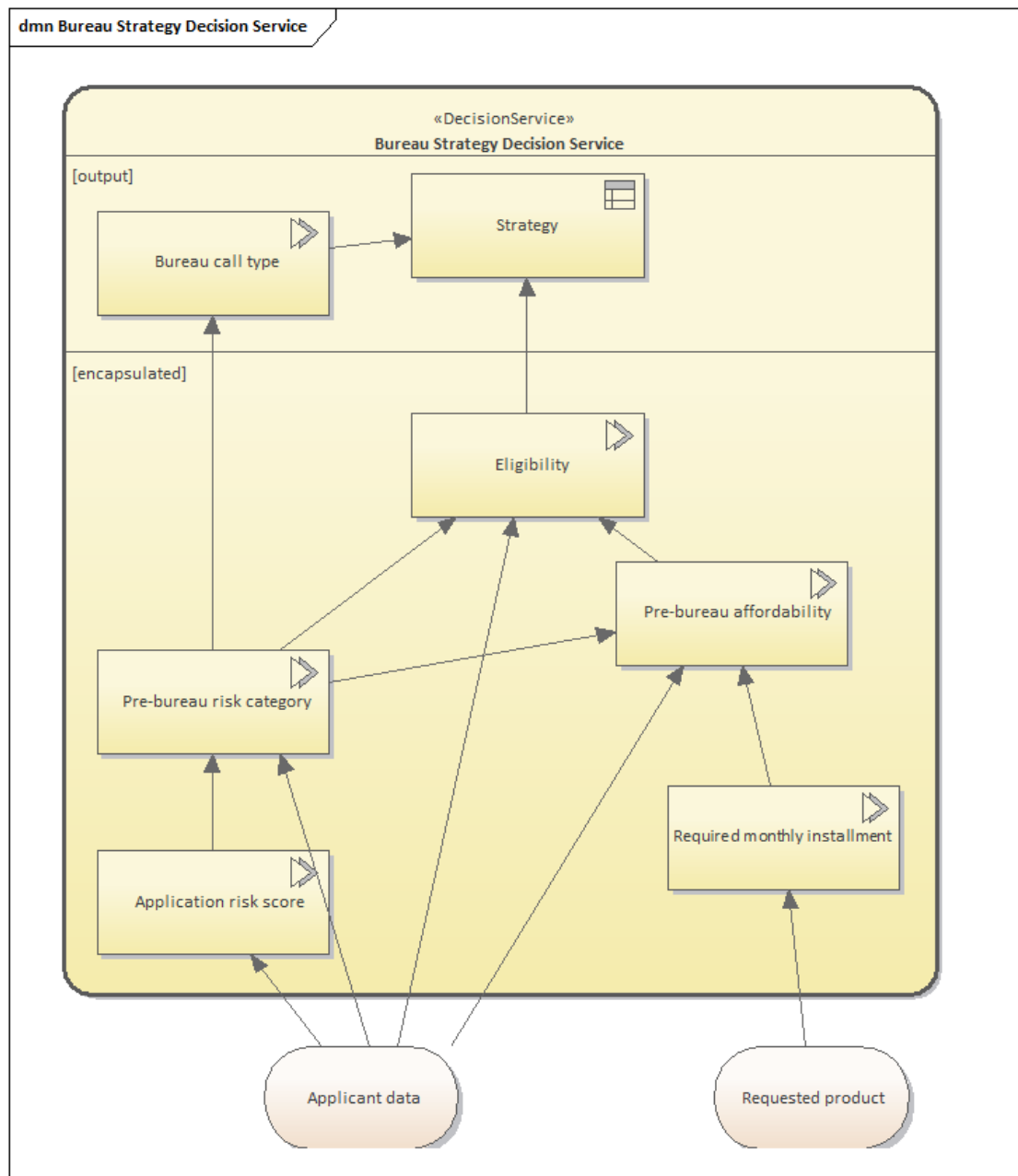
Ces diagrammes sont importés pour montrer différentes perspectives du modèle :

- DRD de toute prise de décision automatisée
- DRD pour le point de décision Demande Révision
- DRD pour le point de décision Decide Routing
- DRD pour le point de décision de la stratégie du Decide Bureau

Ces diagrammes sont importés pour définir les Services Décision :

- Service Décision stratégique du Bureau
- Service Décision de routage

Le diagramme « Bureau Strategy Décision Service » est présenté ici. Il comporte deux éléments de données d'entrée (données du candidat, produit demandé), deux Décisions de sortie (type d'appel du bureau, stratégie) et cinq décisions encapsulées. Note que les modèles de connaissances Métier invoqués ne sont pas affichés sur le diagramme.



Afin de générer du code de production à partir du modèle, vous devrez peut-être exécuter une validation et une simulation pour vous assurer que le modèle importé possède les expressions correctes.

1. Créez un artefact de configuration DMN Sim sur l'un des diagrammes répertoriés et double-cliquez dessus pour l'ouvrir dans la fenêtre Simulation DMN.
2. Les services Décision et les décisions sont répertoriés dans le champ déroulant cible. Une fois que vous avez spécifié une cible, tous les éléments requis sont répertoriés dans la fenêtre.
3. Cliquez sur le bouton Valider (4ème sur la barre d'outils). Si des messages d'erreur ou d'avertissement s'affichent, nous vous suggérons de résoudre les problèmes comme indiqué dans les descriptions d'erreur ou d'avertissement, avant d'effectuer la simulation.
4. Fournissez des valeurs appropriées pour les entrées et exécutez la simulation ou déboguez le modèle par étapes.

Note : L'exemple 'Bureau Strategy Décision Service' est également disponible dans l'EAExemple Modèle . Dans le diagramme ' Démarrage ', sélectionnez ' Métier Modélisation > Exemples DMN > Bureau Strategy Décision Service'.

Diagrammes Exigences Décision

Les éléments modélisés dans les graphes Décision Exigences (DRG) et diagrammes Décision Exigences (DRD) sont Décision, Métier Knowledge Modèle, Input Data, Knowledge Source et Décision Service. Les dépendances entre ces éléments expriment trois types d'exigences : l'information, la connaissance et l'autorité.

Composantes des Diagrammes Exigences Décision

Ce tableau résume la notation de toutes les composantes d'un diagramme Décision Exigences.

Composant	Description
Décision	Un élément Décision désigne l'acte de déterminer une sortie à partir d'un certain nombre d'entrées (Données d'entrée ou Décision), en utilisant une logique de décision exprimée sous forme d'expressions littérales, Tableaux Décision, d'invocations ou de contexte encadré.
Modèle de connaissances Métier	Un Métier Knowledge Modèle désigne un module réutilisable de logique de décision représenté par une fonction, qui comprend zéro, un ou plusieurs paramètres.
Service Décision (élargi)	Un Service Décision peut renfermer un ensemble de décisions réutilisables qui sont invoquées en interne - par exemple, par une autre Décision ou Métier Knowledge Modèle - ou en externe - par exemple, par un Processus BPMN. Une bonne pratique consiste à utiliser un diagramme pour décrire un seul service Décision étendu.
Service Décision (effondré)	Si un élément Décision Service sert d'élément invocable, connecté avec des exigences de connaissances à d'autres éléments avec une logique d'invocation, nous pouvons masquer les détails du Décision Service pour nous concentrer sur les hiérarchies de décision dans leur ensemble.
Des données d'entrée	Un élément de données d'entrée désigne les informations utilisées comme entrée dans une ou plusieurs décisions.
Définition Item	Une définition Item est utilisée pour définir le type et la structure des éléments de données utilisés dans le modèle de décision. Il est principalement référencé par les éléments de données d'entrée comme base pour le type et la structure des données qui devraient être entrées. Il peut également être référencé pour définir la structure d'une sortie. La définition Item contient Ensembles de données qui fournissent des ensembles de valeurs utiles lors de l'exécution de simulations variées.
Source de connaissances	Un élément Knowledge Source désigne une autorité pour un Métier Knowledge Modèle ou Décision.
Exigence d'informations	Une exigence d'information indique que les données d'entrée ou la sortie Décision sont utilisées comme entrée dans une Décision.
Exigence de connaissances	Une Exigence de Connaissances désigne l'invocation d'un Modèle de Connaissances Métier ou d'un Service Décision.
Exigence d'autorité	Une exigence d'autorité dénote la dépendance d'un élément DRG à l'égard d'un

	autre élément DRG qui agit comme source de guidage ou de connaissance.
--	--

Éditeur d'expression Décision

L'éditeur d'expression DMN est la fenêtre dans laquelle vous définirez, réviserez et mettrez à jour les détails de la plupart des différents types d'éléments DMN au sein de votre modèle. Principalement, il est utilisé pour éditer les expressions de valeur des éléments Décision et les éléments BusinessKnowledgeModel (BKM).

Une version différente de l'éditeur d'expression DMN est affichée pour chacun des quatre types d'expression valeur utilisés par les éléments Décision et les éléments BKM. Pour les éléments BKM, un deuxième onglet de fenêtre est également présenté, pour définir les paramètres d'entrée et de sortie utilisés pour appeler le BKM.

Deux versions supplémentaires de DMN Expression Editor existent également pour supporter la modification des éléments ItemDefinition et InputData.

La barre d'outils affichée et la disposition du contenu de la fenêtre dépendent du type d'élément DMN actuellement sélectionné et, le cas échéant, du type d'expression de valeur en cours de définition.

Cette image montre la version de l'éditeur d'expression DMN utilisée pour définir un Tableau de Décision. Dans ce cas, l'élément sous-jacent est un BusinessKnowledgeModel, et donc la logique de décision est « invoquée » par d'autres éléments, avec des entrées et des sorties transmises via des paramètres.

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

Des explications détaillées des fonctionnalités de DMN Expression Editor pour chaque élément et type d'expression sont fournies dans les rubriques d'aide enfants de cette rubrique.

Accéder

Diagramme	<p>Double-cliquez sur un élément DMN sur un diagramme .</p> <p>La fenêtre de l'éditeur d'expression DMN correspondant à l'élément et à son type d'expression s'affiche.</p>
-----------	---

Expressions de valeur

Ce tableau résume quatre types distincts d'expression valeur avec des références aux rubriques d'aide détaillant chacun d'eux.

Type et Icône	Description
Tableau de Décision	Un Tableau de Décision est une représentation tabulaire d'un ensemble d'expressions d'entrée et de sortie associées, organisées en règles indiquant quelle entrée de sortie s'applique à un ensemble spécifique d'entrées d'entrée.
Expression littérale	Une expression littérale spécifie la logique de décision sous la forme d'une expression textuelle qui décrit comment une valeur de sortie est dérivée de ses valeurs d'entrée. Pour supporter la simulation et l'exécution, l'expression littérale peut

	utiliser des fonctions JavaScript .
Contexte encadré	<p>Un contexte encadré est une collection d'entrées de contexte, constituées de paires (nom, valeur), chacune avec une valeur de résultat .</p> <p>Les entrées de contexte fournissent un moyen de décomposer une expression complexe en une série d'expressions simples, fournissant des résultats intermédiaires pouvant être utilisés dans des entrées de contexte ultérieures.</p>
Invocation	<p>Un appel fait appel à un autre élément de modèle (un BusinessKnowledgeModel ou un Décision Service) pour fournir un résultat de décision. L'invocation définit les paramètres qui sont transmis à l'élément « invoqué », fournissant un contexte pour l'évaluation de sa logique de décision. Le résultat de la décision est ensuite renvoyé à l'élément « invoquant ».</p>

Éléments ItemDefinition et InputData

Élément	Description
Définition de l'élément	<p>Les éléments ItemDefinition sont utilisés pour définir des structures de données et, éventuellement, pour restreindre la plage de valeurs autorisées des données. Les définitions d'éléments peuvent aller d'un simple type unique à un type structuré complexe. Les ItemDefinitions sont utilisées pour spécifier le type d'éléments InputData ainsi que les paramètres d'entrée.</p>
Des données d'entrée	<p>Les éléments InputData sont utilisés pour fournir une entrée aux éléments Décision .</p> <p>Le type de données d'un élément InputData est défini à l'aide d'un élément ItemDefinition. Ensembles de données peuvent également être définis dans le cadre d'un ItemDefinition et un élément InputData peut ensuite spécifier un ensemble de données à utiliser lors de l'exécution d'une simulation.</p>

Relation

Un élément DMN Décision Relation fournit une méthode abrégée pratique pour définir une liste de valeurs associées dans un diagramme DMN. Une relation Décision est comme un tableau de relations avec des colonnes et des lignes. L'en-tête de la grille affiche le nom de chaque colonne. Chaque ligne affiche l'ensemble des valeurs des colonnes correspondantes.

Accéder

Boîte à outils	<p>Pour créer une relation Décision :</p> <ul style="list-style-type: none">• Assurez-vous que la Perspective est définie sur : Exigences > Décision Modélisation• Depuis la page Composants DMN de Diagramme Toolbox, faites glisser un élément Décision ou un élément BKM sur un diagramme DMN.• Sélectionnez « Relation » dans le menu contextuel d'expression.• Double-cliquez sur l'élément DMN pour afficher la fenêtre Expression DMN.
----------------	---

Vous pouvez également modifier un élément DMN Décision ou BKM existant en un type Décision-Relation. Pour faire ça:

- Cliquez-droit sur un élément DMN Décision ou BKM du diagramme , et sélectionnez l'option ' Propriétés | Option de menu Propriétés
- Sur la page Général, sélectionnez l'onglet 'Tags' , et dans le champ valeur 'expressionType' cliquez sur la flèche déroulante et sélectionnez 'Relation' ; cliquez sur le bouton OK

Aperçu

Le type de décision *de relation* DMN est une liste verticale contenant des lignes de valeurs. Une clé pour utiliser la relation Décision est le moyen de parcourir les lignes de valeurs à l'aide d'une *boucle For* . La boucle For peut être définie, par exemple, dans une expression littérale Décision associée en tant que formule pour traiter les lignes de l'élément Décision-Relation.

Modification des relations DMN

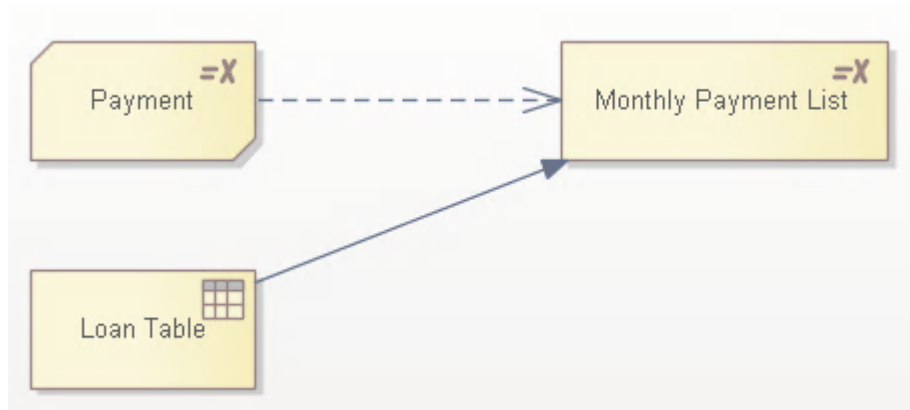
La fenêtre Expression DMN comporte une barre d'outils offrant des options pour ajouter une nouvelle ligne, supprimer une ligne existante et déplacer la ligne sélectionnée vers le haut ou vers le bas.

Vous pouvez aussi:

- Faites glisser l'en-tête de la grille pour repositionner les colonnes.
- Cliquez-droit sur les cellules d'en-tête pour afficher les options du menu contextuel permettant de définir le Type de la colonne sur : ' string ', ' number', ' boolean' ou ' object '.

Exemple - Tableau de prêt

Dans cet exemple, nous avons deux Décisions - 'Tableau de Prêt' et 'Liste de Paiement Mensuel' - et un Modèle de Connaissance Métier - 'Paiement'.



'Prêt Tableau' est une Décision implémentée sous forme de Relation avec quatre colonnes : 'Prêt', 'Principe', 'Durée' et 'Taux Annuel'.

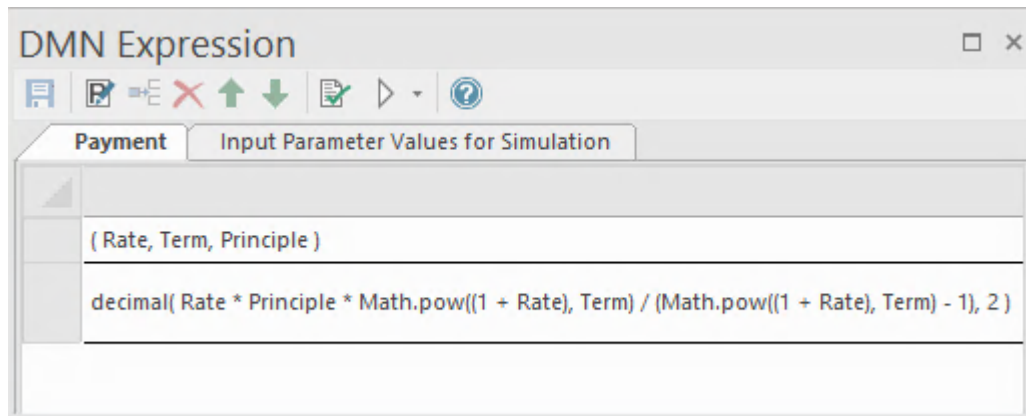
Loan Table				
	Loan : string	Principe : number	Term : number	Annual Rate : number
	Loan 1	100000	360	0.02
	Loan 2	100000	360	0.05
	Loan 3	100000	360	0.10

« Liste de paiements mensuels » est une Décision implémentée sous forme d'expression littérale avec une boucle *for* :

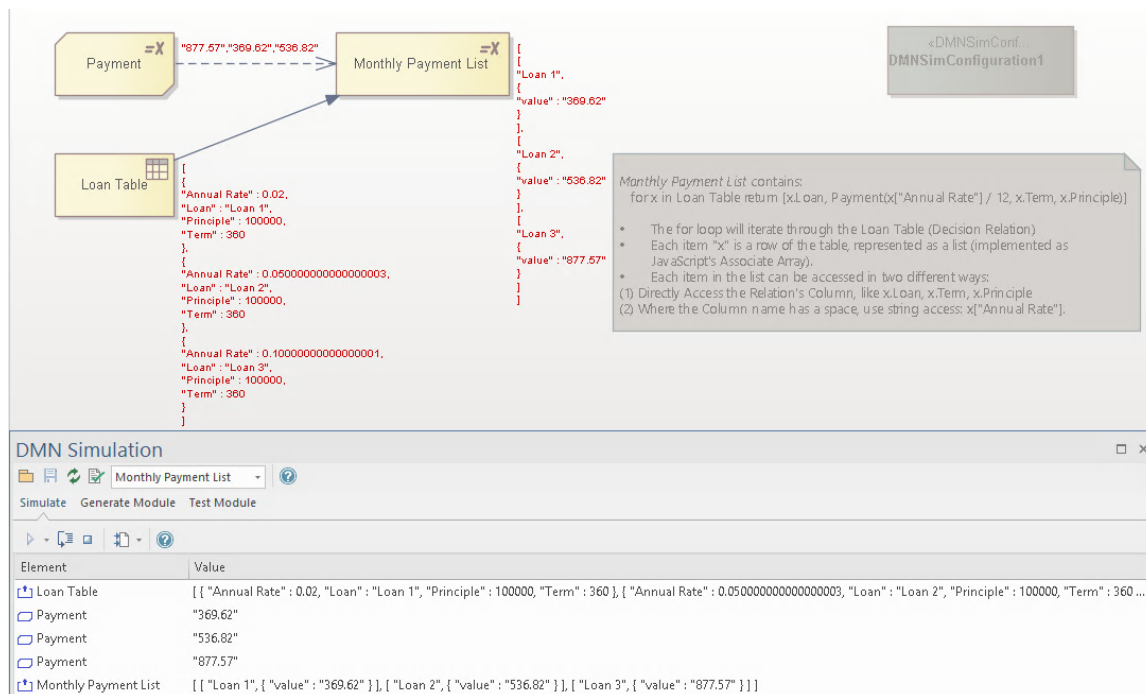
Monthly Payment List	
for x in Loan Table	return [x.Loan, Payment(x["Annual Rate"] / 12, x.Term, x.Principle)]

La boucle *for* parcourra 'Loan Tableau' :

- Chaque élément 'x' est une ligne du tableau, représentée sous forme de liste
- Avec chaque élément de ligne 'x', invoquez Métier Knowledge Modèle 'Paiement' avec les éléments de la liste de lignes
- Chaque élément de la liste est accessible de deux manières différentes :
 - (1) Accédez directement à la colonne Relation, comme x.Loan, x.Term, x.Principle
 - (2) Lorsque le nom de la colonne comporte un espace, utilisez l'accès string : x["Taux annuel"]



Lors de la simulation, les valeurs d'exécution s'affichent dans la fenêtre Simulation et sur le diagramme à côté de l'élément ; vous pouvez cliquer sur une étape pour voir le processus de simulation.



Liste encadrée

Une liste encadrée DMN est un élément Décision qui contient une liste d'expressions encadrées. Ces éléments sont disposés sous forme de liste verticale dans la fenêtre Expression DMN.

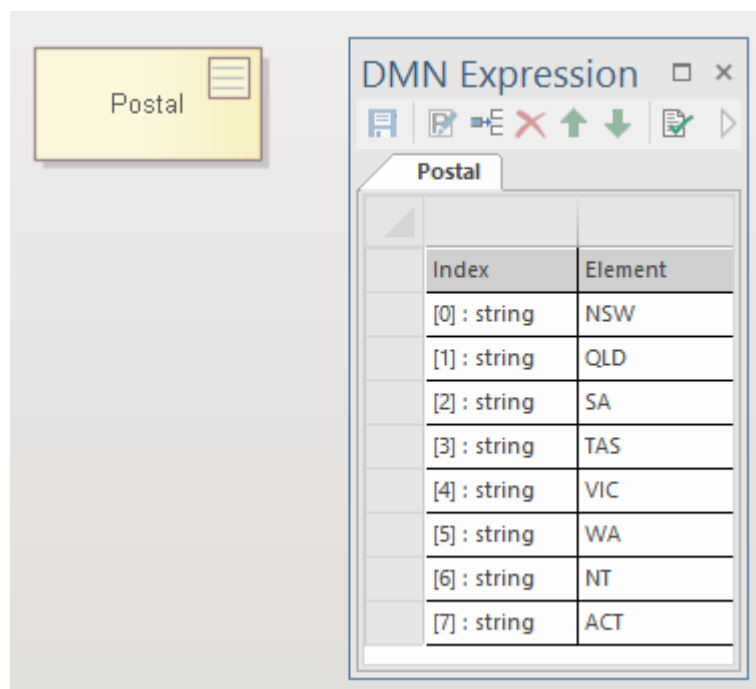
Une liste encadrée est souvent utilisée conjointement avec une expression *de boucle for* contenue dans un élément Décision associé. L'expression *de boucle for* est utilisée pour parcourir chaque ligne de la liste encadrée, en liant le champ d'élément de la liste à la variable correspondante et en évaluant l'expression dans la portée. La sortie de la *boucle for* est une liste contenant l'évaluation de l'expression pour chaque itération individuelle.

