

Blocs de la bibliothèque Item

Activity	2
Batch	9
Catch Item	13
Convey Item	15
Cost by Item	20
Create	23
Equation(I)	31
Executive.....	36
Exit.....	42
Gate.....	44
Get	47
History.....	50
Information.....	52
Query Equation (I).....	55
Queue	62
Queue Equation	67
Queue Matching	75
Read(I)	81
Resource Item.....	84
Resource Manager	88
Resource Pool.....	89
Resource Pool Release	92
Select Item In.....	95
Select Item Out.....	97
Set	100
Shift.....	103
Shutdown	106
Throw Item.....	117
Transport	119
Unbatch	124
Workstation	127
Write(I)	131

Activity



Contient une ou plusieurs entités et les transmet en fonction de leur heure d'arrivée et du délai subi.

[3] Activity <Item>

Process Cost Shutdown Preempt Advanced Status Results Contents

Item Animation BlockAnimation Comments

Processes one or more items simultaneously;
outputs each item as soon as it is finished

OK Cancel

Define capacity

Maximum items in activity: 1

Specify processing time (delay)

Delay is: a constant Delay (D): 1 time units

Define other processing behavior

Simulate multitasking activity

Use shift: Preempt when block goes off shift

Block type: Residence

Aide Left to right

Dans l'onglet **Process**, vous définissez combien d'entités peuvent être traitées à la fois, et combien de temps dure le traitement.

La durée du traitement est spécifiée dans la portion **Specify processing time (delay)** (spécifier la durée du traitement (délai)). Vous choisissez si le délai est une constante, issu du connecteur "D", la valeur d'un attribut, spécifié par une distribution ou issu d'une table de correspondance.

Si l'option **Shutdown** (Arrêts) est activée, vous pouvez utiliser un connecteur continu ou discret pour déclencher les arrêts. Si c'est par l'envoi d'une valeur (connecteur continu), le bloc tout entier est arrêté tant que la valeur sur le connecteur est ≥ 0.5 . Si c'est par l'envoi d'une entité (connecteur discret), seules les entités actuellement en traitement sont affectées, et cela n'empêchera pas de nouvelles entités d'entrer dans le bloc s'il en a la capacité.

Si vous cochez **Enable preemption** (Activer la préemption) dans l'onglet **Preempt** (préemption), des entités pourront être interrompues (libérées prématurément). Les entités interrompues sortent par le connecteur de sortie discret du bas. Sans préemption, c'est l'entité arrivée le plus tôt avec le plus petit délai est transmise la première, par le connecteur de sortie discret du haut.

Si vous choisissez que le bloc doit être contrôlé par un bloc Shift, l'horaire précisera la capacité de traitement du bloc (horaire du type Nombre), ou bien des périodes de non-fonctionnement (horaire de type on.off).

Si le délai est choisi par une table de correspondance, la ligne de la table à considérer sera spécifiée par la valeur d'un attribut sur l'entité à traiter. Une fois la ligne déterminée, les valeurs sur la ligne spécifient une distribution aléatoire, choisie dans le menu déroulant distribution. Pour un délai particulier en fonction de la valeur de l'attribut, utilisez la **Distribution** (distribution Constante) du menu.

Les connecteurs et éléments de dialogue visibles s'ajustent en fonction de vos choix. Seulement si vous cochez **Enable preemption** (Activer la préemption), vous verrez les connecteurs d'interruption sur l'icône.

Onglet Process

Maximum items in activity (nombre maxi dans l'activité) : nombre maximum d'entités que peut contenir le bloc. Cette valeur détermine le seuil de blocage de l'activité.

Shift (horaire) : cet élément n'apparaît que si le bloc est contrôlé par un horaire de type Nombre, qui prévaut sur Nombre maxi dans l'activité (cette option disparaît, car l'horaire définit cette contrainte).

Attribute (attribut) : spécifie quelle valeur d'attribut doit être utilisée pour contrôler le temps de traitement. Le menu apparaît aussi lorsque le délai provient d'une table de correspondance. Dans le cas, la valeur sert à choisir une ligne dans la table.

Delay is : a constant, from the "D" connector, an item's attribute value, specified by a distribution, from a lookup table (le délai est : une constante, issu du connecteur "D", la valeur d'un attribut, spécifié par une distribution ou issu d'une table de correspondance). Ce choix fondamental du mode de calcul du temps de traitement détermine les options de la portion **Specify processing time (delay)** (spécifier la durée du traitement (délai)).

Delay (D) (délai (D)) : affiche le délai courant. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, une unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Display string attributes (afficher les attributs chaîne) : spécifie si les attributs chaîne dans la table de correspondance doivent être affichés sous forme de chaînes ou de valeurs d'indice.

Use block seed (utiliser la base aléatoire du bloc) : en cochant cette case et en indiquant un chiffre pour la base, vous déterminez une base aléatoire locale pour les distributions calculées par ce bloc.