Accéder

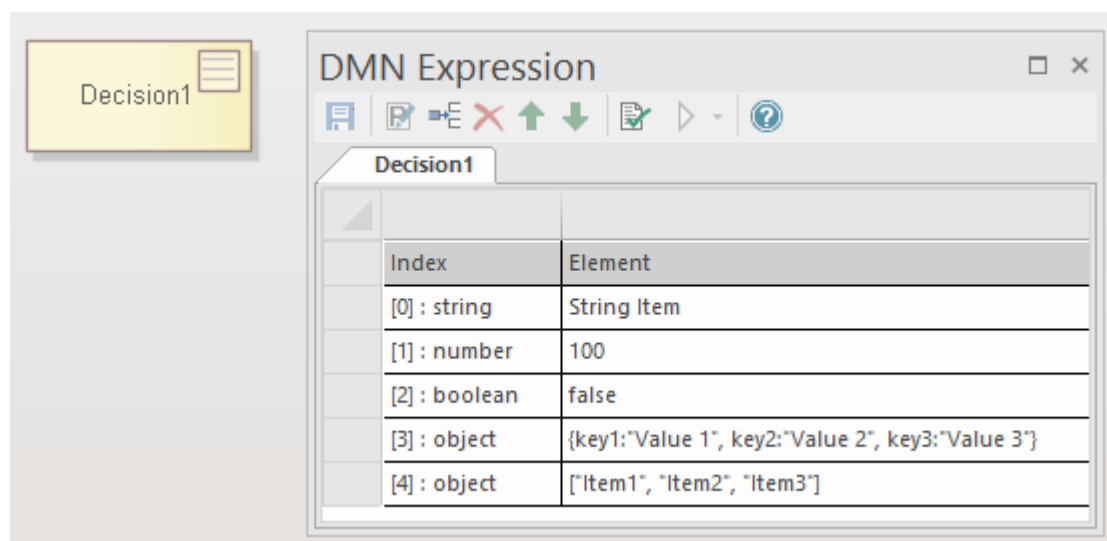
Boîte à outils Diagramme	Faites glisser un élément Décision ou un élément BKM de la boîte à outils sur un diagramme DMN et sélectionnez « Liste » dans le menu contextuel d'expression. Double-cliquez sur l'élément DMN ; la fenêtre DMN Expression s'affiche, affichant les détails de l'élément sélectionné.
Propriétés	Cliquez-droit sur un élément DMN Décision ou BKM du diagramme , et sélectionnez l'option ' Propriétés Option de menu Propriétés . Sur la page Général, sélectionnez l'onglet 'Tags' , et dans le champ valeur 'expressionType' cliquez sur la flèche déroulante et sélectionnez 'Liste'. Cliquez sur le bouton OK . Double-cliquez sur l'élément DMN ; la fenêtre DMN Expression s'affiche, affichant les détails de l'élément sélectionné.

Aperçu

Il est courant que les listes encadrées soient utilisées comme énumérations, où tous les éléments de la liste sont du même type.



Il est également courant que les listes encadrées soient utilisées comme une collection de données, où chaque élément peut avoir un type différent.



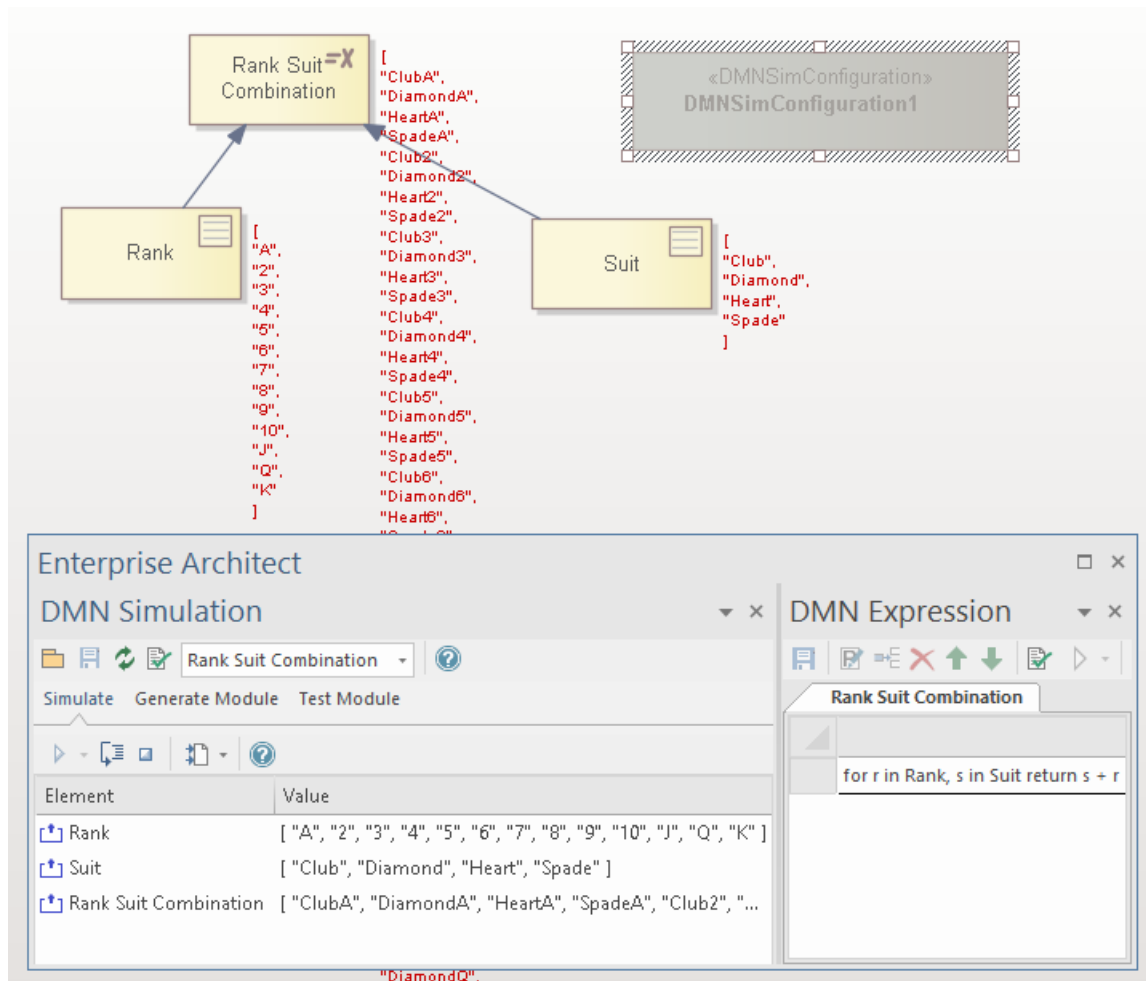
Modification des listes encadrées

La fenêtre DMN Expression comporte une barre d'outils proposant les options « Ajouter un nouvel Item de liste », « Supprimer Item de liste existant » et « Déplacer Item vers le haut ou vers le bas ».

Cliquez-droit sur un élément de liste pour afficher les options du menu contextuel permettant de définir le type de l'élément de liste - string, number, boolean ou object.

Exemple - Rang et couleur au poker

Dans cet exemple, nous avons trois décisions : rang, costume et combinaison de costume de rang.



- Le rang Décision est représenté sous la forme d'une liste encadrée avec 13 Items de « A » à « K »
- Décision Suit est représenté sous forme d'une liste encadrée avec 4 Items : "Club", "Diamond", "Heart" et "Spade".
- La combinaison de combinaison de rang Décision est représentée sous la forme d'une expression littérale avec une boucle *for* : *for r dans Rank, s dans Suit return s + r*

Lorsque plusieurs contextes d'itération sont définis dans la même expression de boucle *for*, l'itération résultante est un produit croisé des éléments des contextes d'itération. L'ordre d'itération va du contexte d'itération interne au contexte d'itération externe.

Dans cet exemple, le produit croisé de Rank (13 éléments) et Suit (4 éléments) est une liste de $13 * 4 = 52$ éléments.

Modifier Dialogue d'expression DMN

La dialogue « Modifier l'expression DMN » est utilisée pour définir des expressions dans les types d'éléments Boxed Content, Invocation et Literal Expression. Il fournit support d'Intelli-sense pour la construction d'expressions basées sur la grammaire FEEL, ainsi que les langages de code pouvant être utilisés pour la génération de code du modèle.

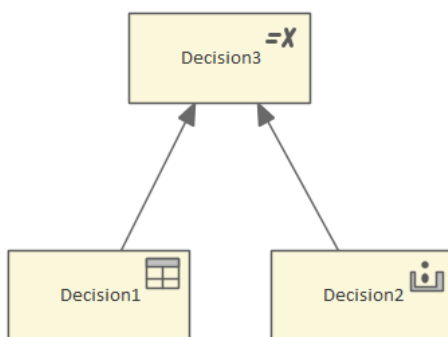
support de l'éditeur d'expression DMN et d'Intelli-sense

Pour vous aider à modifier des expressions avec moins de saisie et moins d'erreurs, Enterprise Architect fournit support Intelli-sense pour la modification des expressions.

Note que les noms des paramètres et des variables d'entrée de contexte peuvent contenir des espaces, conformément à la spécification du langage FEEL. Cette fonctionnalité est destinée à rendre chaque expression facile à lire.

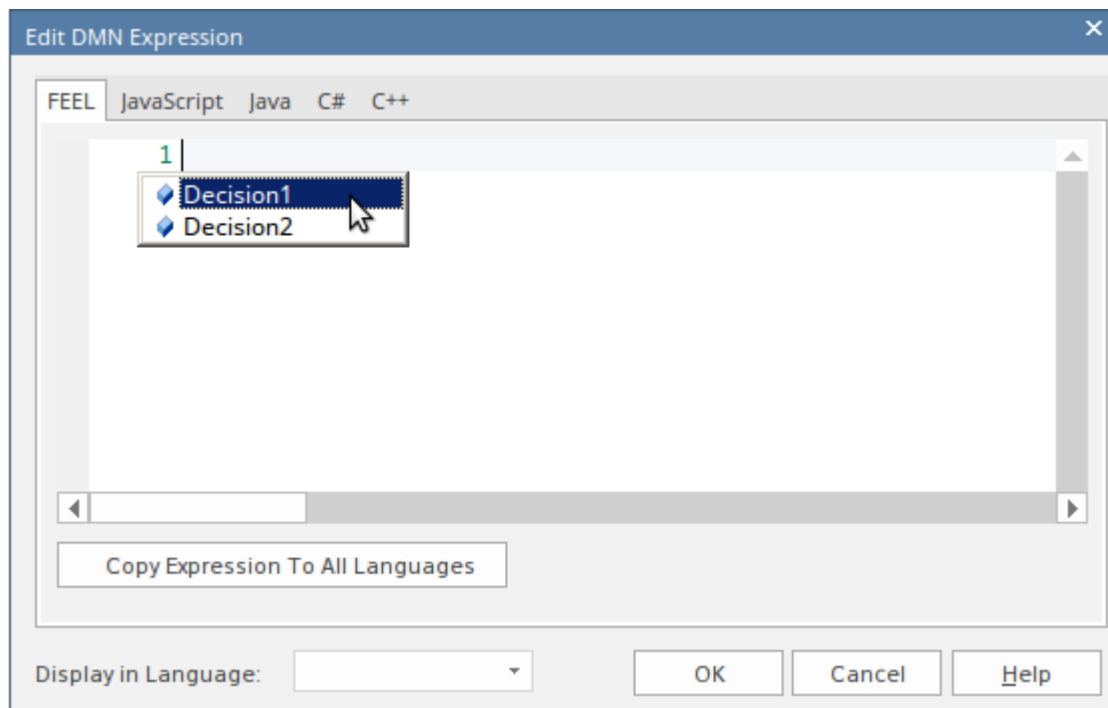
Exemples

Compte tenu de cette hiérarchie de décision, l'expression dans « Décision3 » est capable d'utiliser les résultats des deux décisions référencées.



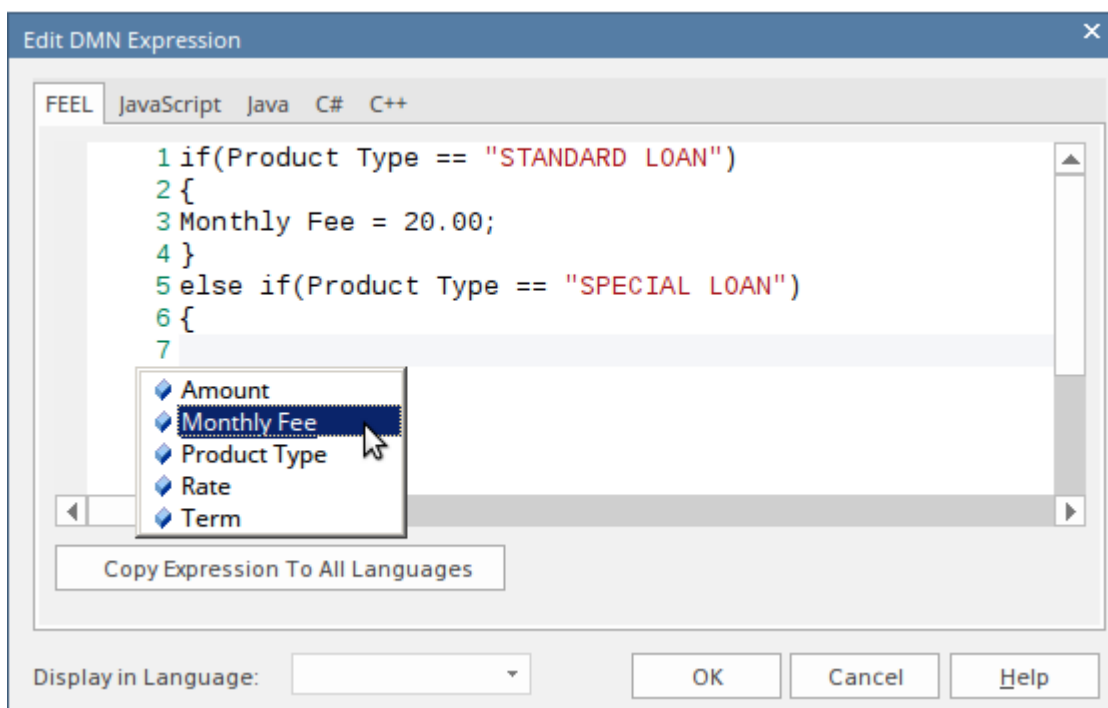
Pour ouvrir la dialogue « Modifier l'expression DMN » :

1. Double-cliquez sur l'élément Décision dans le diagramme , pour afficher la fenêtre Expression DMN.
2. Cliquez-droit sur la ligne d'expression et sélectionnez l'option de menu 'Modifier l'expression'. La dialogue « Modifier l'expression DMN » s'affiche.
3. Cliquez sur une ligne et appuyez sur Ctrl+Espace pour afficher le menu Intelli-sense :



- Pour une expression BusinessKnowledgeModel, tous les paramètres seront inclus
- Pour l'expression Décision, toutes les décisions requises seront incluses
- Toutes les variables d'entrée de contexte antérieures à la variable actuelle seront incluses (les variables d'entrée de contexte ultérieures à la variable actuelle sont exclues)

Dans cet exemple, en modifiant une expression de contexte encadré BKM, les paramètres d'entrée sont affichés dans le menu Intelli-sense :



Sélection de la langue

Le DMN Modèle peut être généré sous forme de code source en JavaScript , Java, C# ou C++. Comme la syntaxe diffère selon les langues, Enterprise Architect fournit des pages de remplacement de langue pour chaque langue. Si aucun code de remplacement n'est spécifié pour une langue, l'expression définie pour la langue FEEL sera utilisée.

Note : Dans le code généré, l'espace à l'intérieur d'un nom de variable sera remplacé par un trait de soulignement.

Expression littérale

Une expression littérale est la forme la plus simple d'expression DMN ; il est communément défini comme une instruction sur une seule ligne ou un bloc conditionnel if-else. L'expression littérale est un type d'expression valeur utilisée à la fois dans les éléments Décision et dans les éléments Métier Knowledge Modèle (BKM). À mesure que l'expression devient plus complexe, vous préférerez peut-être un contexte encadré ou, afin d'améliorer la lisibilité, vous pouvez encapsuler une partie de la logique sous forme de fonction dans la Bibliothèque DMN.

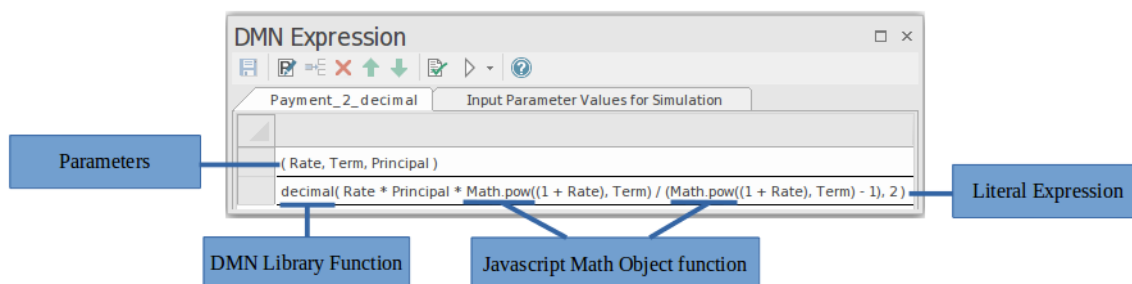
L'icône  dans le coin supérieur droit de l'élément Décision ou BKM indique qu'il est implémenté en tant qu'expression littérale.

Accéder

Diagramme	<p>Sur un diagramme, double-cliquez sur un élément Décision ou BusinessKnowledgeModel.</p> <p>La fenêtre de l'éditeur d'expression DMN s'affiche et affiche les détails de l'élément sélectionné.</p>
-----------	---

Aperçu

Cette image montre la fenêtre de l'éditeur d'expression DMN, telle qu'elle apparaît pour une expression littérale.



L'expression littérale est une représentation textuelle de la logique de décision. Il décrit comment une valeur de sortie est dérivée de ses valeurs d'entrée, à l'aide d'opérations mathématiques et logiques.

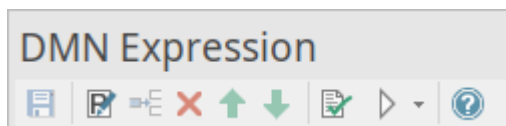
La fenêtre de l'éditeur d'expression présente l'expression littérale sous forme de tableau, avec deux lignes clés :

- Paramètres : définit les paramètres d'entrée utilisés dans l'expression
- Expression littérale : où la formule de l'expression est définie - cela définit le résultat de la Décision

Afin de support la simulation et l'exécution, l'expression littérale peut utiliser des fonctions globales JavaScript ou des fonctions object JavaScript. Les utilisateurs peuvent également créer des fonctions DMN Bibliothèque à utiliser dans les expressions.

Barre d'outils pour l'éditeur d'expression littérale

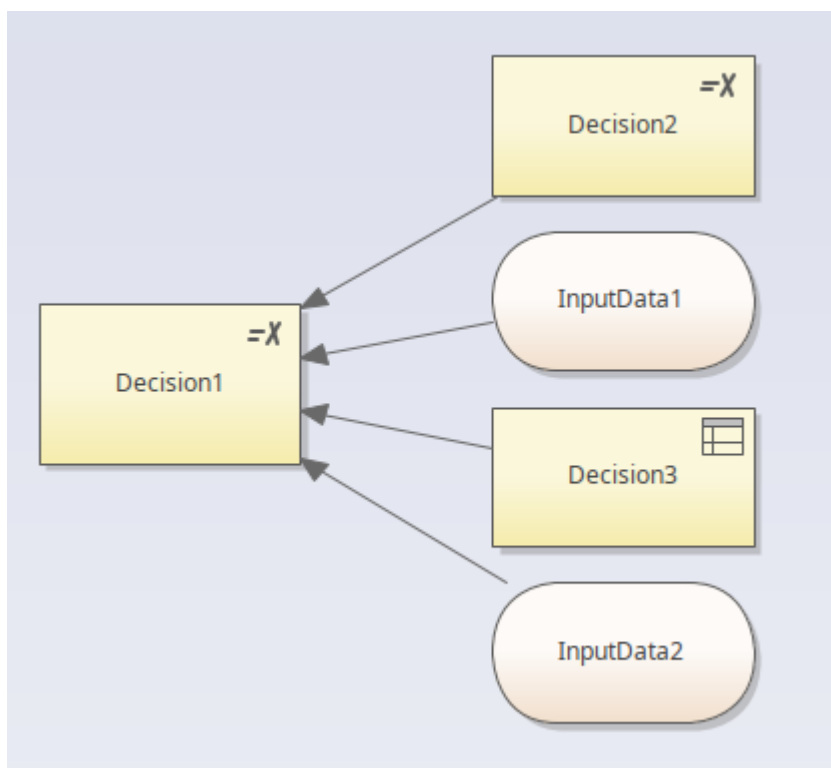
Lorsqu'une expression littérale est sélectionnée, la disposition des fonctionnalités accessibles dans la fenêtre Expression DMN est la suivante :



Pour plus de détails, reportez-vous à la rubrique d'aide *Barre d'outils pour l'éditeur d'expressions littérales*.

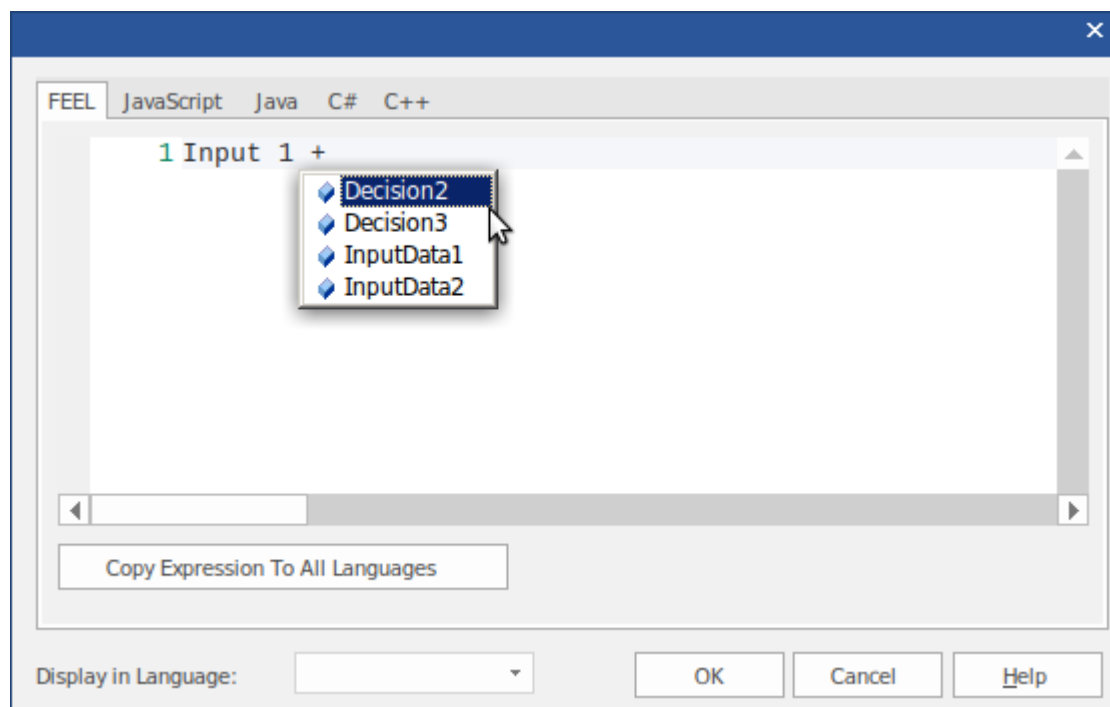
support de l'éditeur d'expression et d'Intelli-sense

Conformément à la spécification du langage Friendly Enough Expression Language (FEEL), les noms de paramètres peuvent contenir des espaces, ce qui facilite la lecture de l'expression. Enterprise Architect fournit également support d'Intelli-sense pour la modification des expressions, permettant une saisie minimale et moins d'erreurs.



Étant donné une hiérarchie de décision telle que celle illustrée, lors de la modification de l'expression de « Décision1 », les entrées de « Décision1 » - à savoir « Décision2 », « Décision3 », « InputData1 » et « InputData2 » - seront disponibles via Intelli-sense. dans l'éditeur.

En cliquant avec le bouton droit sur la ligne « Expression » de la fenêtre Expression DMN, puis en choisissant l'option de menu « Modifier les expressions... », la dialogue de l'éditeur de code d'expression s'affiche. Appuyer sur Ctrl+Espace affiche le menu Intelli-sense :



- Pour les éléments 'Décision', toutes les entrées dans la décision seront affichées
- Pour les éléments Métier Knowledge Modèle (BKM), tous les paramètres d'entrée seront affichés

Le Modèle DMN peut être généré sous forme de code source en JavaScript, Java, C# ou C++ ; Étant donné que certaines langues peuvent avoir une syntaxe différente pour certaines expressions, Enterprise Architect fournit des pages de remplacement de langue pour chaque langue. Si aucun code de remplacement n'est spécifié pour une langue, l'expression définie pour la langue FEEL sera utilisée.









Dans le code généré, l'espace à l'intérieur d'un nom de variable sera remplacé par un trait de soulignement.

Barre d'outils pour l'éditeur d'expression littérale

Lorsqu'une expression littérale est sélectionnée, la fenêtre Expression DMN affiche une barre d'outils spécifique à ce type d'expression.

Options barre d'outils

Ce tableau fournit des descriptions des fonctionnalités accessibles depuis la barre d'outils dans la fenêtre Expression DMN lorsqu'une expression littérale est sélectionnée.

Possibilités	Description
	Cliquez sur ce bouton pour enregistrer la configuration dans la Décision ou le BusinessKnowledgeModel actuel.
	Cliquez sur ce bouton pour modifier les paramètres du Métier Knowledge Modèle .
	Cette option est désactivée pour les expressions littérales.
	Cette option est désactivée pour les expressions littérales.
	Cette option est désactivée pour les expressions littérales.
	Cette option est désactivée pour les expressions littérales.
	Cliquez sur ce bouton pour effectuer la validation de l'Expression Littérale. Enterprise Architect effectuera une série de validations pour vous aider à localiser les erreurs dans l'expression.
	Ce bouton est activé lorsque l'expression littérale est définie pour un élément BusinessKnowledgeModel.

Exemple - Remboursement du prêt

Ce *paiement* Métier Knowledge Modèle (BKM) est mis en œuvre comme une expression littérale.

DMN Expression	
Payment	Input Parameter Values for Simulation
(Rate, Term, Principle)	
decimal(Rate * Principle * Math.pow((1 + Rate), Term) / (Math.pow((1 + Rate), Term) - 1), 2)	

- Le BKM définit trois paramètres : Taux, Durée et Principe

Définissez les valeurs des paramètres d'entrée et évaluez le modèle :

Payment		Input Parameter Values for Simulation
Payment		
Rate : number	0.06 / 12	
Term : number	30 * 12	
Principle : number	300000	

- La valeur du paramètre d'exécution sera affichée ; par exemple, Taux = 00,005
- Le résultat du BKM sera évalué par l'expression littérale et la valeur est affichée sur la ligne de déclaration ; par exemple, retour = 1798,65

Payment		Input Parameter Values for Simulation
(Rate = 0.005, Term = 360, Principle = 300000)return = "1798.65"		
decimal(Rate * Principle * Math.pow((1 + Rate), Term) / (Math.pow((1 + Rate), Term) - 1), 2) (evaluation result = "1798.65")		

Bien que la formule puisse être écrite sur une seule ligne, elle est assez compliquée. Nous pouvons refactoriser ce modèle avec la fonction intégrée et le contexte encadré pour améliorer la lisibilité :

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

- Le contexte encadré définit deux entrées appariées expression-variable ; ces variables servent de « variables locales », qui peuvent être utilisées dans des expressions ultérieures
- valeur de retour : l'expression peut utiliser la valeur des 'variables locales'
- Toutes les expressions dans un contexte encadré peuvent utiliser des fonctions intégrées définies dans la Bibliothèque Gabarit — *Bibliothèque* personnalisable ; par exemple, les fonctions PMT(...) et decimal(...) sont utilisées dans cet exemple

Le résultat de la simulation est exactement le même qu'une expression littérale :

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

Invocation

Une invocation est un conteneur pour les liaisons de paramètres qui fournissent le contexte pour l'évaluation du corps d'un Métier Knowledge Modèle . Il existe deux cas d'utilisation courants pour une invocation :

- Lier une donnée d'entrée au Métier Knowledge Modèle
- Lier des paramètres ou des variables d'entrée de contexte au Métier Knowledge Modèle

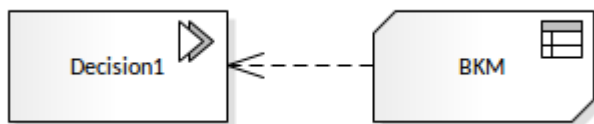
Un exemple de chacun est fourni dans les sous-rubriques de cette rubrique d'aide.

Accéder

Diagramme	Double-cliquez sur l'élément Décision ou l'élément BKM approprié. La fenêtre Expression DMN s'affiche, affichant les détails de l'élément sélectionné.
-----------	---

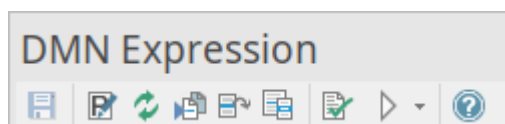
Aperçu

Une Invocation est un type d'expression valeur applicable à la fois aux éléments Décision et aux éléments Métier Knowledge Modèle . Il s'agit d'une représentation tabulaire de la façon dont la logique de décision définie au sein d'un élément invocable (un Métier Knowledge Modèle ou un Décision Service) est invoquée par une Décision ou par un autre Métier Knowledge Modèle .



Barre d'outils pour l'éditeur d'appels

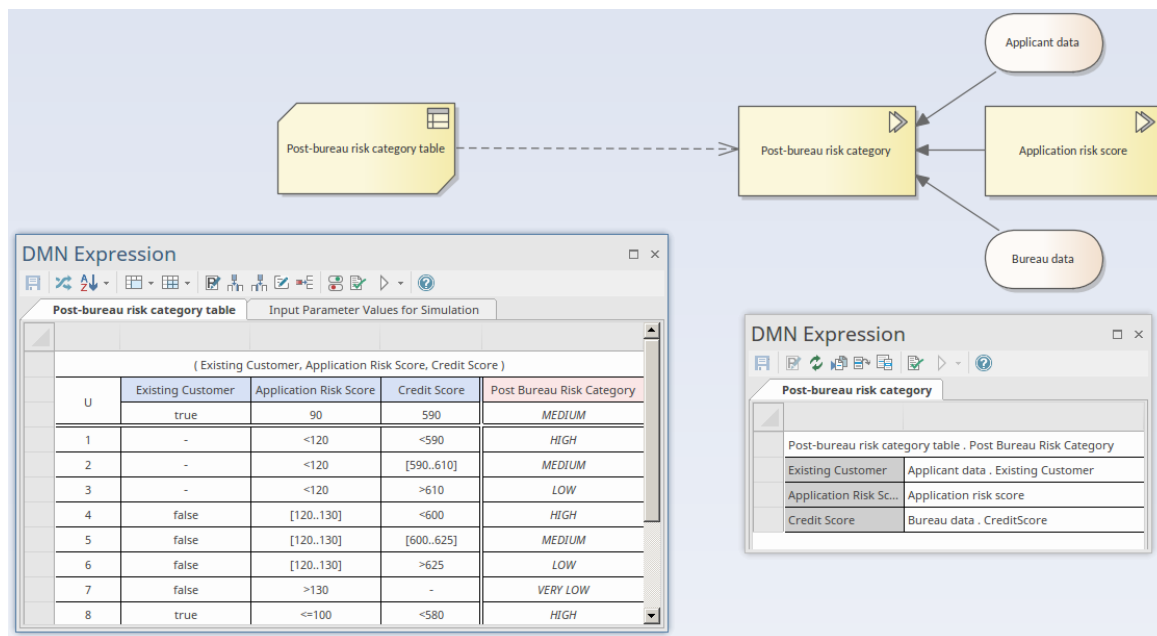
Lorsqu'une Invocation est sélectionnée, un certain nombre de facilités pour travailler dessus sont accessibles depuis la barre d'outils de la fenêtre Expression DMN :



Pour plus de détails, reportez-vous à la rubrique d'aide « Barre d'outils pour l'éditeur d'appels ».

Reliures

Les liaisons de paramètres d'une invocation fournissent le contexte pour l'évaluation du corps de l'élément invocable.

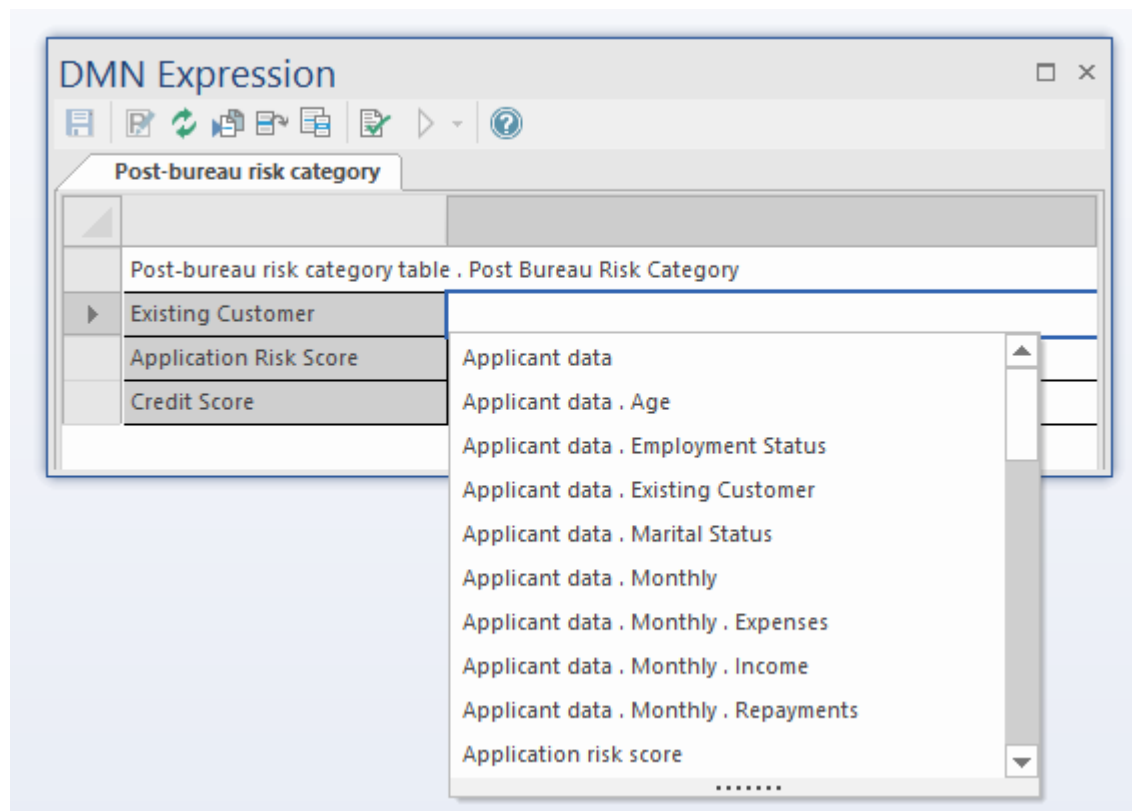


Dans cet exemple :

- La Décision 'Catégorie de risque post-bureau' est représentée comme une Invocation se connectant au Métier Knowledge Modèle 'tableau des catégories de risque post-bureau', implémenté sous forme de Tableau de Décision
- La Décision 'Catégorie de risque post-bureau' est la cible de trois connecteurs d'exigences d'informations à partir de deux éléments de données d'entrée et d'un élément Décision .
- La liste de liaison lie les valeurs d'entrée aux paramètres du Métier Knowledge Modèle
- L'invocation spécifie également la « OutputClause » demandée ; dans le cas où un Tableau de Décision a plusieurs Clauses de Sortie définies, l'Invocation doit demander explicitement une Clause de Sortie comme résultat de l'expression

Contributions

Les entrées d'autres éléments Decisions et InputData peuvent être définies en appuyant sur la barre d'espace dans le champ :



Sortir

Comme une Invocation ne peut invoquer qu'un seul Métier Knowledge Modèle , la sortie est définie par la sortie Métier Knowledge Modèle .

Barre d'outils pour l'éditeur d'appels

Lorsqu'une expression d'appel est sélectionnée, la barre d'outils de la fenêtre Expression DMN fournit des options spécifiques à ce type d'expression.

Options barre d'outils

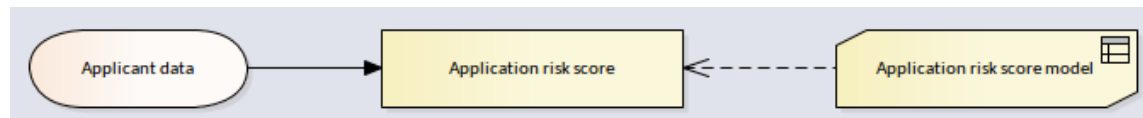
Ce tableau fournit des descriptions des fonctionnalités accessibles dans la fenêtre Expression DMN lorsqu'une Invocation est sélectionnée.

Possibilités	Description
	Cliquez sur ce bouton pour enregistrer la configuration dans la Décision ou le BusinessKnowledgeModel actuel.
	Cliquez sur ce bouton pour modifier les paramètres du Métier Knowledge Modèle .
	Applicable aux expressions valeur d'Invocation, pour les éléments Décision et les éléments Métier Knowledge Modèle (BKM). Cliquez sur ce bouton pour synchroniser avec le BKM invoqué. Par exemple, si le BKM change de nom, de paramètres, de sorties ou de types, cliquez sur ce bouton pour synchroniser ces modifications.
	Applicable aux expressions valeur d'Invocation, pour les éléments Décision et les éléments Métier Knowledge Modèle (BKM). Cliquez sur ce bouton pour définir ou modifier un BKM comme une invocation.
	Applicable aux expressions valeur d'Invocation, pour les éléments Décision et les éléments Métier Knowledge Modèle (BKM). Cliquez sur ce bouton pour ouvrir le BKM invoqué dans la fenêtre DMN Expression.
	Applicable aux expressions valeur d'Invocation, pour les éléments Décision et les éléments Métier Knowledge Modèle (BKM). Lorsqu'un BKM est implémenté en tant que Tableau de Décision , il peut définir plusieurs clauses de sortie ; l'invocation sur ce BKM devra peut-être spécifier quelle sortie est demandée. Cliquez sur ce bouton pour lister toutes les sorties disponibles dans un menu contextuel ; la sortie actuellement configurée est vérifiée.
	Effectuer la validation de l'invocation. Enterprise Architect effectuera une série de validations pour vous aider à localiser les erreurs dans la définition d'appel.
	Ce bouton est activé lorsque l'Invocation est définie pour un Métier Knowledge Modèle . Sélectionnez l'onglet 'Valeurs des paramètres d'entrée pour Simulation ', complétez les champs et cliquez sur ce bouton. Le résultat du test sera présenté sur le Tableau de Décision , avec les valeurs d'exécution des entrées et des sorties affichées et les règles valides mises en évidence. Vous pouvez utiliser cette fonctionnalité pour tester unitairement un Métier Knowledge Modèle sans connaître le contexte et invoqué ultérieurement par une

	<p>Décision ou un autre Métier Knowledge Modèle .</p> <p>Des options de menu sont disponibles pour ce bouton de la barre d'outils. Pour plus d'informations, consultez la rubrique d'aide <i>de Simulate DMN Modèle</i> .</p>
--	---

Exemple 1 - Lier les données d'entrée au Modèle de connaissances Métier

Un exemple complet peut être créé avec un Modèle Motif (dans le ruban, sélectionnez 'Simulate > Décision Analysis > DMN > Appliquer Perspective > DMN Décision > Décision With BKM : Create Modèle (s)').



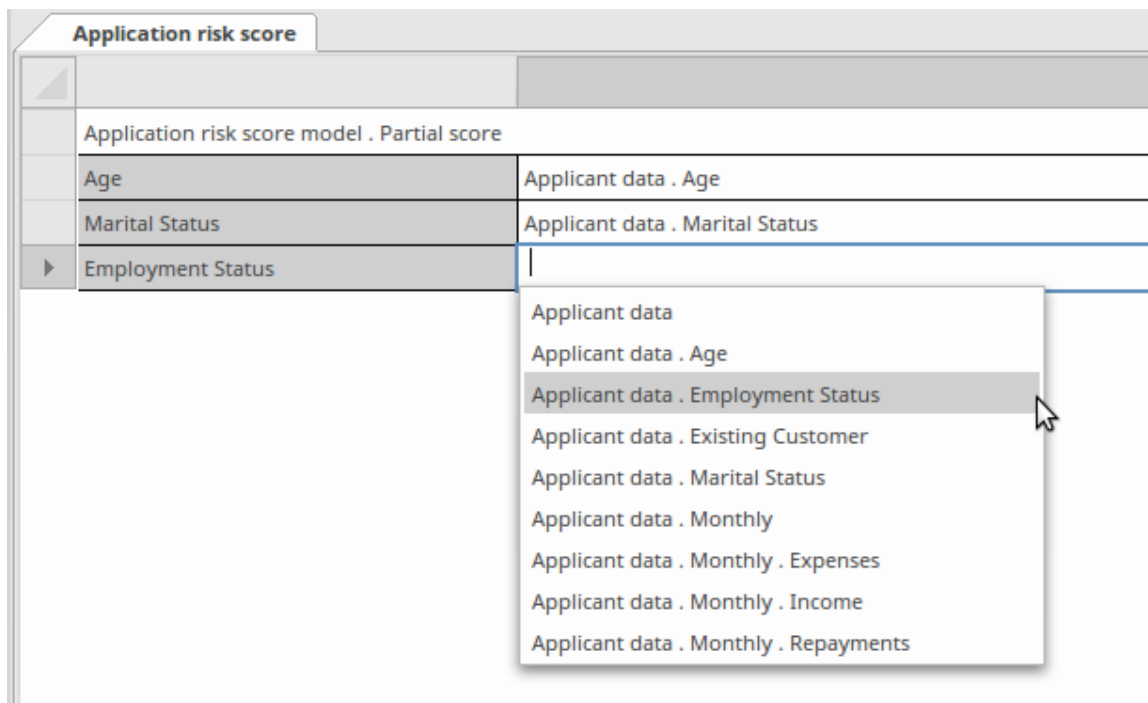
Dans cet exemple, les données d'entrée des données du candidat sont saisies dans la définition des données du candidat, qui comporte trois composants.

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

Le modèle de score de risque de l'application Métier Knowledge Modèle est implémenté sous la forme d'un Tableau de Décision avec trois entrées et une sortie.

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

Le score de risque de l'application Décision est implémenté sous forme d'invocation pour lier les composants « feuille » des données d'entrée aux paramètres du BKM.

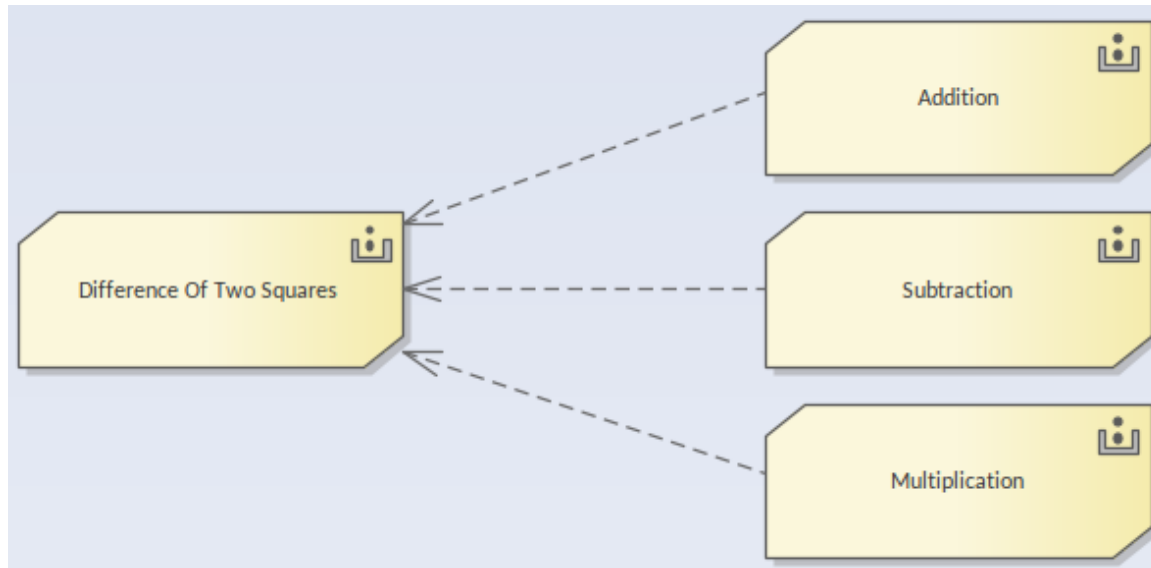


Afin de faciliter la liaison, la saisie semi-automatique est prise en charge pour l'expression de liaison.

Les instructions complètes modélisation et de simulation sont disponibles dans la documentation de Motif .

Exemple 2 - Lier des variables d'entrée de contexte à Métier Knowledge Modèle

Un exemple complet peut être créé avec un Modèle Motif (dans le ruban, sélectionnez 'Simulate > Décision Analysis > DMN > Appliquer Perspective > DMN Métier Knowledge Modèle Exemples > Métier Knowledge Modèle Invocation : Create Modèle (s)).



Dans cet exemple, la *différence de deux carrés* Métier Knowledge Modèle (BKM) est implémentée sous forme de contexte encadré :

- La *somme variable de ab* est implémentée sous forme d'invocation en liant les paramètres *a* et *b* à l'*addition* BKM
- La *différence variable de ab* est implémentée sous forme d'invocation en liant les paramètres *a* et *b* à la *soustraction* BKM.
- La *différence variable des carrés* est implémentée sous forme d'invocation en liant les variables locales *somme de ab* et *différence de ab* à la *multiplication* BKM.

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

Afin de faciliter la liaison, la saisie semi-automatique est prise en charge pour l'expression de liaison.

Les instructions complètes modélisation et de simulation sont disponibles dans la documentation de Motif .

Achèvement automatique des expressions DMN

DMN définit de nombreuses expressions, telles que FunctionDefinition, DecisionTable, Boxed Context, Invocation et Literal Expression. Les paramètres, les arguments et la logique de ces expressions sont largement implémentés par le texte.

Pour rendre modélisation simple et fiable, Enterprise Architect fournit une complétion automatique facilitée, aidant à fournir :

- Valeurs autorisées de ItemDefinition
- Entrées d'entrée/sortie d'un Tableau de Décision
- Informations requises

Valeurs autorisées de ItemDefinition

L'idée est de définir les énumérations valeur autorisées dans ItemDefinition, puis de composer une liste de sélection chaque fois que ces valeurs sont demandées.