Distribution : permet de choisir une distribution dans une liste. En fonction de la distribution, les arguments appropriés s'affichent. Voir le bloc Random Number sur les distributions et leurs arguments.

Simulate multitasking activity (simuler une activité multitâche) : si cochée, le temps de traitement spécifié sera divisé de manière dynamique par le nombre d'entités présentes dans l'Activité. Permet de simuler facilement un opérateur ayant plusieurs tâches à effectuer et ne pouvant être que sur une tâche à la fois, ou un processeur informatique qui partage son temps sur plusieurs tâches.

Use shift (utiliser un horaire) : sélectionnez l'horaire applicable à ce bloc. L'horaire peut être soit ON/OFF, soit indiquer un nombre maximum d'entités dans le bloc.

Preempt when block goes off shift (préemption si horaire off) : si sélectionnée, le bloc forcera une interruption lorsque l'horaire est inactif.

Onglet Costs

Define processing costs (définir des coûts de traitement) : cochez cette case pour activer les informations de coût.

Processing cost (coût de traitement) : le coût ajouté à chaque entité pour chaque unité de temps passée dans l'activité. L'unité de temps utilisée pour définir ce taux doit être cohérente avec l'unité de temps utilisée dans le modèle. Le coût temporel pour chaque entité est ajouté à l'attribut "_cost" de chaque

entité. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, une unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Cost per item (coût par entité) : le coût ajouté à l'attribut "_cost" de chaque entité passant par l'activité. Ce coût est constant, quel que soit le temps passé dans le bloc.

Total cost (coût total) : le coût total associé à cette activité pour toutes les entités ayant passé ou séjourné dans le bloc.

Onglet Shutdown

Enable shutdown (activer les arrêts) : cochez cette case pour activer les informations d'arrêt et voir le connecteur d'arrêt (SD) sur l'icône.

SD (shutdown) input is from (l'entrée sur SD provient d'un) : choisit entre une entrée continue ou discrète. Cela modifie les options du menu *When signal is received at SD input...*

When signal is received at SD input, shutdown (lorsqu'un signal est reçu sur SD, arrêter) : ce menu permet de choisir quelle(s) entité(s) sont arrêtées. Dans le cas d'un signal discret, les choix sont : toutes les entités en cours, une entité au hasard, ou une entité dans la valeur d'attribut correspond à une certaine valeur. Le dernier choix fait apparaître un menu pour choisir l'attribut. Dans le cas d'un signal continu, l'unique possibilité est : tout le bloc.

Shutdown duration specified by (durée de l'arrêt spécifiée par) : option disponible uniquement dans le cas d'un signal discret. Permet de spécifier un attribut qui contrôlera l'arrêt.

When activity shuts down (en cas d'arrêt) : permet de spécifier l'action à prendre concernant les entités en cours de traitement au moment de l'arrêt. Les choix sont : **discard items that are being processed** (rebuter les entités) ; **keep items** (conserver les entités) ; **resume process after shutdown** (terminer après l'arrêt) ; **keep items** (conserver les entités), **restart process after shutdown** (recommencer après l'arrêt) ; et **finish processing items before shutting down** (terminer les entités avant de se mettre en arrêt).

Onglet Preempt

Enable preemption (activer la préemption) : active les options de préemption (interruption d'activité). Rend visible les connecteurs PEV_In et PE_ItemOut.

PE (preempt) input is from (l'entrée sur P provient d'un) : choisit si la préemption est contrôlée par une entrée continue ou discrète. Cela modifie les options du menu **When signal is received at PE input, preempt** (lorsqu'un signal est reçu sur PE, préempter).

When signal is received at PE input, preempt (lorsqu'un signal est reçu sur P, préempter) : Dans le cas d'un signal continu, les choix possibles sont :

- * **the item closest to finishing** (l'entité plus près de finir) : si sélectionné, lors d'une interruption, l'entité interrompue est celle à laquelle il reste le moins de temps de traitement.
- * **the item furthest from finishing** (l'entité plus loin de finir) : si sélectionné, lors d'une interruption, l'entité interrompue est celle à laquelle il reste le plus de temps de traitement.
- * **has the lowest priority (higher value -> lower priority)** (l'entité de plus basse priorité (valeur haute -> priorité basse)) : si sélectionné, lors d'une interruption, l'entité interrompue est celle de moindre priorité (l'entité de plus haute valeur pour sa priorité).
- * **all items currently in processing** (toutes les entités en cours) : si sélectionné, lors d'une interruption, toutes les entités présentes dans le bloc sont interrompues.

Dans le cas d'un signal discret, les choix possibles sont :

items whose attribute matches this attribute at "PE" (les entités dont l'attribut correspond à l'attribut sur "PE") : si sélectionné, lors d'une interruption, la ou les entités à préempter seront celles ayant la même valeur d'attribut que celle sur le connecteur PE.