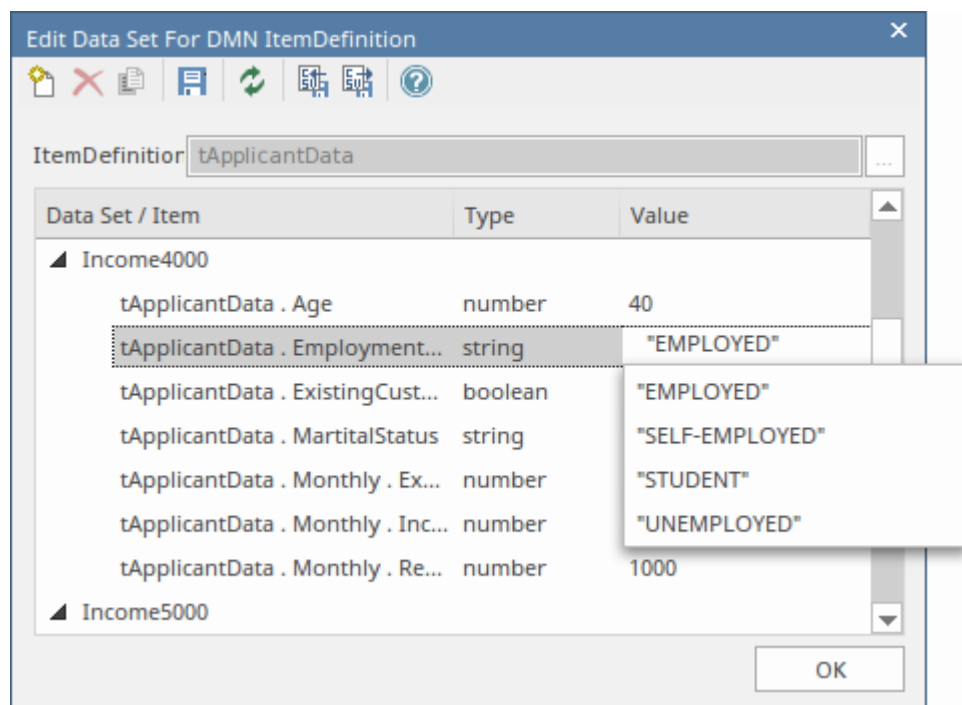
Dans cet exemple, ItemDefinition 'Données du candidat'. Le statut d'emploi définit une énumération de valeurs autorisées.

Item Name	Data Type	Allowed Values
tApplicantData	Age : number	Type in Allowed Value Enumerations...
tApplicantData	MaritalStatus : string	"S", "M"
tApplicantData	EmploymentStatus : string	"EMPLOYED", "SELF-EMPLOYED", "STUDENT", "UNEMPLOYED"
tApplicantData	ExistingCustomer : boolean	true, false
tApplicantData	Monthly	Income : number Repayments : number Expenses : number

Lors de la modification des valeurs pour les données d'entrée saisies dans cette définition d'élément, appuyez sur la barre d'espace du clavier pour afficher une liste de valeurs parmi lesquelles sélectionner.

Item Name	Data Type	Allowed Values
tApplicantData	Age : number	25
tApplicantData	MaritalStatus : string	"M"
tApplicantData	EmploymentStatus : string	"e"
tApplicantData	ExistingCustomer : boolean	"EMPLOYED", "SELF-EMPLOYED", "STUDENT", "UNEMPLOYED"
tApplicantData	Monthly	60000, 20, 3000

Nous pourrions également définir plusieurs ensembles de données pour InputData, car la fonctionnalité de complétion automatique est disponible dans cette dialogue .



Entrées d'entrée/sortie d'un Tableau de Décision

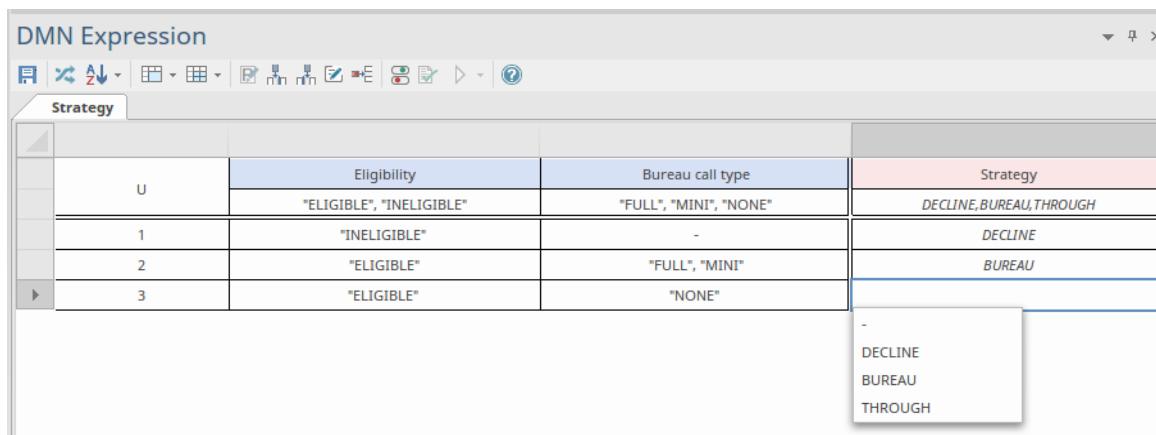
Prenons l'exemple de la définition d'élément « Stratégie » :

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

Nous pouvons rapidement remplir le champ 'Valeurs autorisées' d'un Tableau de Décision par sélection :

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

On peut alors remplir rapidement les règles Tableau de Décision par sélection :



DMN Expression

Strategy

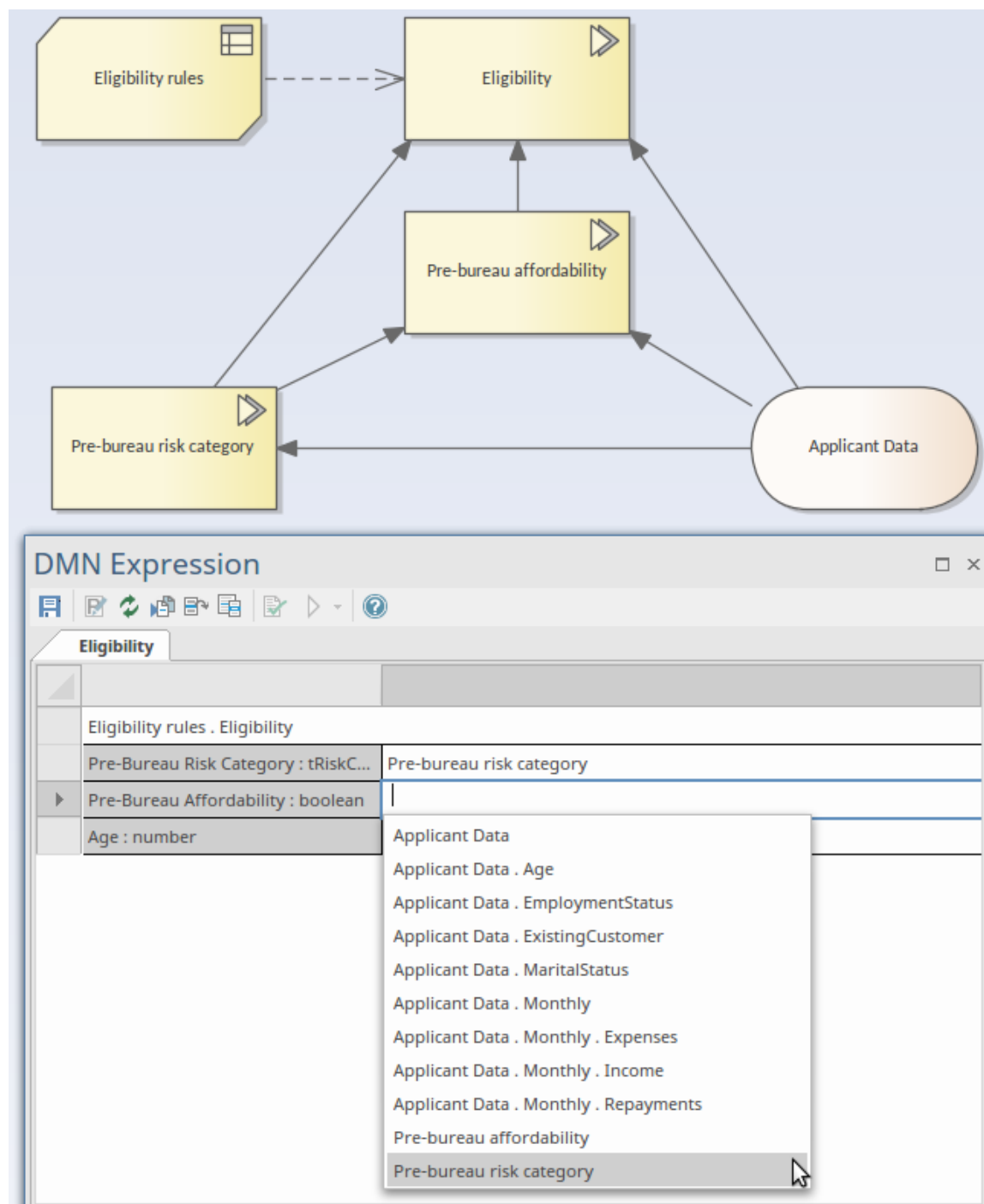
	Eligibility	Bureau call type	Strategy
U	"ELIGIBLE", "INELIGIBLE"	"FULL", "MINI", "NONE"	DECLINE, BUREAU, THROUGH
1	"INELIGIBLE"	-	DECLINE
2	"ELIGIBLE"	"FULL", "MINI"	BUREAU
3	"ELIGIBLE"	"NONE"	-

-
DECLINE
BUREAU
THROUGH

Note : le '-' par défaut signifie 'Indéfini'.

Exigence d'informations

Sur une hiérarchie de décision, une décision peut accéder aux décisions requises et aux données d'entrée ; ces éléments requis forment une liste de variables pouvant être utilisées par la décision.



Dans cet exemple, Décision « Éligibilité » nécessite deux décisions – « Catégorie de risque pré-bureau » et « Accessibilité pré-bureau » - et un élément de données d'entrée « Données du candidat ».

Lors de la définition des valeurs de liaison pour les « règles d'éligibilité » du BusinessKnowledgeModel invoquées, une liste de saisie semi-automatique vous prompt à effectuer une sélection. Dans cette liste, il y a des noms de sous-décisions - des composants feuilles des données d'entrée. Avec cette fonctionnalité, vous pouvez facilement configurer une invocation.

Validation des expressions DMN

DMN définit de nombreuses expressions, telles que FunctionDefinition, DecisionTable, Boxed Context, Invocation et Literal Expression. Les paramètres, les arguments et la logique de ces expressions sont largement implémentés par le « texte ».

Pour rendre modélisation plus simple et plus fiable, Enterprise Architect propose deux fonctionnalités : Auto Complétion et Validation.

- Validation : identifie les erreurs modélisation causées par des fautes de frappe, une logique incomplète, une incohérence, etc.
- Complétion automatique : vous pouvez sélectionner une string de texte dans une liste d'énumérations plutôt que de saisir le texte dans

Dans cette rubrique, nous allons vous montrer comment valider une expression DMN.

Accéder

Fenêtre d'expression DMN	Simuler > Analyse Décision > DMN > Expression DMN : Bouton Valider
Fenêtre Simulation DMN	Simuler > Analyse Décision > DMN > Ouvrir Simulation DMN > Simuler : icône Valider

Validations communes

Validation du nom de variable

The screenshot shows the 'DMN Expression' window with the 'BKM1' tab selected. The 'Input Parameter Values for Simulation' section contains a table with the following entries:

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

Below the table, the 'System Output' window shows the results of the validation:

```

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

Dans cet exemple, le Boxed Context Métier Knowledge Modèle BKM1 définit deux paramètres, « Entrée 1 » et « Entrée 2 », et deux variables locales, « Variable locale 1 » et « Variable locale 2 ». L'expression a été validée et les résultats sont affichés dans l'onglet « Validation DMN » de la fenêtre Sortie système.

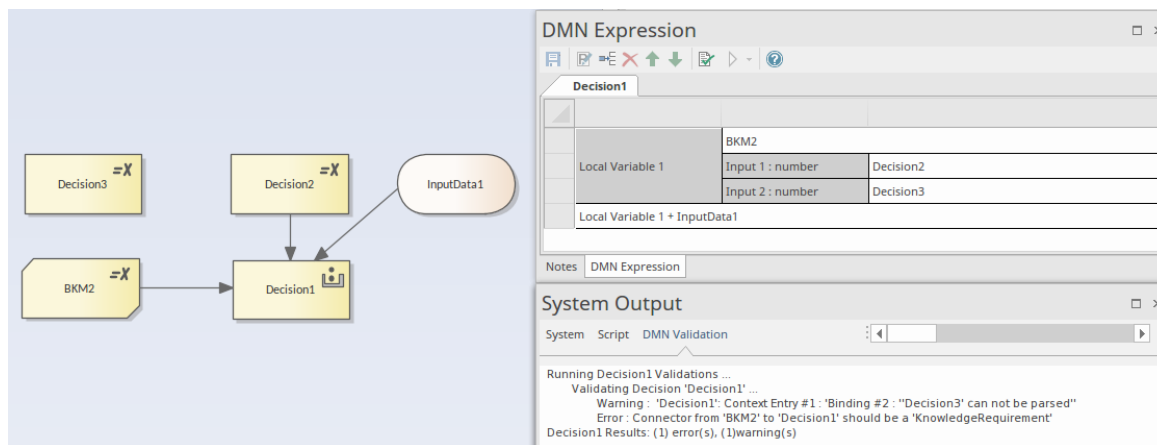
- L'entrée de contexte n° 1 a échoué en raison d'une erreur typographique ; il devrait s'agir de l'opérateur '-', mais l'utilisateur a tapé ou copié '-'
- L'entrée de contexte n°2 a échoué car il n'y a pas d'espace entre « Entrée » et le chiffre 2 ; note que la fonction 'ceiling()' est définie dans la Bibliothèque DMN afin qu'elle puisse être analysée avec succès
- L'entrée de contexte n° 3 a échoué car il n'y a pas d'espace entre "Local" et "Variable".

Il est difficile d'identifier visuellement ce type d'erreur. L'exécution de la validation peut aider à identifier les erreurs et vous pouvez ensuite facilement effectuer une correction.

Validation des dépendances

Une décision peut nécessiter d'autres décisions, données d'entrée et modèles de connaissances métier ; ces relations sont identifiées par les connecteurs InformationRequirement et KnowledgeRequirement.

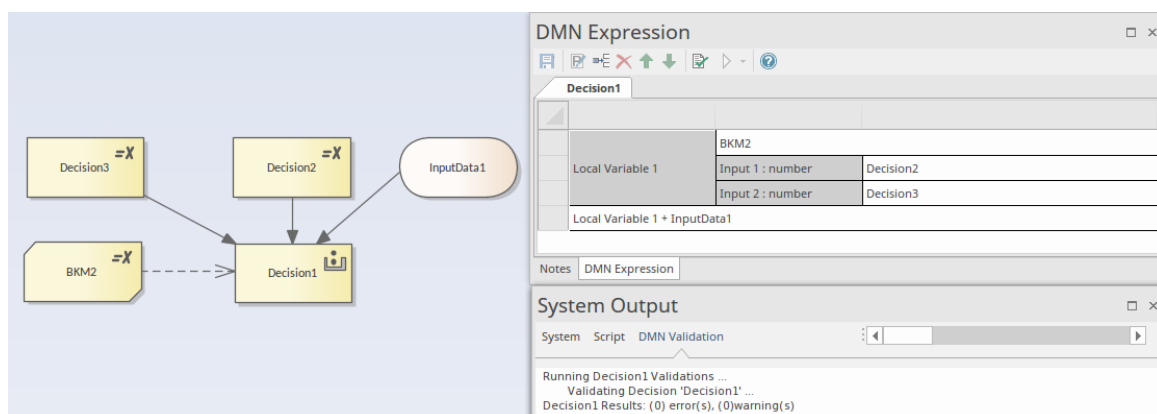
Lorsque le graphique devient complexe, il est fort possible que certains connecteurs soient manquants ou que le mauvais type de connecteur soit utilisé.



Dans cet exemple, cliquez sur le bouton Valider, Enterprise Architect affichera que :

- « Décision3 » est utilisée par « Décision1 » en se liant à un paramètre du BKM2 appelé ; cependant, il n'est pas défini - un connecteur InformationRequirement est manquant
- L'invocation définie dans « Décision 1 » n'est pas valide ; le type de connecteur de « BKM2 » à « Décision1 » doit être une exigence de connaissances

Après avoir résolu ces problèmes, exécuter à nouveau la validation :




Contexte encadré

Un contexte encadré est une collection d'entrées de contexte, présentées sous la forme d'un tableau, suivies d'une expression de résultat final.

Ces entrées de contexte consistent en une variable associée à une expression valeur et peuvent être considérées comme des résultats intermédiaires. Cela permet de décomposer des expressions complexes en une série d'expressions simples, le résultat final étant évalué sous une forme beaucoup plus simple.

Le type Boxed Context est pris en charge dans les types d'éléments *Décision* et *Métier Knowledge Modèle*. Il est indiqué

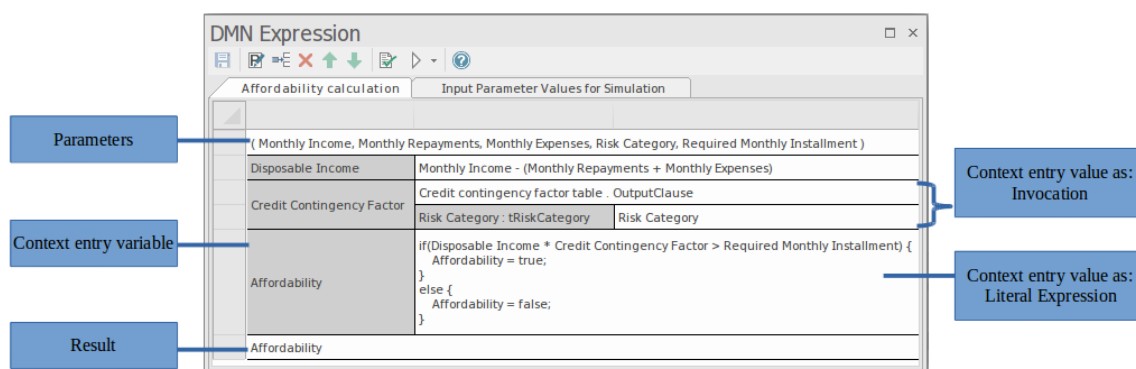
par l'icône .

Accéder

Diagramme	Sur un diagramme, double-cliquez sur un élément Décision ou un élément BKM. La fenêtre de l'éditeur d'expression DMN s'affiche, affichant les détails de l'élément sélectionné.
-----------	---

Aperçu

Cette image montre la fenêtre de l'éditeur d'expression DMN telle qu'elle apparaît pour une expression de contexte encadré.

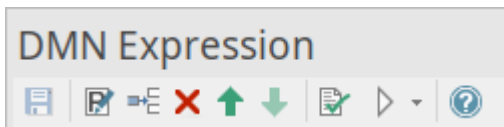


Un contexte encadré est une collection d'entrées de contexte, présentées sous la forme d'un tableau, suivies d'une expression de résultat final. Chaque entrée de contexte se compose d'une variable et d'une expression valeur. La variable peut être considérée comme un résultat intermédiaire et elle peut être utilisée dans l'expression valeur de toute entrée de contexte ultérieure. L'expression valeur d'une entrée de contexte peut être soit une expression littérale, soit une invocation, et peut utiliser toutes les entrées disponibles telles que des paramètres (d'un élément BKM), des données d'entrée ou des résultats de décision, ainsi que toutes les variables de contexte précédemment définies.

Le résultat final d'une expression de contexte encadré est déterminé en parcourant chaque entrée de contexte tour à tour, en évaluant l'expression valeur et en attribuant son résultat à la variable, puis en évaluant finalement l'expression de résultat. L'expression de résultat peut également utiliser n'importe quelle entrée ou variable locale, mais doit être évaluée pour fournir un résultat.

Barre d'outils pour l'éditeur de contexte encadré

Lorsqu'une expression Boxed Context est sélectionnée, la disposition des fonctionnalités accessibles dans la fenêtre DMN Expression est la suivante :

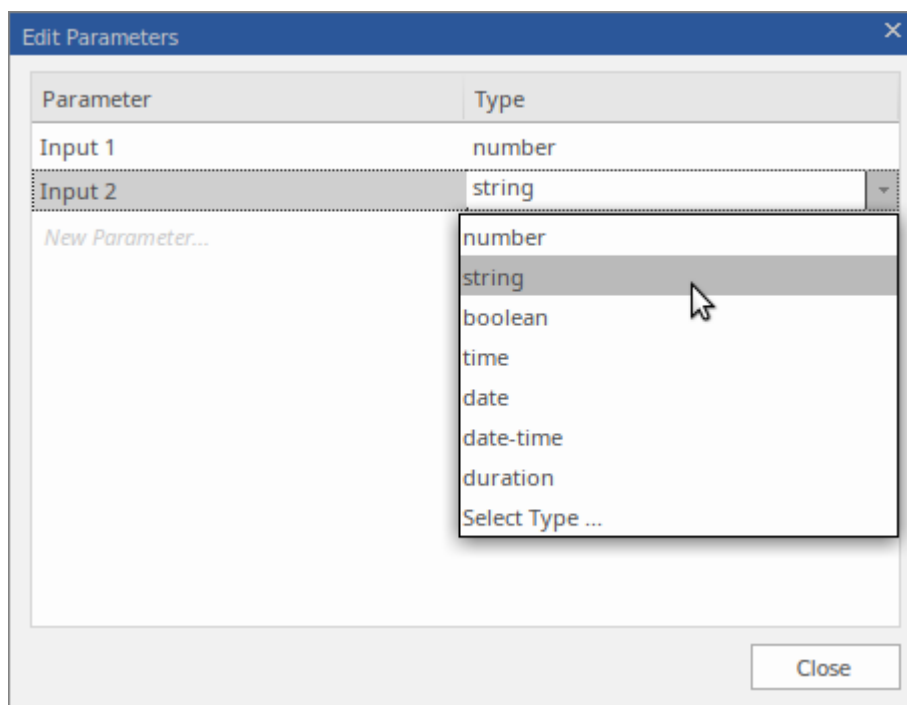


Pour plus de détails, reportez-vous à la rubrique d'aide « Barre d'outils pour l'éditeur de contexte encadré ».

Spécification des paramètres

Dans le cas des éléments BusinessKnowledgeModel, les paramètres sont utilisés pour transmettre les valeurs d'entrée fournies par l'élément appelant. La logique de décision du BKM est évaluée à l'aide des paramètres d'entrée et le résultat est renvoyé à l'élément appelant. Par défaut, un élément BKM est créé avec deux paramètres d'entrée, « Entrée 1 » et « Entrée 2 ».

Cliquez sur l'icône  dans la barre d'outils de la fenêtre Expression DMN pour afficher la fenêtre « Modifier les paramètres ».



Ici, vous pouvez modifier les noms des paramètres, définir leurs types de données, créer des paramètres supplémentaires ou supprimer ceux existants.

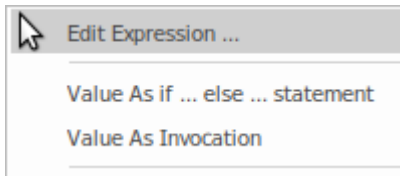
Spécification des entrées de contexte

Chaque entrée de contexte se compose d'une paire variable-expression.

Le nom de la variable peut être n'importe quel texte de votre choix et peut même contenir des espaces. Pour modifier le nom de la variable, cliquez sur la cellule pour la sélectionner, puis cliquez à nouveau ou appuyez sur F2 pour passer en mode édition. Pour quitter le mode édition, cliquez ailleurs ou appuyez sur la touche Entrée.

En général, il n'est pas nécessaire de spécifier un type de données pour l'expression ou les variables - le type sera déduit de la valeur. Cependant, si vous avez l'intention de générer du code pour des langages compilés tels que Java, C++ ou C#, vous devrez spécifier le type de toutes les variables d'entrée de contexte.

L'expression valeur d'une entrée de contexte peut être soit une expression littérale, soit une invocation et peut utiliser toutes les entrées disponibles, telles que des paramètres (d'un élément Métier Knowledge Modèle), des données d'entrée ou des résultats de décision, ainsi que toutes variables de contexte préalablement définies. . Un clic droit sur la cellule d'expression affiche un menu contextuel qui propose des options pour afficher un éditeur de code d'expression ou pour définir l'expression valeur en tant qu'instruction If-Else ou Invocation.



Vous pouvez également modifier l'expression valeur en saisissant du texte directement dans la cellule d'expression.









Pour plus d'informations sur la façon de spécifier des expressions littérales ou des invocations, veuillez consulter les rubriques d'aide traitant de ces sujets.

Barre d'outils pour l'éditeur de contexte encadré

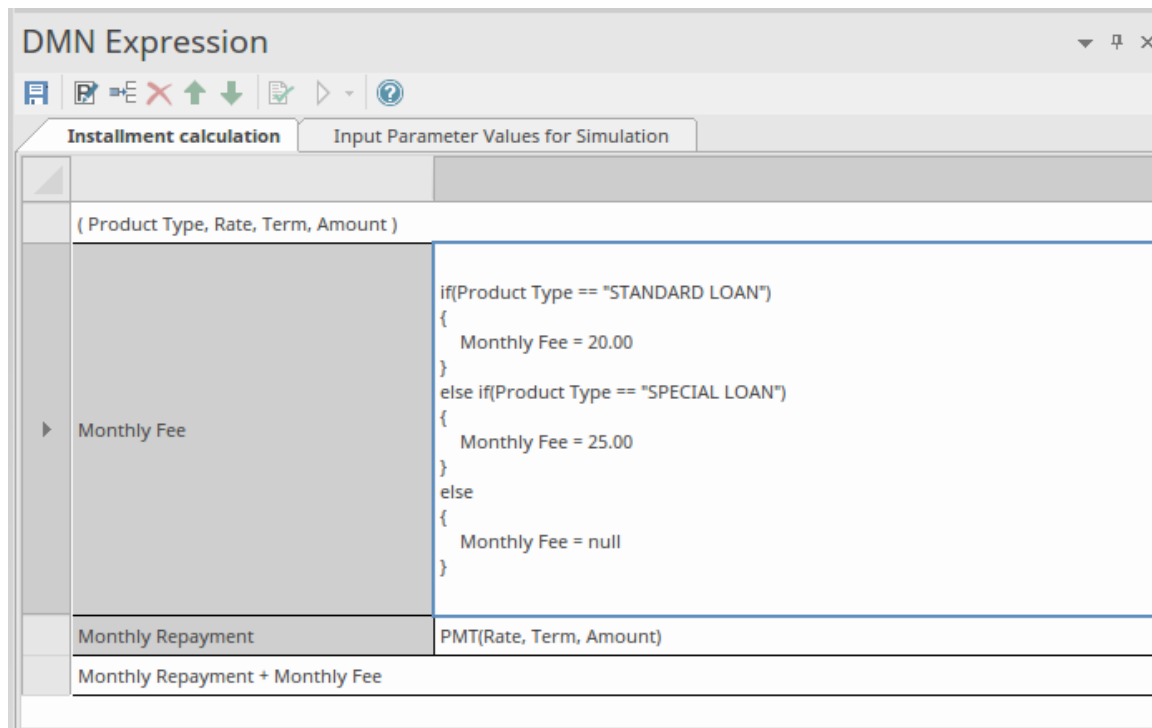
Ce tableau fournit des descriptions des fonctionnalités accessibles dans la fenêtre Expression DMN lorsqu'un contexte encadré est sélectionné.

Options barre d'outils

Cette barre d'outils est destinée au contexte encadré.

Possibilités	Description
	Enregistrez les modifications apportées à l'élément Décision ou BusinessKnowledgeModel actuellement sélectionné.
	Affichez la fenêtre « Modifier les paramètres », dans laquelle vous pouvez spécifier le nom et le type de données de chaque paramètre transmis lors de l'appel de la logique de décision d'un élément BusinessKnowledgeModel.
	Créez une nouvelle entrée de contexte et ajoutez-la à la liste des entrées de contexte.
	Supprimez l'entrée de contexte actuellement sélectionnée.
	Déplacez l'entrée contextuelle actuellement sélectionnée d'une position vers le haut dans la liste.
	Déplace l'entrée de contexte actuellement sélectionnée vers le bas d'une position dans la liste.
	Effectuez la validation du BoxedContext. Enterprise Architect effectuera une série de validations pour vous aider à découvrir les erreurs dans la définition BoxedContext.
	<p>Ce bouton est activé lorsqu'un Tableau de Décision est défini pour un élément Métier Knowledge Modèle .</p> <p>Sélectionnez l'onglet 'Valeurs des paramètres d'entrée pour Simulation ', complétez les champs, puis cliquez sur ce bouton. Le résultat du test sera présenté sur le Tableau de Décision , avec les valeurs d'exécution des entrées et des sorties affichées et les règles valides mises en évidence.</p> <p>Vous pouvez utiliser cette fonctionnalité pour tester unitairement un élément BusinessKnowledgeModel, sans spécifier son contexte.</p> <p>Un certain nombre d'options de menu sont disponibles pour ce bouton de la barre d'outils. Pour plus d'informations, consultez la rubrique d'aide <i>Simuler DMN Modèle</i> .</p>

Exemple - Calcul des mensualités du prêt



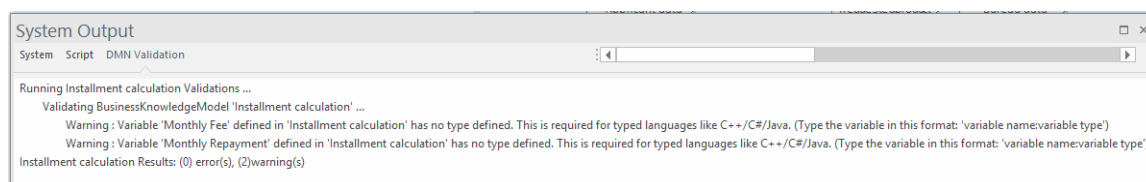
Le calcul des versements Métier Knowledge Modèle (BKM) est implémenté en tant que contexte encadré.

- Le BKM définit quatre paramètres : Type de produit, Taux, Durée et Montant.
- Le contexte encadré définit deux entrées de paire variable-expression ; ces variables servent de « variables locales » qui peuvent être utilisées dans des expressions ultérieures
- valeur de retour : L'expression peut utiliser la valeur des 'variables locales'
- Toutes les expressions dans un contexte encadré peuvent utiliser des fonctions intégrées, qui sont définies dans la *Bibliothèque Gabarit* — DMN personnalisable ; les fonctions PMT(...) et decimal(...) sont utilisées dans cet exemple

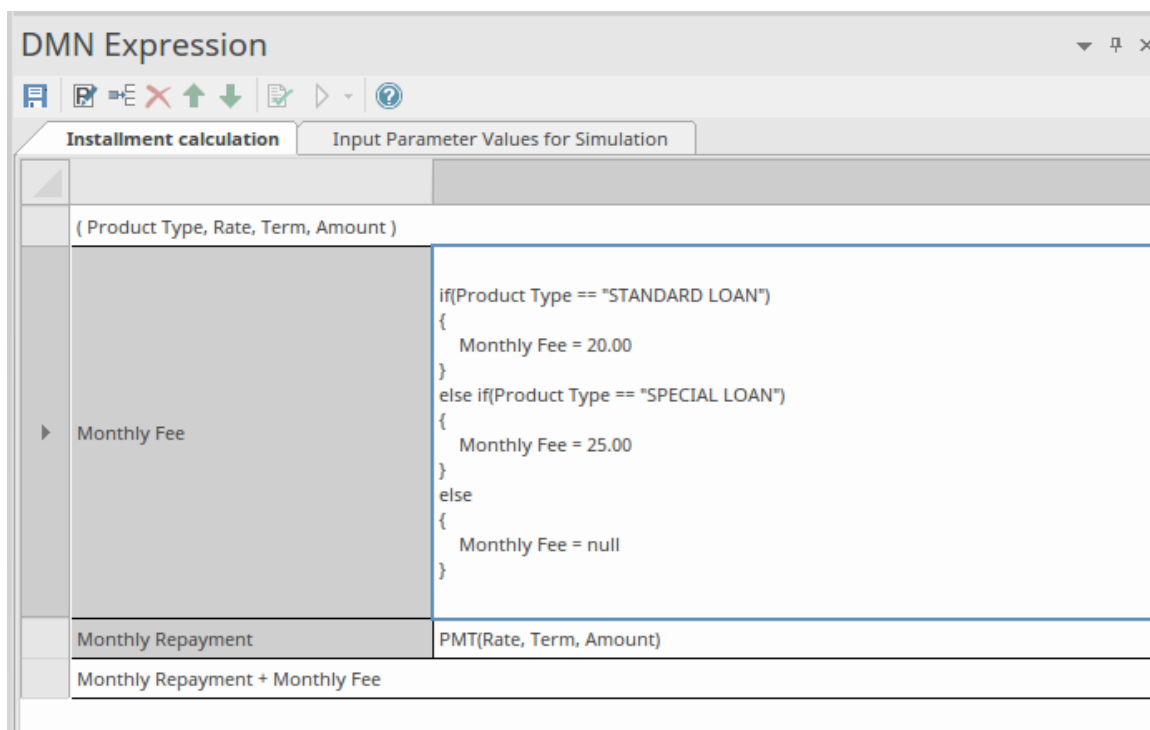
Spécifier Type de variable d'entrée de contexte

En général, l'expression et les variables n'ont pas besoin de spécifier un type, qui est déduit de la valeur fournie. Cette fonctionnalité est prise en charge de manière générique par JavaScript, qui est utilisé pour Simulation DMN d'Enterprise Architect.

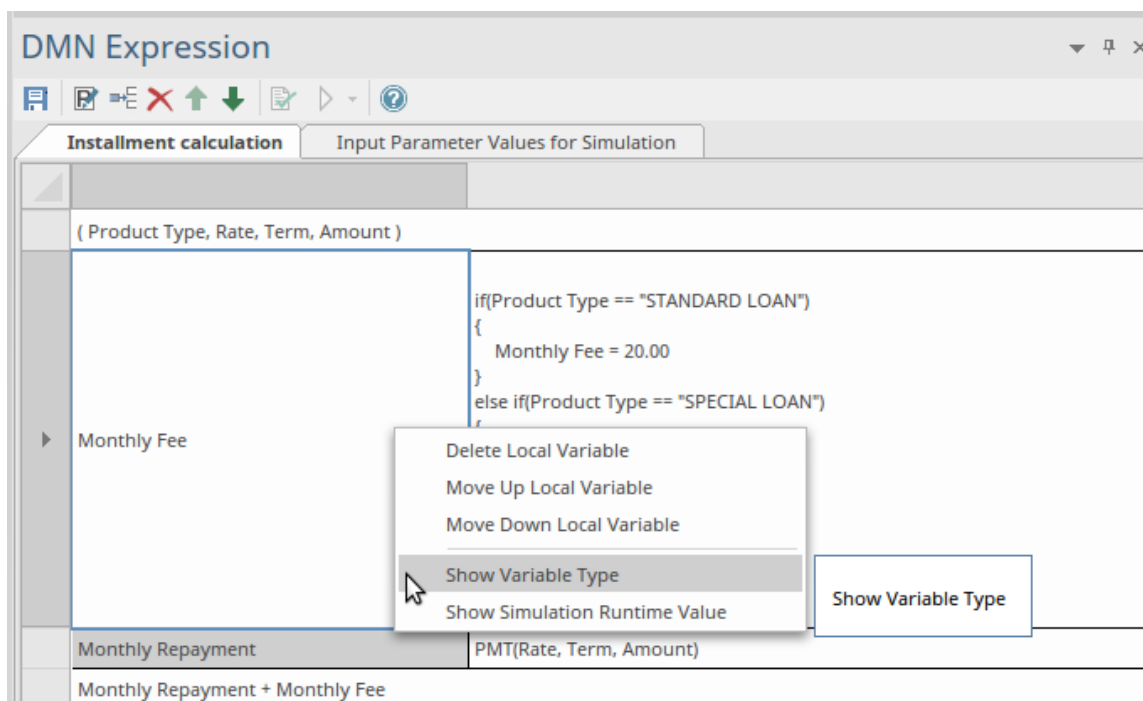
Cependant, si vous souhaitez générer du code à partir d'un modèle DMN vers des langages compilés tels que Java, C++ ou C#, vous devrez spécifier le type pour chaque variable d'entrée de contexte. Sinon, si vous validez le modèle, vous verrez des avertissements tels que :



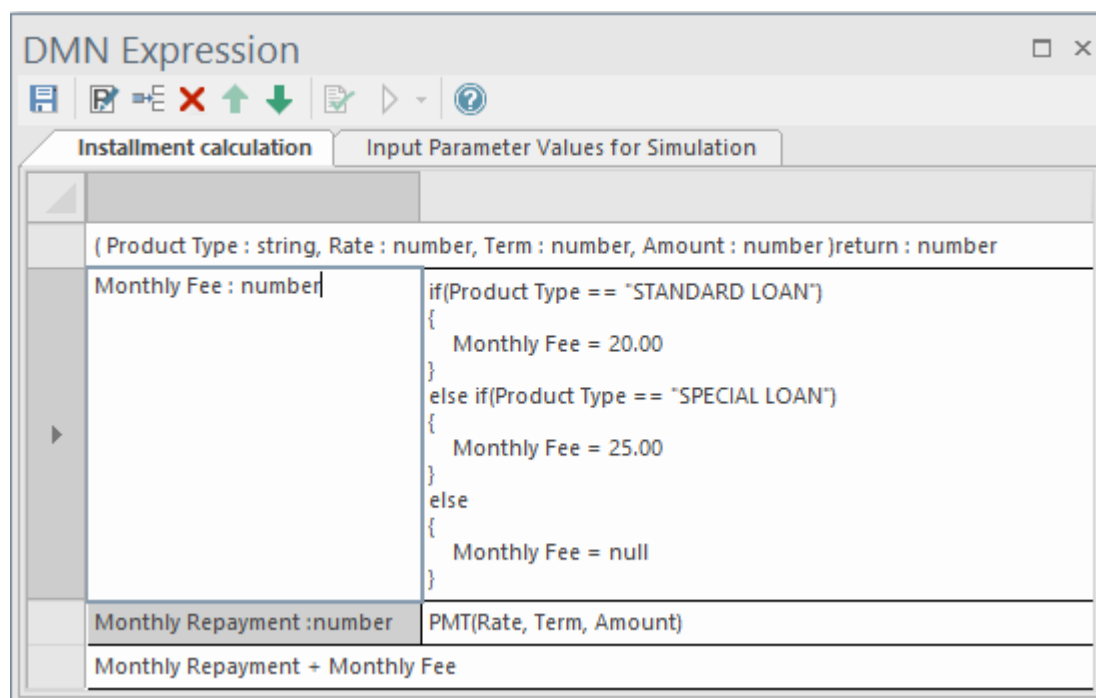
Cliquez-droit sur la variable de saisie de contexte (frais mensuels, remboursement mensuel) dans ce modèle.




Sélectionnez l'option 'Afficher Type de variable'.



Tapez maintenant le type de variable, en l'ajoutant au nom de la variable et en le séparant par deux points, comme indiqué ici.

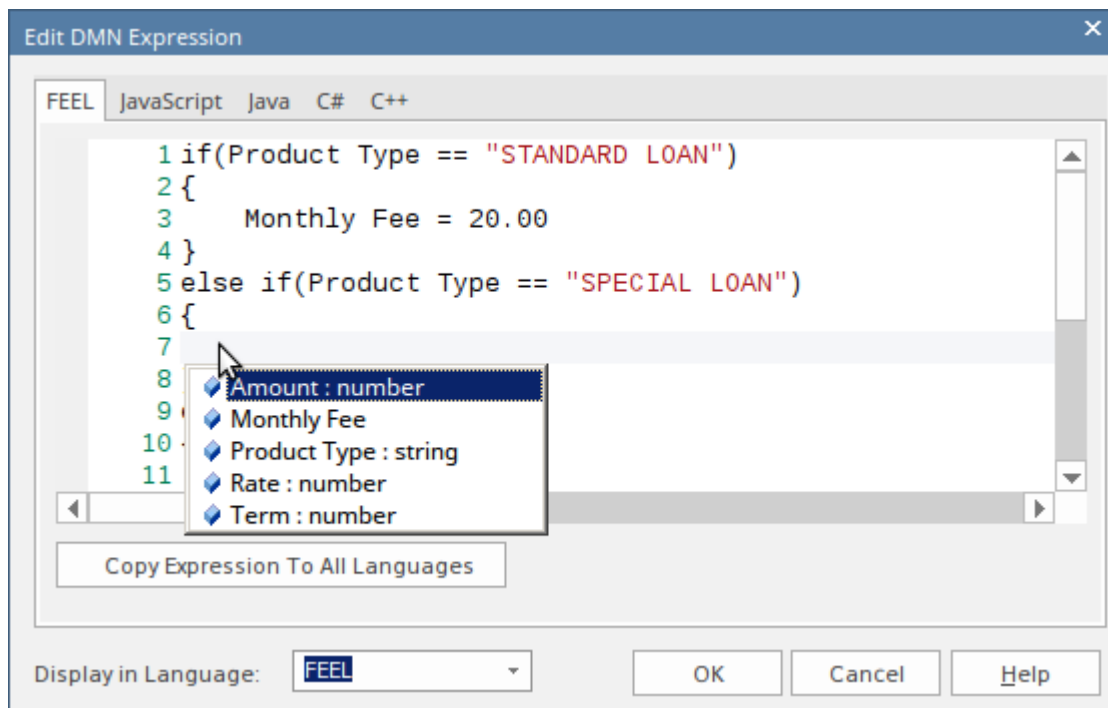


Cliquez ensuite sur le bouton Enregistrer de la barre d'outils pour enregistrer l'expression, puis cliquez sur le bouton  pour valider à nouveau le modèle.

Éditeur d'expression et Support d'Intelli-sense

Le paramètre et le nom de la variable de Context Entry peuvent contenir des espaces, selon la spécification du langage FEEL. Cette fonctionnalité rend l'expression facile à lire. Afin de vous aider à modifier les expressions avec moins de saisie et en faisant moins d'erreurs, Enterprise Architect fournit support Intelli-sense pour la modification des expressions :

Pour éditer une expression, cliquez-droit sur l'expression (dans le champ de droite) et sélectionnez l'option de menu 'Modifier les expressions'. La dialogue « Expression » s'affiche. Cliquez sur la ligne souhaitée et appuyez sur Ctrl+Espace pour afficher le menu Intelli-sense :



- Toutes les variables d'entrée de contexte antérieures à celle actuelle seront incluses (les variables d'entrée de contexte postérieures à celle actuelle sont exclues)
- Pour un Métier Knowledge Modèle (BKM), tous les paramètres seront inclus
- Pour une Décision, toutes les Décisions requises seront incluses

Le modèle DMN peut être généré sous forme de code source pour JavaScript, Java, C# et C++. Étant donné que certaines langues peuvent avoir une syntaxe différente pour certaines expressions, Enterprise Architect fournit des pages de remplacement de langue pour chaque langue. Si aucun code de remplacement n'est spécifié pour une langue, l'expression définie pour la langue FEEL sera utilisée.

Dans le code généré, l'espace à l'intérieur d'un nom de variable sera remplacé par un trait de soulignement.

Simulation du Modèle de Connaissance Métier

Sélectionnez l'onglet « Valeurs des paramètres d'entrée pour Simulation » et remplissez chaque champ.

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

Cliquez sur le bouton Enregistrer puis sur le bouton Simulation de la barre d'outils ; le résultat du test sera présenté dans l'expression Boxed Context.


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

- La valeur du paramètre d'exécution sera affichée ; par exemple, « Taux = 0,00375 »
- La valeur d'exécution de la variable 'Context Entry' sera affichée ; par exemple, « Remboursement mensuel = 1 520,06 »
- Le résultat du Métier Knowledge Modèle (BKM) sera évalué par la dernière saisie et les valeurs affichées sur la ligne de déclaration ; par exemple, "retour = 1540,06"

Vous pouvez utiliser cette fonctionnalité pour tester unitairement un BKM sans connaître le contexte afin qu'il puisse ensuite être invoqué par une Décision ou un autre BKM.

Tableau de Décision

Un Tableau de Décision est une représentation tabulaire d'un ensemble d'expressions d'entrée et de sortie associées, organisées en règles indiquant quelle entrée de sortie s'applique à un ensemble spécifique d'entrées d'entrée.

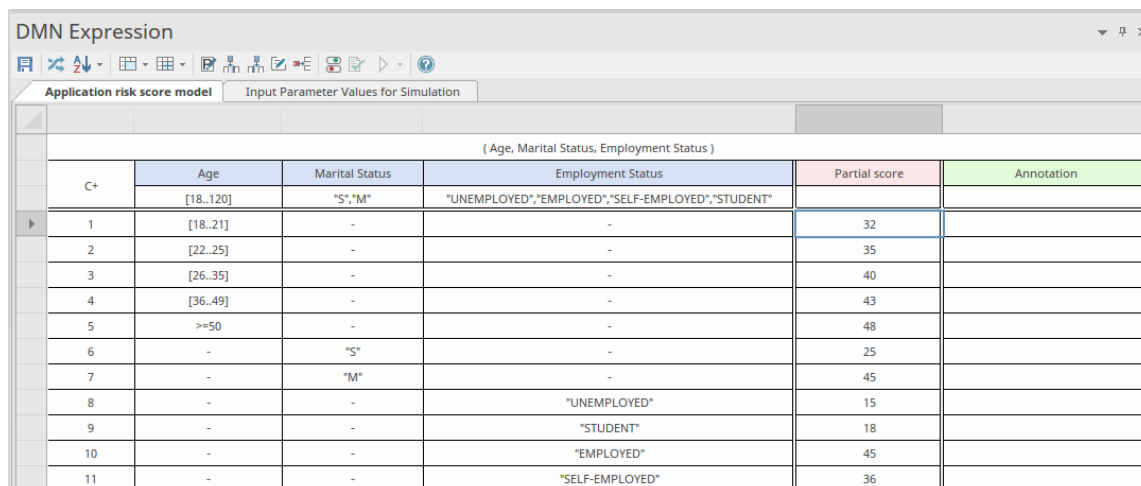
Tableaux Décision sont pris en charge à la fois par les types d'éléments *Décision* et *Métier Knowledge Modèle*. Ils sont désignés par l'icône  dans le coin supérieur droit de l'élément sur un diagramme.

Accéder

Diagramme	<p>Sur un diagramme, double-cliquez sur un élément Décision ou BusinessKnowledgeModel.</p> <p>La fenêtre DMN Expression s'affiche, affichant les détails de l'élément sélectionné.</p>
-----------	--

Aperçu

Cette image montre la fenêtre DMN Expression telle qu'elle apparaît pour un Tableau de Décision.



The screenshot shows the 'DMN Expression' window with a toolbar and a tab titled 'Application risk score model'. Below the tab is a table with the following structure:

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

Un Tableau de Décision se compose de :

- La stratégie Hit Tableau (C+, U, A, P, etc.) qui spécifie la manière dont les règles sont appliquées
- Une liste de règles (1, 2, 3, 4 et ainsi de suite), où chaque ligne de règle contient des entrées d'entrée spécifiques et des entrées de sortie correspondantes
- Une liste de clauses d'entrée (sous les en-têtes bleus), définies comme des expressions qui impliquent généralement une ou plusieurs valeurs d'entrée
- Une liste de clauses de sortie (sous l'en-tête rose), définissant la sortie correspondant à un ensemble spécifique d'entrées
- Annotations facultatives pour chaque règle (sous l'en-tête vert) que vous pouvez ajouter via la barre d'outils de la fenêtre

Une clause d'entrée se compose d'une expression et d'une liste facultative de valeurs autorisées (la ligne juste en dessous des en-têtes de colonnes). Très souvent, l'expression est simplement une valeur d'entrée non modifiée ; cependant, il peut également s'agir d'une expression impliquant plusieurs valeur d'entrée ou peut-être d'une instruction conditionnelle telle que « Score de risque d'application > 100 ». Les valeurs autorisées s'appliquent au *résultat de l'expression* plutôt qu'aux

valeurs d'entrée utilisées.

Chaque clause de sortie se compose d'un identifiant (un nom) et encore une fois d'une liste facultative de valeurs autorisées pour cette clause.

Le Tableau de Décision doit contenir toutes les entrées - et uniquement ces entrées - nécessaires pour déterminer un résultat.

Pour déterminer quelles règles sont appliquées, les expressions définies dans les clauses d'entrée sont évaluées pour les entrées données et les *résultats de l'expression* sont ensuite utilisés pour trouver des règles avec des entrées d'entrée correspondantes.

Lorsque la fenêtre de l'expression DMN n'est pas suffisamment large ou profonde pour afficher toutes les colonnes et toutes les lignes, vous pouvez utiliser les barres de défilement pour accéder au contenu masqué ou faire glisser les bordures vers l'extérieur pour augmenter la largeur de chaque colonne. Les largeurs de colonnes « Entrée » et « Sortie » sont initialement les mêmes, mais vous pouvez ajuster la largeur de chaque colonne indépendamment des autres, en faisant glisser la bordure de la colonne soit dans le tableau, soit dans la barre grise juste en dessous des noms d'onglets.

Barre d'outils pour l'éditeur Tableau de Décision

Lorsqu'un Tableau de Décision est sélectionné, les fonctionnalités disponibles dans la fenêtre DMN Expression sont accessibles via la barre d'outils en haut de la fenêtre, comme indiqué :

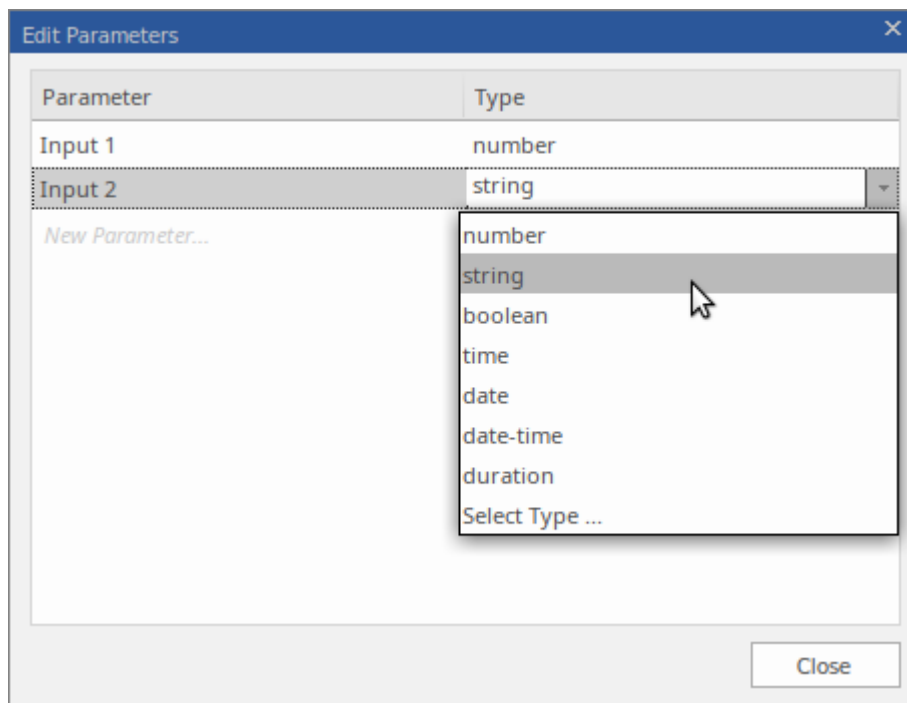


Pour plus de détails, reportez-vous à la rubrique d'aide *de la barre d'outils de l'éditeur Tableau de Décision*.

Paramètres

Dans le cas des éléments Métier Knowledge Modèle (BKM), les paramètres sont utilisés pour transmettre les valeurs d'entrée fournies par l'élément appelant. La logique de décision du BKM est évaluée à l'aide des paramètres d'entrée et le résultat est renvoyé à l'élément appelant. Par défaut, un élément BKM est créé avec deux paramètres d'entrée, « Entrée 1 » et « Entrée 2 ».

Cliquez sur l'icône  dans la barre d'outils de la fenêtre DMN Expression pour afficher la dialogue 'Modifier les paramètres'.



Ici, vous pouvez modifier les noms des paramètres, réorganiser la séquence, définir leurs types de données, créer des paramètres supplémentaires ou supprimer ceux existants.

Politique Hit

Cliquez-droit sur le 'Hit Policy Indicator', puis choisissez la Hit Policy souhaitée dans le menu contextuel. Les différentes stratégies Hit Tableau sont décrites en détail dans la rubrique d'aide *sur les stratégies Hit Tableau de Décision*.


Clauses d'entrée

Une clause d'entrée d'un Tableau de Décision est définie comme une expression. Très souvent, l'expression est simplement une valeur d'entrée non modifiée ; cependant, il peut également s'agir d'une expression impliquant plusieurs valeur d'entrée ou il peut être défini comme une instruction conditionnelle, telle que « Score de risque d'application > 100 ». Les valeurs autorisées s'appliquent au *résultat de l'expression* plutôt qu'aux valeurs d'entrée utilisées et, par conséquent, le type des valeurs doit correspondre au type du résultat de l'expression.

Tableaux Décision sont créés avec deux clauses d'entrée par défaut, « Entrée 1 » et « Entrée 2 ». Le type de données pour ces deux clauses est « numéro ». Dans la fenêtre Expression DMN, les clauses d'entrée sont affichées sous forme d'en-têtes de colonnes sur le Tableau de Décision . Pour modifier une clause d'entrée, cliquez sur l'en-tête de colonne pour sélectionner la cellule, puis cliquez à nouveau ou appuyez sur F2 pour modifier.

La saisie semi-automatique est prise en charge lors de la modification des clauses d'entrée. Cela signifie que, pour les éléments Décision , toutes les entrées connectées à l'élément Décision sont disponibles pour la sélection dans une liste. De même, pour les éléments Métier Knowledge Modèle , les paramètres d'invocation sont proposés à la sélection dans une liste. Consultez la rubrique d'aide *sur la saisie semi-automatique des expressions DMN* pour plus d'informations.

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

Pour ajouter des colonnes supplémentaires d'entrées d'entrée au Tableau de Décision , cliquez sur l'icône  dans la barre d'outils de la fenêtre DMN Expression.

Pour supprimer des colonnes de saisie du tableau , cliquez-droit dans la colonne de saisie indésirable, puis sélectionnez l'option 'Supprimer la colonne de saisie' dans le menu contextuel.

L'ordre des colonnes d'entrée dans le tableau peut être réorganisé en faisant glisser et en déposant les colonnes vers de nouvelles positions. (Faites glisser la cellule sans étiquette tout en haut de la colonne tableau vers la position souhaitée.)

Valeurs autorisées

Lors de la définition d'une colonne « Entrée » ou « Sortie », la deuxième ligne de la colonne définit les valeurs autorisées. Il s'agit d'une cellule facultative dans la colonne, mais utile pour clarifier les entrées dans les lignes en dessous. Lors de l'exécution d'une validation, chacune des cellules situées sous la cellule Valeurs autorisées est vérifiée pour s'assurer qu'elle est conforme à l'expression de cette cellule.

Les expressions utilisées dans cette cellule dépendent de la manière dont la colonne « Entrée » ou « Sortie » est saisie. Par exemple:

- Numéro - [18 ..35]
- String : « Élevé », « Faible », « Moyen »
- Booléen - vrai, faux
- Indéfini - '-'

Valeurs autorisées pour le remplissage rapide

L'expression d'entrée/sortie à laquelle cela fait référence peut être une valeur simple ou une expression FEEL complexe ; cependant, s'il est directement lié au champ « Valeurs autorisées » d'un ItemDefinition, appuyer sur la barre d'espace activera une option de remplissage rapide pour définir les « Valeurs autorisées » telles que définies dans ItemDefinition (généralement référencées via un élément InputData).

U	Temperature
1	Hot, Warm, Frozen, Cold

Lignes à remplissage rapide

Une fois le champ « Valeurs autorisées » défini, en plus de restreindre les valeurs pouvant être utilisées lors de la définition des règles dans le tableau , le champ « Valeurs autorisées » offre également à l'utilisateur une option de remplissage rapide. Ceci est invoqué, dans une cellule de règle, en appuyant sur la barre d'espace et en sélectionnant l'élément requis :


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

-
 Hot
 Warm
 Frozen
 Cold

Pour plus de détails, consultez la rubrique d'aide *Achèvement automatique des expressions DMN*.

Clauses de sortie

Une clause de sortie se compose d'un nom, d'un type de données et d'une liste facultative de valeurs autorisées. Pour modifier une clause de sortie, cliquez sur la cellule d'en-tête de colonne pour sélectionner la cellule, puis cliquez à nouveau ou appuyez sur F2 pour modifier.

Pour ajouter des colonnes supplémentaires d'entrées de sortie au Tableau de Décision, cliquez sur l'icône  dans la barre d'outils de la fenêtre de l'éditeur d'expression.

Pour supprimer des colonnes de sortie du tableau, cliquez-droit dans la colonne de sortie indésirable, puis sélectionnez l'option 'Supprimer la colonne de sortie' dans le menu contextuel.

L'ordre des colonnes dans le tableau peut être réorganisé en faisant glisser et en déposant les colonnes vers de nouvelles positions. (Faites glisser la cellule sans étiquette tout en haut de la colonne tableau vers la position souhaitée.)

Type de données pour les clauses d'entrée/sortie

Pour que la simulation fonctionne, il est essentiel de définir le type de données pour toutes les clauses d'entrée et de sortie. Les validations de plage, d'espacement et de chevauchement sont prises en charge pour les clauses de type « numéro », mais la validation ne peut pas être effectuée si le type n'a pas été spécifié. La génération de code pour les langages typés tels que C++, C# et Java nécessite que les types de données soient spécifiés. Lorsque le type de données est spécifié comme « string », il n'est pas nécessaire de mettre chaque string littérale entre guillemets. Les valeurs String sont affichées en italique si le type a été déclaré.


Pour définir le type de données, cliquez-droit sur l'en-tête de colonne Entrée ou Sortie et sélectionnez le type requis dans la liste.

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

- ✓ type: string
- type: boolean
- type: number
- type: date
- type: time
- type: duration
- Delete Input Column

Définir les règles Tableau de Décision


Les règles Tableau de Décision sont définies en spécifiant les entrées d'entrée et les entrées de sortie correspondantes dans les cellules d'une ligne tableau. Pour les types de données « nombre », les entrées d'entrée peuvent être spécifiées sous la forme d'une valeur unique ou sous la forme d'une plage de nombres, telle que « <10 », « >100 » ou « [2..8) ». (Lors de la définition de plages de nombres, l'utilisation de round indique que le nombre limite n'est PAS inclus ; l'utilisation de crochets indique que le nombre limite est inclus.) Les entrées de sortie doivent spécifier une seule valeur par cellule.

Des règles supplémentaires peuvent être ajoutées à la liste des règles en cliquant sur l'icône  dans la barre d'outils. Les règles indésirables peuvent être supprimées du tableau en cliquant avec le bouton droit sur la règle et en sélectionnant l'option « Supprimer la ligne de règle » dans le menu contextuel.

Les règles existantes peuvent être copiées et collées dans le tableau en sélectionnant d'abord les règles (utilisez « Ctrl+Clic » pour ajouter/supprimer de la sélection), puis en utilisant les options de menu « Copier les règles dans le presse-papiers » et « Coller les règles depuis le presse-papiers » pour effectuer le copier-coller. Les règles copiées peuvent ensuite être modifiées en sélectionnant et en modifiant des entrées de cellule individuelles.

Si le champ « Valeurs autorisées » est défini pour une string ou une expression booléenne, la barre d'espace peut être utilisée pour afficher une liste de valeurs parmi lesquelles sélectionner, comme indiqué dans la section *Valeurs autorisées - Lignes à remplissage rapide* précédente.

Les règles peuvent également être triées dans le tableau, soit par :

- En cliquant sur l'icône  dans la barre d'outils, puis en choisissant « Trier par entrée » ou « Trier par sortie », ou
- Cliquez avec le bouton droit sur des règles individuelles dans le tableau et sélectionnez l'option « Déplacer la règle vers le haut » ou « Déplacer la règle vers le bas » dans le menu contextuel.

Pour déterminer quelles lignes tableau sont sélectionnées pour la sortie, les *expressions* définies par les clauses d'entrée sont évaluées pour les entrées données et les *résultats* des expressions sont ensuite comparés aux entrées d'entrée des lignes du tableau. Lorsque les résultats de l'expression correspondent aux entrées d'entrée d'une ligne tableau, cette ligne est sélectionnée pour la sortie.

La « politique Hit » du Tableau de Décision détermine comment les lignes correspondantes du tableau sont ensuite utilisées pour produire sa sortie.

Formats de règles

Vous pouvez choisir - à l'aide d'une icône de la barre d'outils - d'afficher le Tableau de Décision dans l'un des trois formats, comme indiqué ici.

Format Rule-as-Row, où la règle est développée le long de lignes avec les entrées, les sorties et les annotations définies dans des colonnes :

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

Format de règle en tant que colonne, dans lequel les règles sont développées en colonnes avec les entrées, sorties et annotations définies le long des lignes :

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

Format Rule-as-Crosstab, dans lequel les règles sont formées à partir d'entrées définies comme un ensemble de lignes AND une combinaison de colonnes, avec des sorties définies dans les cellules qui se croisent. (Note que ce format masque les champs 'Annotation') :

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

A la fin d'une simulation, dans un Tableau de Décision croisé, les entrées d'entrée et les entrées de sortie associées sont mises en évidence. Par exemple, dans ce traitement de simulation, une remise de 0,10 a été obtenue pour un client Métier dont la taille de la commande était inférieure ou égale à 10 et dont la livraison n'était pas applicable.

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

Paramètres du tableau croisé

Dans le format Rule-as-Crosstab, comme les entrées forment à la fois des lignes et des colonnes et que les sorties se trouvent aux intersections, les étapes de définition des valeurs sont légèrement différentes de celles des deux autres formats.

- Pour ajouter un autre type d'entrée, cliquez-droit sur l'en-tête de la colonne d'entrée et sélectionnez l'option « Ajouter une entrée ». Vous êtes invité à saisir le nom de l'entrée ; l'entrée est ajoutée sous la forme d'un ensemble de champs sous les champs de la colonne actuelle.
Pour supprimer un type de saisie des colonnes, cliquez-droit sur son ensemble de champs et sélectionnez l'option 'Supprimer la saisie'. Le nom de l'entrée et son ensemble de champs sont supprimés des en-têtes de colonnes.
- Pour ajouter un autre type de sortie, cliquez-droit sur le bloc Sortie en haut à gauche de la fenêtre et sélectionnez l'option 'Ajouter une sortie'. Vous êtes invité à saisir le nom de la sortie ; le nom est ajouté au bloc Output et une nouvelle ligne est ajoutée à chaque cellule du corps de la fenêtre.

Pour supprimer un type de sortie, cliquez-droit sur le nom du type dans le bloc Sortie en haut à gauche de la fenêtre et sélectionnez l'option 'Supprimer la sortie'. Le nom de la sortie et ses champs dans la grille sont supprimés.

3. Vous pouvez basculer entre les types d'entrée pour en sélectionner un pour les en-têtes de ligne. Cliquez-droit sur la ligne et cliquez sur l'option 'Sélectionner l'entrée comme en-tête de ligne'. Cela affiche une liste des types d'entrée ; cliquez sur le type à utiliser comme en-tête de ligne ; les autres types sont regroupés dans les colonnes.
4. Pour ajouter une ligne ou une colonne d'entrée valeur aux entrées, cliquez-droit sur une ligne ou une colonne actuelle et sélectionnez l'option « Ajouter une ligne d'entrée d'entrée » ou « Ajouter une colonne d'entrée d'entrée », selon le cas. Un prompt s'affiche pour le nom de l'entrée d'entrée ; lorsque vous saisissez ceci, la ligne ou la colonne appropriée est ajoutée au Tableau de Décision .
Pour supprimer une ligne ou une colonne de saisie valeur , cliquez-droit dessus et sélectionnez l'option 'Supprimer la ligne d'entrée de valeur' ou 'Supprimer la colonne d'entrée de valeur'. La ligne ou la colonne sélectionnée est supprimée du tableau .
5. Dans les colonnes d'entrée, chaque ligne correspond à un type d'entrée. Si vous souhaitez déplacer la ligne d'un type d'entrée au-dessus ou en dessous d'un autre, cliquez-droit dessus et sélectionnez l'option « Déplacer l'entrée vers le haut » ou « Déplacer l'entrée vers le bas ». Le menu contextuel fournit uniquement les options qui peuvent être actionnées, de sorte qu'il n'est pas possible de déplacer, par exemple, la dernière ligne vers le bas, l'option « Déplacer l'entrée vers le bas » n'est pas répertoriée.

Barre d'outils pour l'éditeur Tableau de Décision

Ce tableau fournit des descriptions des fonctionnalités accessibles dans la fenêtre DMN Expression lorsqu'un Tableau de Décision est sélectionné.

Options barre d'outils

Icône	Description
	Enregistrez les modifications apportées à l'élément Décision ou BusinessKnowledgeModel actuellement sélectionné.
	Basculez la vue du Tableau de Décision entre Rule-as-Row, Rule-as-Column et Rule-as-Crosstab. Vous pouvez également cliquer sur la flèche déroulante et sélectionner le format souhaité.
	Cliquez sur « Trier par entrée » pour trier les règles par colonnes d'entrée ; cliquez sur « Trier par sortie » pour trier les règles par colonnes de sortie. Les colonnes peuvent être glissées et déposées pour organiser l'ordre de tri.
	Fusionnez les cellules des règles adjacentes, où le contenu des entrées d'entrée est le même. Vous pouvez modifier le contenu des cellules fusionnées. Lors de la simulation, les éléments fusionnés sont mis en surbrillance.
	Divisez les cellules d'entrée d'entrée qui ont été précédemment fusionnées.
	Affichez la fenêtre « Modifier les paramètres », dans laquelle vous pouvez spécifier les noms et les types de données des paramètres transmis lors de l'appel de la logique de décision d'un élément BusinessKnowledgeModel.
	Ajoutez une colonne de saisie au Tableau de Décision .
	Ajoutez une colonne de sortie au Tableau de Décision .
	Ajoutez une colonne d'annotation (avec une cellule d'en-tête verte) au tableau , dans laquelle vous pouvez enregistrer de courtes notes ou des commentaires sur la règle. (Voir les illustrations de la ligne Règle en tant que ligne/Règle en tant que colonne plus tôt.) Vous pouvez ajouter plusieurs colonnes d'annotation si nécessaire, en tapant un titre de colonne approprié dans chaque cellule d'en-tête. Pour supprimer une colonne d'annotation, cliquez-droit dessus et sélectionnez l'option 'Supprimer la colonne d'annotation'.
	Joindre une règle au Tableau de Décision .
	Afficher ou masquer les champs de valeurs autorisées pour les colonnes « Entrée » et « Sortie ». Les valeurs autorisées définies pour une entrée ou une sortie seront utilisées pour la validation et l'édition de complétion automatique.
	Effectuer la validation du Tableau de Décision . Enterprise Architect effectuera une


	série de validations pour vous aider à découvrir d'éventuelles erreurs dans le Tableau de Décision .
	<p>Ce bouton est activé lorsqu'un Tableau de Décision est défini pour un élément BusinessKnowledgeModel.</p> <p>Sélectionnez l'onglet 'Valeurs des paramètres d'entrée pour Simulation ', complétez les champs, puis cliquez sur ce bouton. Le résultat du test sera présenté sur le Tableau de Décision , avec les valeurs d'exécution des entrées et des sorties affichées et les règles valides mises en évidence.</p> <p>Vous pouvez utiliser cette fonctionnalité pour tester unitairement un élément BusinessKnowledgeModel, sans spécifier son contexte.</p> <p>Un certain nombre d'options de menu sont disponibles pour ce bouton de la barre d'outils. Pour plus d'informations, consultez la rubrique d'aide <i>Simuler DMN Modèle</i> .</p>

Tableau de Décision Hit Policy

La Hit Policy précise le résultat du Tableau de Décision en cas de chevauchement de règles. Le caractère unique dans une cellule particulière Tableau de Décision indique le type tableau et reflète sans ambiguïté la logique de décision.

Politiques Hit unique :

- **Unique** : aucun chevauchement n'est possible et toutes les règles sont disjointes ; une seule règle peut être mise en correspondance (c'est la valeur par défaut)
- **Quoi** qu'il en soit : il peut y avoir un chevauchement, mais toutes les règles de correspondance affichent des entrées de sortie égales pour chaque sortie, donc n'importe quelle correspondance peut être utilisée
- **Priorité** : plusieurs règles peuvent correspondre, avec des entrées de sortie différentes ; cette politique renvoie la règle de correspondance avec la priorité de sortie la plus élevée
- **Premièrement** : plusieurs règles (qui se chevauchent) peuvent correspondre, avec des entrées de sortie différentes ; le premier accès par ordre de règle est renvoyé

Politiques Hit multiples :

- **Ordre de sortie** : renvoie tous les résultats dans un ordre de priorité de sortie décroissant
- **Ordre des règles** : renvoie tous les résultats dans l'ordre des règles
- **Collecter** : renvoie tous les résultats dans un ordre arbitraire ; un opérateur ("+", "<", ">", "#") peut être ajouté pour appliquer une fonction simple aux sorties

Les opérateurs de collecte sont :

- **+** (somme) : le résultat du Tableau de Décision est la somme de toutes les sorties distinctes
- **<** (min) : le résultat du Tableau de Décision est la plus petite valeur de toutes les sorties
- **>** (max) : le résultat du Tableau de Décision est la plus grande valeur de toutes les sorties
- **#** (count) : le résultat du Tableau de Décision est le nombre de sorties distinctes

Exemple de politique Hit unique

La politique Hit « unique » est le type de Tableau de Décision le plus populaire et toutes les règles sont disjointes.

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

Exemple de politique Hit prioritaire

Dans un tableau avec la politique Hit « Prioritaire », plusieurs règles peuvent correspondre, avec différentes entrées de sortie. Cette stratégie renvoie la règle de correspondance avec la priorité de sortie la plus élevée.

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

Note : La liste des valeurs autorisées permet de définir la priorité de sortie. Ici, les valeurs autorisées sont répertoriées comme INELIGIBLE et ELIGIBLE ; INELIGIBLE est défini comme ayant une priorité plus élevée que ELIGIBLE.

Un résultat de simulation possible pourrait ressembler à ceci :

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

Les règles de correspondance sont mises en surbrillance, mais la sortie de la règle 2 est choisie car INELIGIBLE a une priorité plus élevée que ELIGIBLE.

Exemple de politique Hit de collecte-somme

Pour un Tableau de Décision avec la politique Hit « Collect-Sum » (C+), le résultat du Tableau de Décision est la somme de toutes les sorties distinctes.

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

Dans cet exemple, le score partiel de sortie est calculé comme suit : $43 + 45 + 45 = 133$.

Validation Tableau de Décision

Un Tableau de Décision est l'une des expressions DMN les plus courantes et les plus utiles utilisées pour exprimer la logique de décision. Cependant, modélisation d'un Tableau de Décision peut également être compliquée, surtout si plusieurs clauses d'entrée sont utilisées en combinaison pour de nombreuses règles Tableau de Décision. Enterprise Architect permet de valider facilement Tableaux Décision, comme expliqué dans cette rubrique.

Accéder

Fenêtre d'expression DMN	Simuler > Analyse Décision > DMN > Expression DMN : Bouton Valider
Fenêtre Simulation DMN	Simuler > Analyse Décision > DMN > Ouvrir Simulation DMN > Configurer : bouton Valider

Détection des entrées hors de portée

Il est de bonne pratique de définir des « valeurs autorisées » pour les clauses d'entrée et les clauses de sortie d'un Tableau de Décision. La liste des « valeurs autorisées » est utilisée pour effectuer une vérification de plage des valeurs d'entrée d'entrée et de sortie pour les règles tableau.

DMN Expression

Application risk score model Input Parameter Values for Simulation

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

Notes DMN Expression

System Output

System Script DMN Validation Help System

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

Dans cet exemple :

- La clause d'entrée « Âge » définit une plage de [20..120] ; cependant, l'entrée d'entrée pour la règle 1 spécifie une plage de [18..21] ; ceci est en dehors de la plage des valeurs autorisées, donc la règle 1 est signalée comme invalide
- La clause « État civil » définit ses valeurs autorisées comme une énumération de « S, M » ; la règle 12 spécifie une valeur de « D », par conséquent cette règle est également signalée comme invalide

Ces problèmes peuvent être corrigés, soit en mettant à jour les « valeurs autorisées », soit en modifiant les entrées d'entrée pour les règles non valides, en fonction des règles métier réelles.

Détection d'exhaustivité - signaler les lacunes dans les règles

Les lacunes dans les règles d'un Tableau de Décision signifient que, étant donné une combinaison de valeurs d'entrée, aucune règle ne correspond. Cela indique qu'une logique ou une règle peut manquer (sauf si une sortie par défaut est définie).

Lorsque le Tableau de Décision contient de nombreuses règles qui spécifient des plages de numéros, il devient difficile de détecter visuellement les lacunes et prend beaucoup de temps pour composer et exécuter des cas de test exhaustifs.

Par exemple:

DMN Expression

Post-bureau risk category table

Input Parameter Values for Simulation

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

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

Notes DMN Expression

System Output

System Script DMN Validation Help System

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

La validation signale une lacune dans les règles. Une inspection plus approfondie révèle une erreur dans la règle 9. L'entrée d'entrée (580..600] doit être [580..600].

Détection de chevauchement de règles pour la stratégie Hit unique

Lorsque les règles se chevauchent, pour une combinaison donnée de valeurs d'entrée, plusieurs règles correspondent. Il s'agit d'une violation si le Tableau de Décision spécifie sa politique Hit comme « Unique ».

Lorsque le Tableau de Décision contient de nombreuses règles qui spécifient des plages de numéros, il devient difficile de détecter visuellement les lacunes et prend beaucoup de temps pour composer et exécuter des cas de test exhaustifs.

Par exemple:

DMN Expression

Post-bureau risk category table Input Parameter Values for Simulation

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

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

Notes DMN Expression

System Output

System Script DMN Validation Help System

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

La validation signale un chevauchement dans les règles, impliquant les règles 4 et 5. Une inspection plus approfondie révèle que le chevauchement existe dans la troisième entrée « Score de crédit », où « <610 » chevauche « [600..625] ». Vous pouvez corriger ce problème soit en modifiant la règle 4 par « <600 », soit en modifiant la règle 5 par « [610..625] », pour refléter les règles métier réelles.

Intégrer dans un élément de classe UML

Après avoir créé et simulé un Décision Modèle , vous pouvez générer un module DMN en Java, JavaScript , C++ ou C# et le tester.

Le module DMN peut être intégré à un élément de classe UML , de sorte que le code généré à partir de cet élément de classe peut réutiliser le module DMN et être bien structuré. Puisqu'un élément Class peut définir un Statemachine , après intégration avec le module DMN, la simulation Statemachine Exécutable pourra de manière générique utiliser la puissance du module DMN.

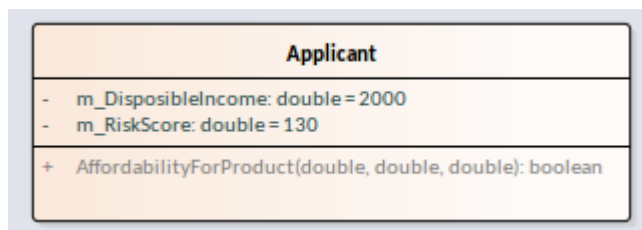
Dans cette rubrique, nous expliquerons le processus d'intégration d'un Modèle DMN avec un élément de classe UML , en considérant :

- Exigence de l'élément de classe
- Modèles DMN
- Liaison DMN à la classe et à Intelli-sense
- Génération de code sur l'élément Class

Exigence de l'élément de classe

Supposons que nous ayons un *candidat* de classe avec une opération *AffordabilityForProduct* qui évalue si le candidat peut se permettre un produit de prêt.

Un modèle simplifié ressemble à ceci :



Le groupe *Candidat* contient deux attributs, qui sont en fait calculés à partir de données plus basiques telles que le revenu mensuel du candidat, ses dépenses, ses remboursements existants, son âge et sa situation professionnelle.

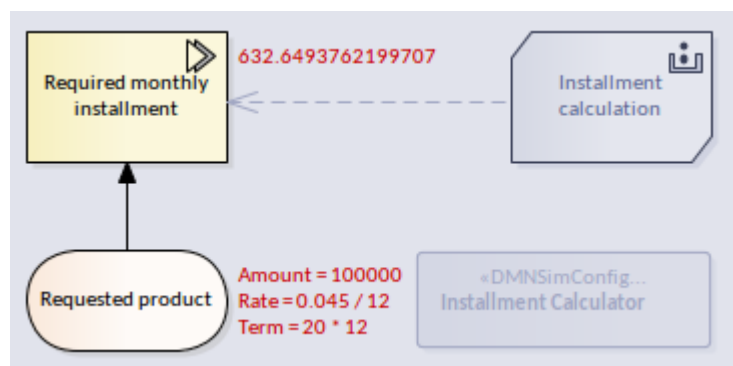
Dans cet exemple, cependant, nous simplifions le modèle en sautant ces étapes et en fournissant directement le revenu disponible et le score de risque.

Modèles DMN

Dans cet exemple, nous avons deux modèles DMN disjoints pour montrer qu'une classe UML peut intégrer plusieurs modèles DMN.

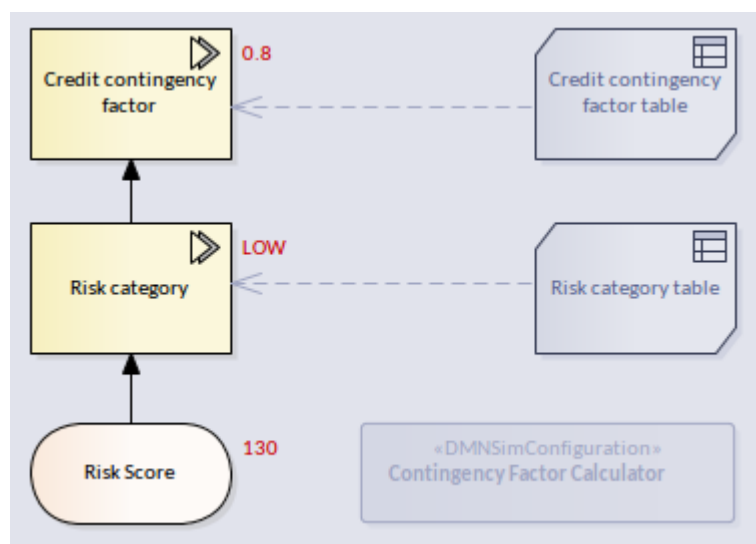
Calculateur de versements

Ce modèle DMN calcule le remboursement mensuel en fonction du montant, du taux et des modalités. Il est composé d'un InputData, d'une Décision et d'un Métier Knowledge Modèle .



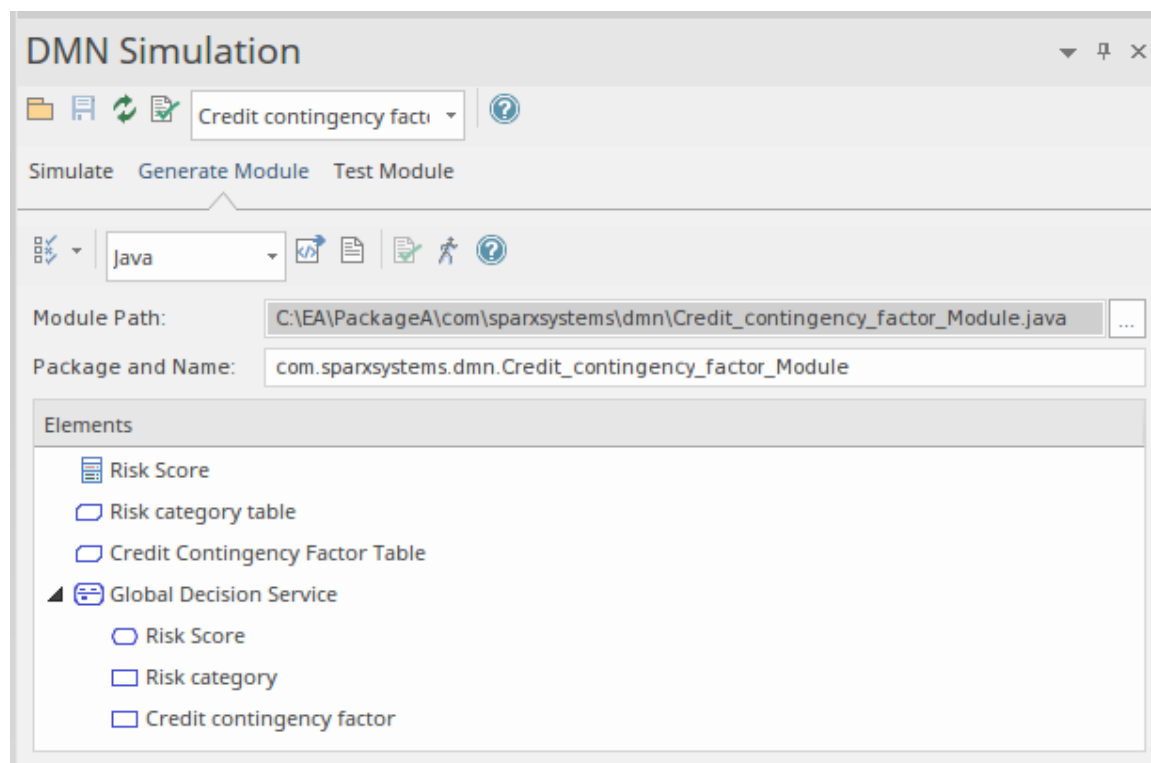
Calculateur du facteur de contingence du crédit

Ce modèle DMN calcule le facteur de contingence du crédit en fonction du score de risque du demandeur. Il est composé d'un InputData, de deux Décisions et de deux Modèles de Connaissances Métier.



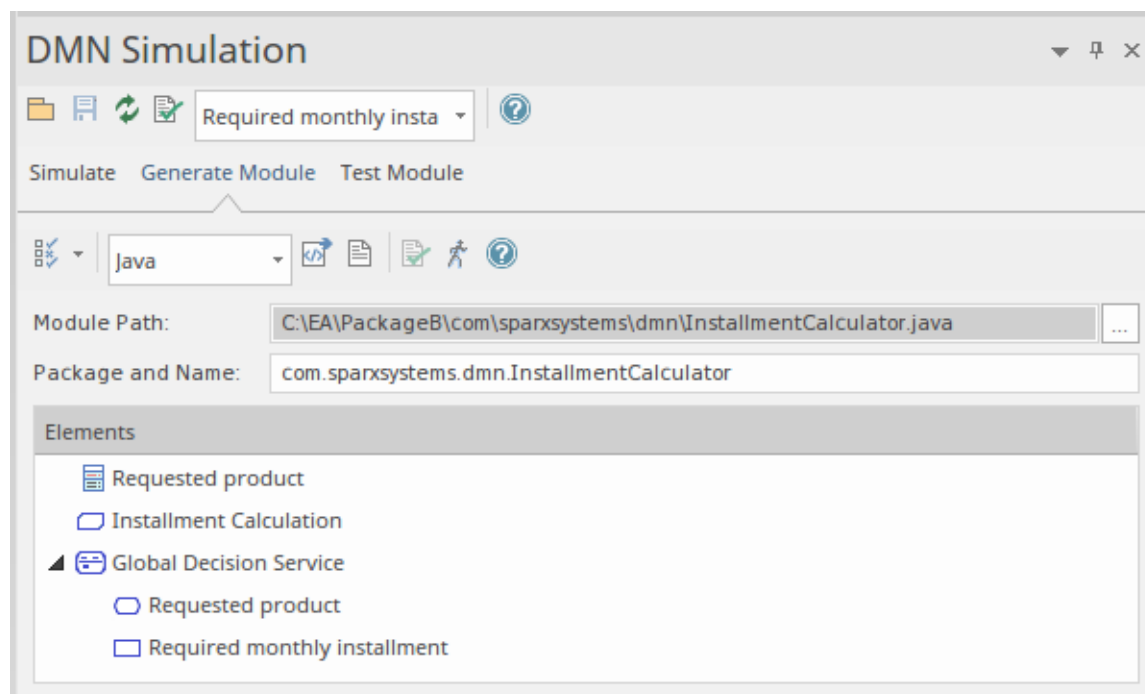
Note : Dans cet exemple, nous nous concentrons sur la façon d'intégrer des modules DMN dans un élément Class ; Les détails des éléments DMN ne sont pas décrits ici.

Générer du code pour les deux modèles DMN

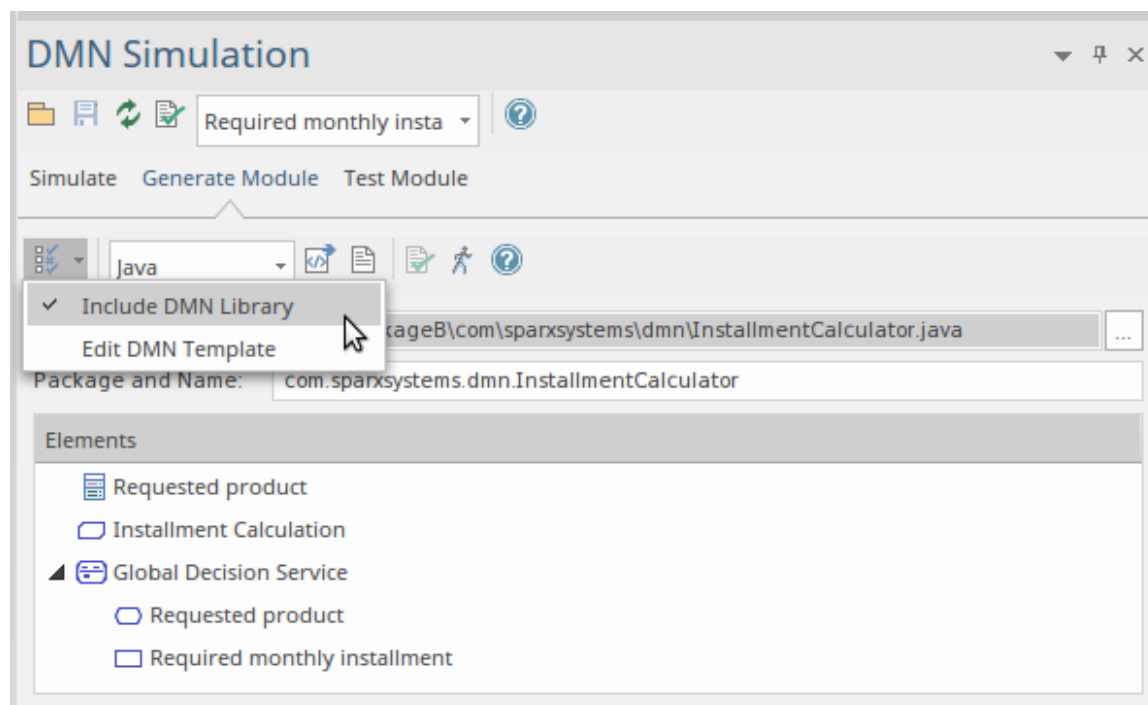


Cliquez sur l'icône Générer du Code, et vérifiez que vous voyez bien cette string dans la fenêtre Sortie Système, onglet 'DMN' :

Le module DMN est compilé avec succès.



Note : Puisque ce modèle utilise une fonction PMT intégrée, la Bibliothèque DMN doit être incluse :

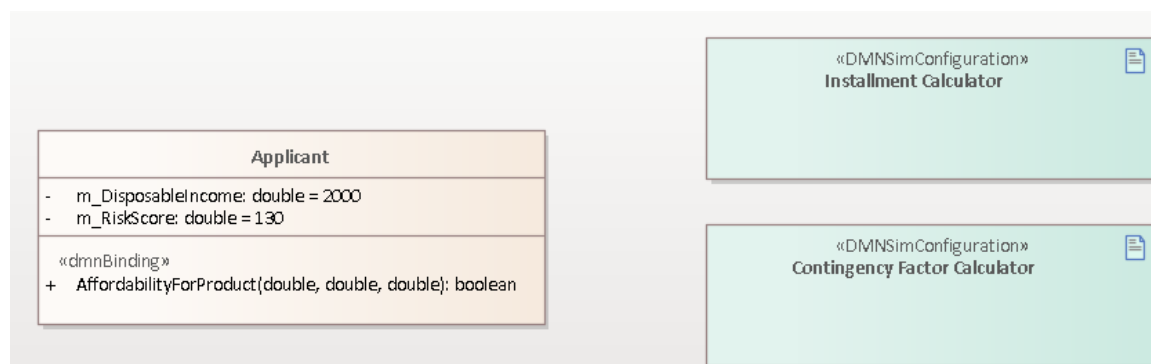


Cliquez sur l'icône Générer du Code, et vérifiez que vous voyez bien cette string dans la fenêtre Sortie Système, page 'DMN' :

Le module DMN est compilé avec succès.

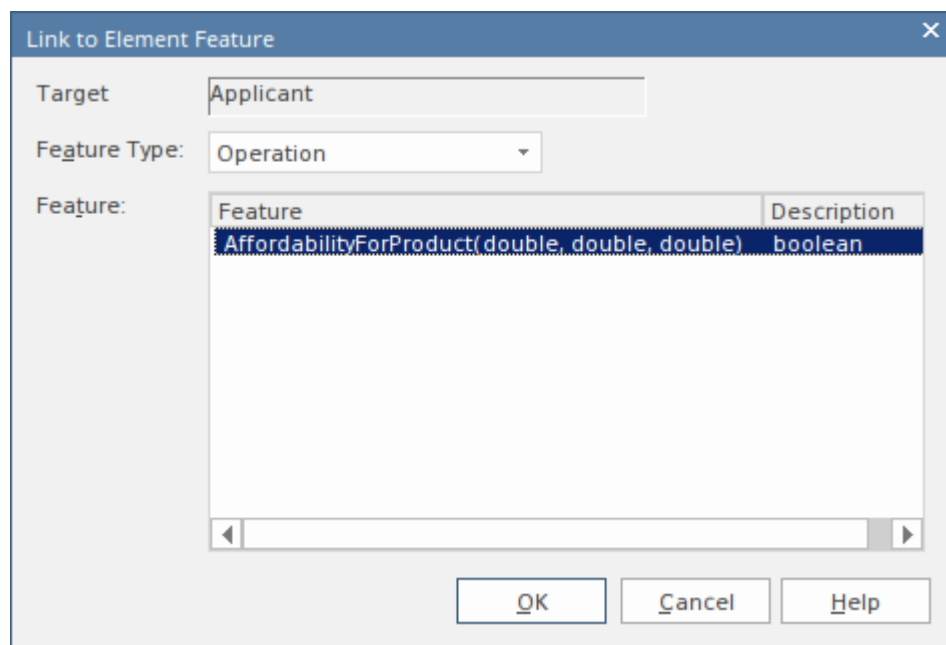
Liaison DMN à la classe et à Intelli-sense

Placez les deux artefacts DMNSimConfiguration sur le diagramme de classes .



Utilisez Quick Linker pour créer un connecteur de dépendance du *candidat* de classe à chacun des artefacts DMN.

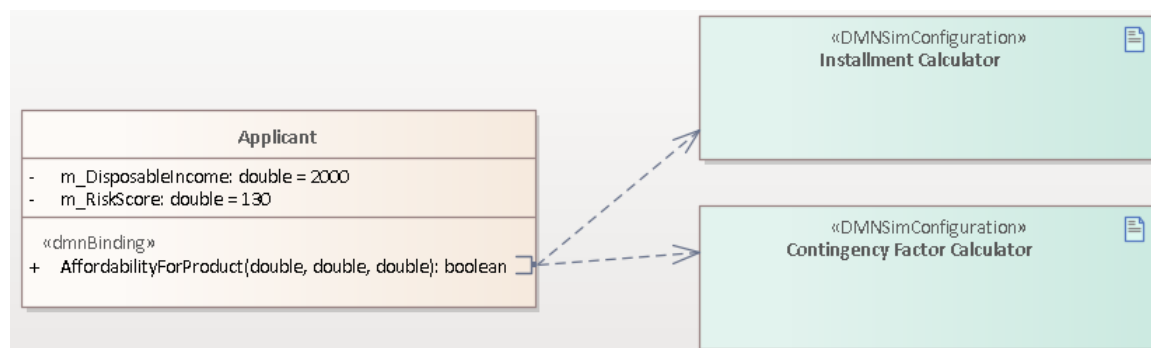
A la création du connecteur, une dialogue vous prompt de choisir l'opération à lier au module DMN.



Lorsque le module DMN est lié à l'opération :

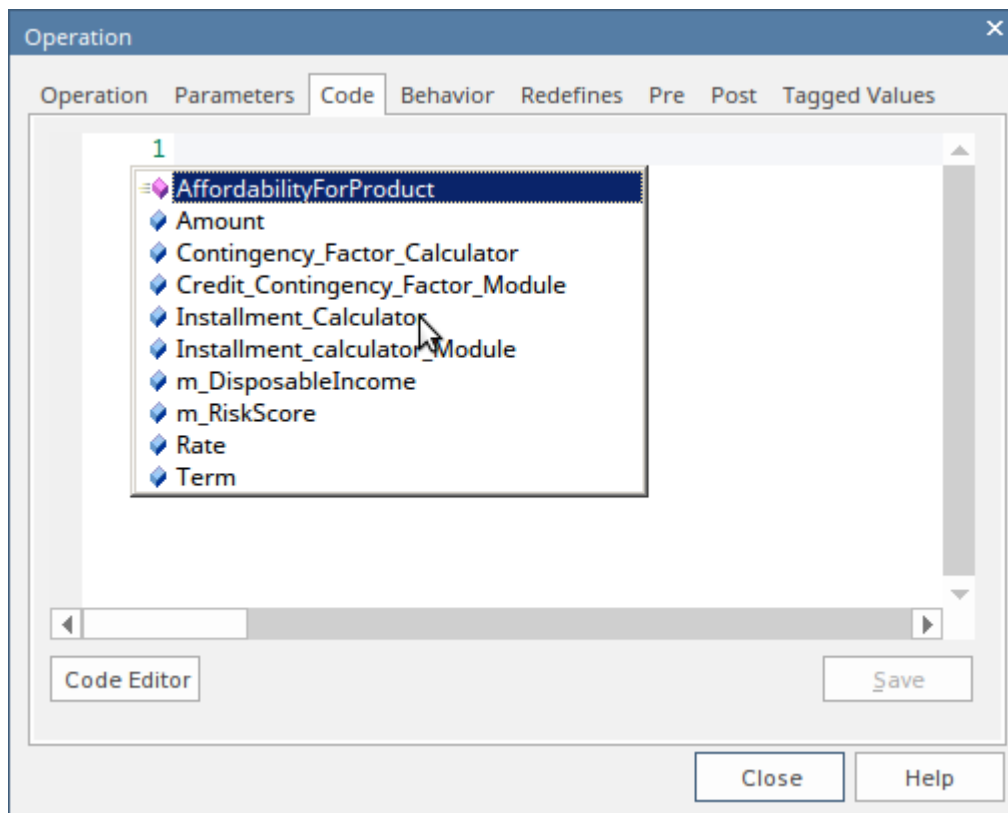
- L'opération prend un stéréotype « dmnbinding »
- Le connecteur Dépendance est lié à l'opération

Plusieurs artefacts DMN peuvent être liés à la même opération.



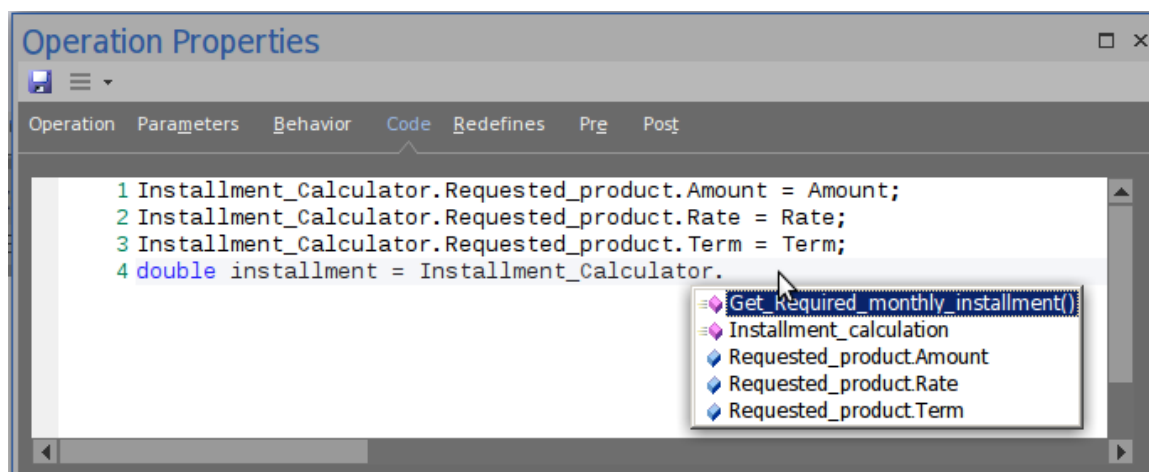
Après les liaisons DMN, Intelli-sense pour l'éditeur de code de l'opération prendra support les modules DMN. Pour déclencher l'Intelli-sense, utilisez ces combinaisons de touches :

- Ctrl+Espace - dans la plupart des cas
- Ctrl+Shift+Espace - lorsque Ctrl+Espace ne fonctionne pas après une parenthèse '(' ; par exemple, les arguments d'une fonction, ou à l'intérieur des parenthèses d'une condition 'If'

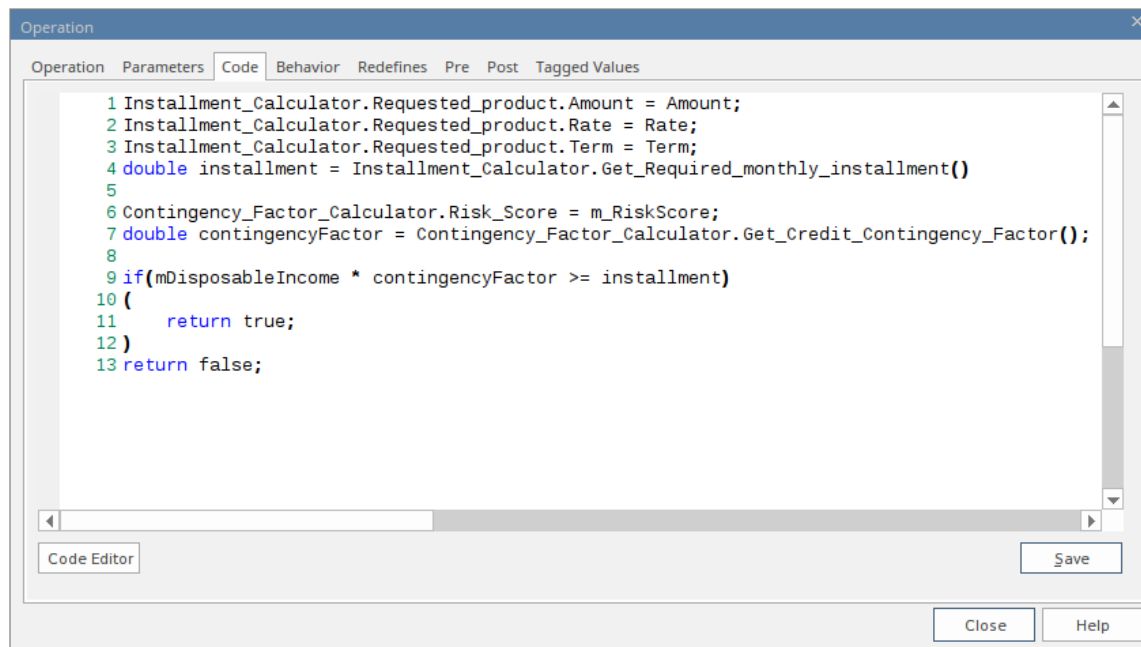


- Les attributs de classe seront répertoriés - m_RiskScore, m_DisposableIncome
- Les paramètres de fonctionnement seront répertoriés - Montant, Taux, Durée
- Les opérations seront répertoriées - AffordabilityForProduct
- Tous les modules DMN liés seront répertoriés - Contingency_Factor_Calculator, Installment_Calculator

Il est assez simple de composer le code avec support Intelli-sense. En accédant au module DMN, toutes les données d'entrée, décisions et modèles de connaissances Métier seront répertoriés pour la sélection.



Cette illustration montre que nous sélectionnons « Get_Required_monthly_installment() » dans Installment_Calculator. Il s'agit de la mise en œuvre finale de l'opération.



Génération de code pour la classe (avec intégration DMN)

'Générer Code sur Classe Candidat' produit ce code :

```

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

```

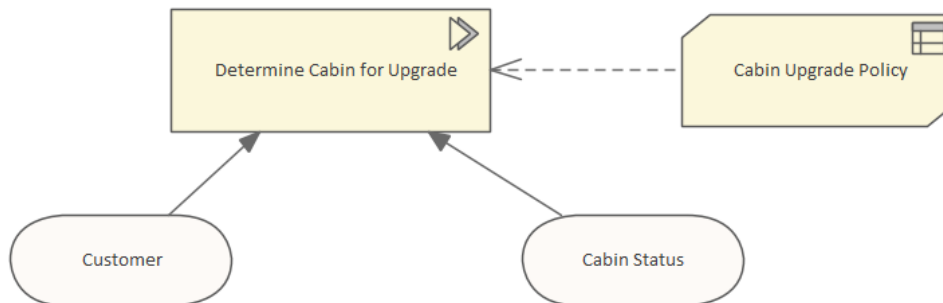
- Le(s) module(s) DMN sont générés en tant qu'attributs de la classe
- Le code de l'opération dmnbinding est mis à jour

Note : Que l'option de génération soit 'Overwrite' ou 'Synchroniser', le code de l'opération sera mis à jour s'il possède le stéréotype 'dmnbinding'.

Exemple Diagramme

Imaginez que vous êtes un agent de réservation d'une compagnie aérienne travaillant au comptoir d'enregistrement d'une compagnie aérienne nationale très fréquentée. Il est essentiel de faire décoller l'avion à temps, car les retards peuvent entraîner des frais appliqués par les contrôleurs d'aéroport, la nécessité de voler à une altitude inférieure, augmentant le coût du carburant et d'autres pénalités.

Un message du superviseur apparaît sur votre écran indiquant que la cabine économique est en sursur réservation ; vous devrez surclasser certains passagers en Métier ou en Première Classe, mais quels passagers choisir et dans quelle cabine les surclasser ? Une décision doit être prise, mais quels facteurs doivent être pris en compte ? Ceci peut être enregistré dans une Décision Modèle à l'aide d'un diagramme Décision Exigences .



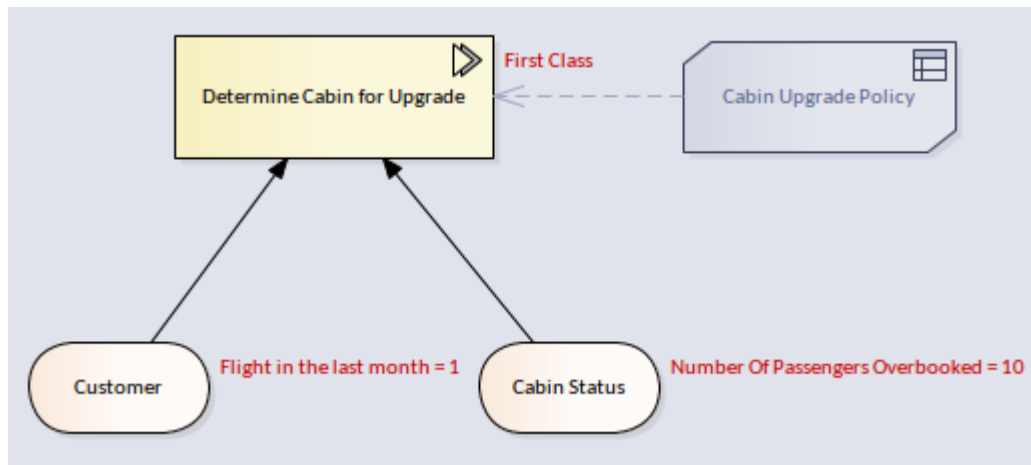
C'est utile, mais l'agent d'enregistrement très occupé devra quand même peser tous les facteurs et prendre une décision impartiale. Un passager mécontent devrait-il avoir la priorité sur un voyageur fréquent de niveau Gold, ou le fait qu'un passager particulier soit en correspondance sur un vol international doit-il avoir la priorité. Ces « règles » peuvent toutes être consignées dans un Tableau de Décision , indiquant clairement quels passagers doivent bénéficier d'un surclassement et dans quelle cabine : Métier ou Première Classe. Cela facilitera grandement la prise de décision et les règles pourront être formulées, convenues et vérifiées pour leur cohérence au siège social. Dans cet exemple, nous avons gardé les choses simples et utilisé deux facteurs : premièrement, le nombre de vols effectués par le passager au cours du mois dernier et deuxièmement, le degré de sursur réservation de la cabine.

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

Le tableau est divisé en colonnes et en lignes. Il existe trois types de colonnes : les entrées nécessaires à la prise de décision, les sorties résultant de l'application des règles et les annotations.

Ceci est encore une fois très utile, mais nécessite toujours que l'agent d'enregistrement très occupé soit en mesure de trouver toutes les informations requises pour trouver la bonne ligne dans le Tableau de Décision . Même si toutes ces informations étaient disponibles, une mauvaise décision pourrait toujours résulter d'une erreur humaine lors de la sélection de la mauvaise ligne dans le tableau .

Heureusement, les modèles Décision peuvent être automatisés et générés sous forme de code de programmation pouvant être exécuté par une application. Ainsi, notre agent d'enregistrement très occupé n'aurait rien à faire ni à prendre de décision ; pendant qu'il ou elle enregistrerait les passagers, si un passager en particulier avait droit à un surclassement, cela serait visible sur l'écran de l'ordinateur. Dans le diagramme suivant, le modèle a été simulé afin que le personnel commercial et technique puisse convenir que le modèle a été défini correctement. N'importe quel nombre d'ensembles de données définis par l'utilisateur peut être utilisé pour tester le modèle avant de générer le code de programmation qui exécuter dans le système d'enregistrement et affichera le résultat à l'utilisateur final.



Lors du développement des modèles, un utilisateur commercial ou technique peut parcourir la simulation et le système montrera à cet utilisateur quelle ligne du Tableau de Décision a été déclenchée pour déterminer le résultat. Ceci est très utile dans les modèles composés de plusieurs décisions.

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

Il est courant que les règles qui régissent la décision de mise à niveau changent. Par exemple, le service marketing peut décider de récompenser les passagers qui voyagent sur des vols long-courriers. Le diagramme Décision Exigences peut être modifié pour inclure la nouvelle entrée, le Tableau de Décision modifié et le code de programmation régénéré. Une fois les changements apportés aux systèmes aéroportuaires, les bons passagers seront automatiquement surclassés. L'agent d'enregistrement pouvait toujours consulter les Tableaux Décision lors d'une séance de formation et de briefing pour comprendre les règles.

Module de génération de code et Test

Une fois qu'un modèle Décision est créé et simulé, vous pouvez générer un module DMN en Java, JavaScript, C++ ou C#. Ce module DMN peut être utilisé avec Enterprise Architect BPSim Moteur d'Exécution, Statemachine Exécutable ou votre propre projet.

Enterprise Architect fournit également une page « Module Test », qui est un pré-processus pour l'intégration de DMN avec BPMN. Le concept est de fournir un ou plusieurs éléments BPMN2.0::DataObject, puis de tester si une Décision cible spécifiée peut être évaluée correctement ou non.

Si une erreur ou une exception se produit, vous pouvez créer un script Analyzer pour déboguer le code du module DMN et du client Test.

Après ce processus de « module Test », Enterprise Architect garantit que les éléments BPMN2.0 :: DataObject fonctionneront bien avec le module DMN.

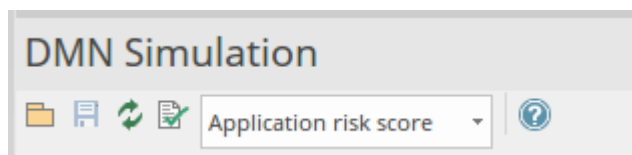
Vous configurez ensuite BPSim en chargeant des DataObjects et en attribuant des décisions du module DMN aux Propriétés BPSim, qui seront ensuite utilisées comme conditions sur les flux Séquence sortant d'une Passerelle.

Accéder

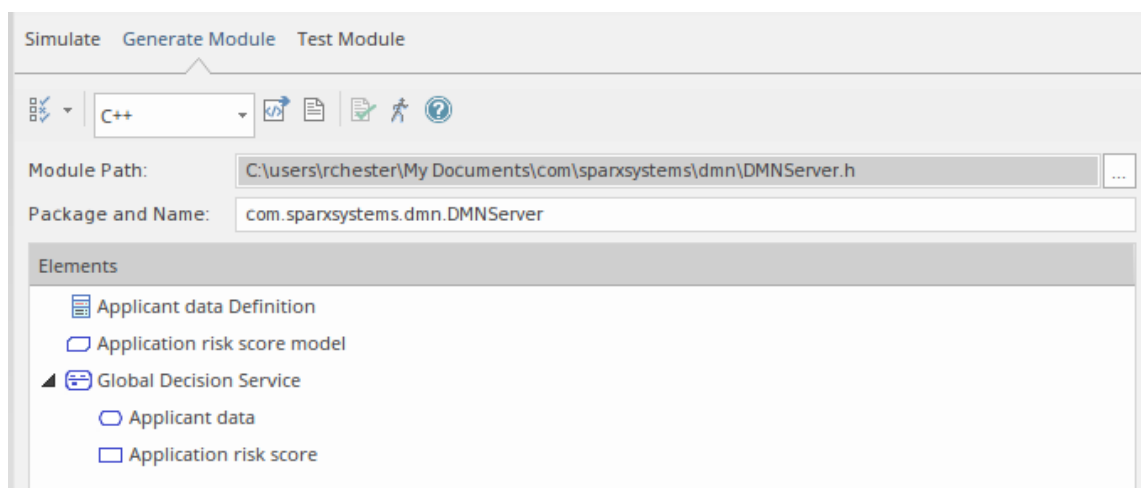
Ruban	Simuler > Décision Analysis > DMN > Open DMN Simulation > Générer Module
-------	--


Module DMN : génération de code

Dans la fenêtre Simulation DMN, sélectionnez la structure DMN à partir de laquelle vous souhaitez générer le module, dans le champ de saisie des données de la barre d'outils.



Cliquez sur l'onglet 'Générer Module', puis Ctrl+clic sur les noms des éléments DMN que vous souhaitez générer sur le serveur.




Dans le champ de saisie des données de la barre d'outils de l'onglet, sélectionnez la langue dans laquelle générer, et dans le champ « Chemin du module », cliquez sur l'icône  et accédez à l'emplacement du chemin dans lequel générer le

module (note , pour Java, le chemin doit correspondre à la structure Paquetage).

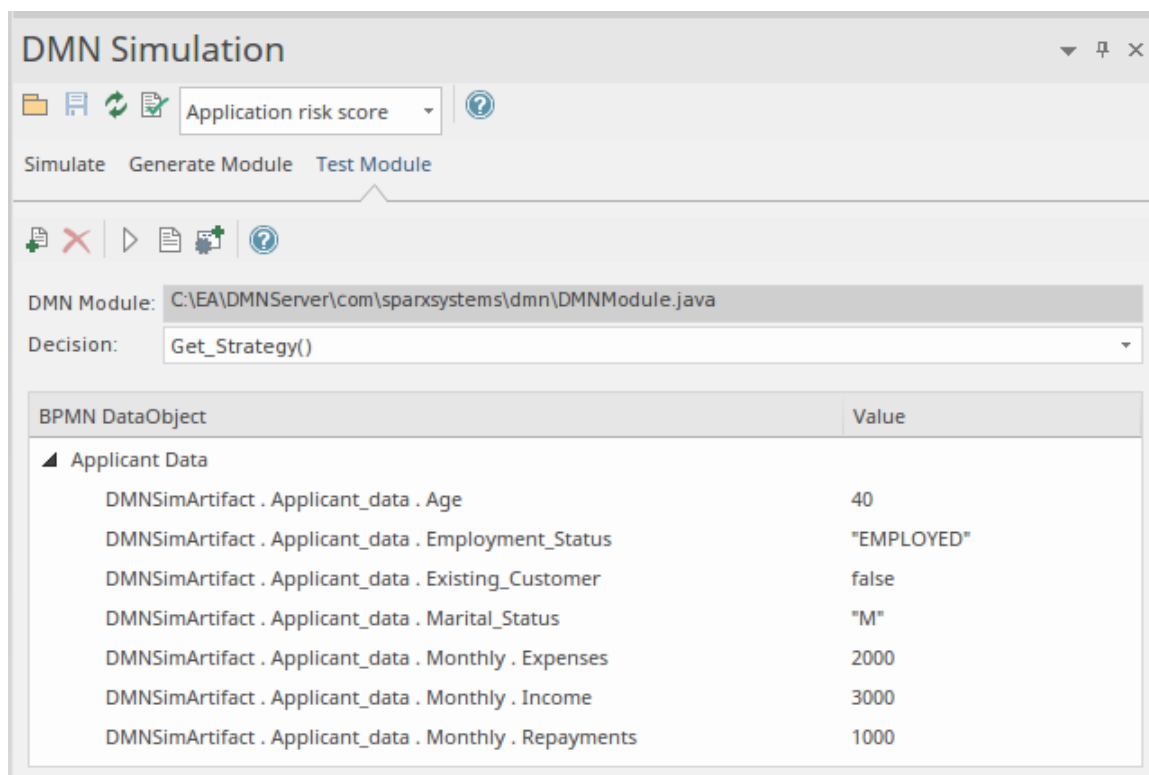
Cliquez sur le bouton Générer ().

Une fois la génération terminée, cliquez sur le bouton  pour ouvrir l'onglet ' Test Module' du module généré.

Serveur DMN : module Test

Lorsque vous utilisez le bouton  pour sélectionner l'onglet ' Test Module', le champ 'DMN Module' sera rempli automatiquement avec le chemin du serveur DMN généré du module que vous avez généré le plus récemment dans l'onglet ' Générer Module'. Si nécessaire, sur le champ ' Décision ' cliquez sur la flèche déroulante et sélectionnez la Décision souhaitée.

Cliquez sur le bouton Ajouter un objet de données BPMN () dans la barre d'outils et sélectionnez un ou plusieurs (Ctrl+Clic) objet(s) de données BPMN2.0 à ajouter à la liste dans le panneau principal.




Cliquez maintenant sur le bouton Exécuter dans la barre d'outils. Dans la fenêtre Sortie système, ce message indique que le serveur DMN et le DataObject BPMN2.0 peuvent bien fonctionner ensemble pour évaluer la décision sélectionnée :

Exécution du client Test pour le serveur DMN...

dmnServer.Application_risk_score : 133,0

Résultat : 133.0

La course s'est terminée avec succès.

S'il y a des erreurs, créez un script Analyzer en cliquant sur le bouton de la barre d'outils  et utilisez le script pour résoudre le problème.

Important : Cette étape ' Test Module' est recommandée avant d'intégrer DMNServer.java avec l' Enterprise Architect BPSim Moteur d'Exécution . Consultez la rubrique d'aide *Intégrer un module DMN dans BPSim pour Simulation* .

Génération de code et connexion à BPMN

- Générer le serveur DMN en Java, JavaScript , C++ ou C#
- Exécuter / Déboguier tests de la version Java du Serveur DMN
- Connecter le serveur DMN à l' Enterprise Architect BPSim Moteur d'Exécution

Erreurs courantes et solutions

- Types de variables : les modèles DMN utilisant le langage FEEL (Simulate with JavaScript), la saisie des variables n'est pas obligatoire ; cependant, lors de la génération de code dans des langages compilés, vous devez saisir une variable - il existe des options de menu contextuel et des valeurs de balise pour définir le type d'une variable.
- Puisqu'une expression DMN autorise les espaces, afin de clarifier les données d'entrée composites, il doit y avoir un espace avant et après le « ». dans l'expression ; par exemple, « Données du candidat ». "Âge" est valide, alors que "Données du candidat. Âge" n'est pas valide
Note que lorsque vous utilisez la fonctionnalité de complétion automatique, ce problème ne se posera pas.
- L'exécution de la validation vous aidera à localiser la plupart des problèmes modélisation ; faites-le avant la simulation et la génération de code

Notes

- La compilation avec Java nécessite un accès complet en lecture-écriture au répertoire cible ; la compilation échouera si le chemin du module est défini uniquement sur « C: » ou « C:\Program Files (x86) »