Preempt only if block is full (interruption uniquement si bloc plein) : si coché, une interruption ne peut arriver que lorsque le nombre d'entités dans le bloc est égal à la "capacité"

Transfer item properties (transférer les propriétés) : si cochée, un tableau apparaît où vous pouvez spécifier les attributs à transférer depuis l'entité de préemption sur l'entité préemptée.

Store remaining process time in attribute (stocker délai restant dans l'attribut) : le délai restant est stocké dans l'attribut spécifié si cette case est cochée.

Use this attribute as delay (utiliser cet attribut pour temps de traitement) : utiliser l'attribut où l'on stocke le délai restant pour spécifier le temps de traitement. Si l'attribut n'existe pas ou que le connecteur d'interruption n'est pas lié, le délai du dialogue est utilisé.

Onglet Advanced Status

Advanced status connector is (le connecteur Advanced status est) :

Table Driven (défini par le tableau) : Les quatre premières colonnes de ce tableau représentent les différents états possibles de l'activité.

Busy (active) vaut true si au moins une entité réside dans le bloc pour être traitée.

Blocked (bloquée) vaut true si au moins une entité est empêchée de sortir du bloc après traitement.

Shutdown (arrêt) vaut true si l'activité a été arrêtée via le connecteur SD.

Offshift (horaire off) vaut true si l'activité es associée à un horaire actuellement inactif.

Les quatre premières colonnes du tableau listent toutes les combinaisons possibles de ces quatre états. La cinquième colonne permet de spécifier une valeur en sortie pour le connecteur Advanced Status (AS) pour chacun des 16 combinaisons possibles. Le paramétrage par défaut du connecteur AS est d'émettre 1 si l'activité est en cours de traitement, et zéro pour tous les autres états.

Bit Encoded (codé pour Gantt) : La valeur sur ce connecteur est "encodée" pour transmettre des informations sur l'état : Bloqué (bit 28), En arrêt (bit 29), Horaire (bit 30), et Longueur (bit 31). Le nombre est décodé par le bloc Gantt Chart de la bibliothèque Plotter.

Onglet Results

Length (current, average, maximum) (longueur (courante, moyenne, maxi)) : nombre (en cours, moyen, maximum) d'entités dans le bloc.

Wait (current, average, maximum) (attente (courante, moyenne, maxi)) : temps de séjour (en cours, moyen, maximum) d'une entité dans le bloc.

Preemptions (préemptions) : le nombre d'entités interrompues durant le traitement.

Shutdowns (arrêts) : le nombre d'entités ayant été arrêtées.

Offshift (entités en horaire Off) : le nombre d'entités.

Blocked (entités bloquées) : le nombre d'entités souhaitant mais ne pouvant sortir.

Utilization (utilisation) : Pourcentage du temps durant lequel le bloc a été utilisé sur l'ensemble de la simulation, en relation avec la capacité du bloc. Le temps d'utilisation pris en compte est le temps où le bloc a contenu des entités, qu'il s'agisse d'entités en cours de délai, ou d'entités ayant déjà subi le délai voulu mais retenues parce qu'aucun bloc ne les avait tirées. Ainsi, avec une capacité de 5, si le bloc traite 2 entités 100% du temps, l'utilisation est de .40 (40%). Cependant des cases à cocher permettent d'affiner ce que doit prendre en compte le taux d'utilisation. Par exemple, les entités en arrêt peuvent être considérées comme participant au taux d'utilisation, ou la capacité de l'Activité être en état utilisé lorsqu'elle est en horaire Off.

Si l'on entre dans les détails, l'utilisation est calculée par : (intégrale des entités dans l'activité) / (intégrale de la capacité de l'activité).

1.) **Numerator** (numérateur) : Les entités dans le bloc peuvent être dans divers états. Par défaut seules les entités effectivement en traitement sont comptées pour le numérateur. Et vous avez l'option d'inclure les entités bloquées, en horaire off ou arrêtées dans le numérateur.

2.) **Denominator** (dénominateur) : par défaut la capacité n'est pas affectée par les horaires ou les arrêts, et ainsi le dénominateur va augmenter durant la simulation quel que soit l'état de l'activité. Cependant si vous décochez les cases "off shift" ou "down", lorsque l'activité passe en horaire off ou en arrêt, le dénominateur n'augmentera PAS.

Activity state statistics (état de l'activité) : cinq statistiques sont suivies, en pourcentage du temps et de la capacité. Elles sont calculées en divisant l'intégrale de l'état en question par l'intégrale de la capacité de l'Activité. NOTE : si l'activité est soumise à des horaires et à des arrêts, le signal qui intervient en premier est celui qui est tracé par l'horloge, c'est à dire que si l'activité passe en horaire off, puis subit un arrêt, l'activité continuera d'accumuler du temps en horaire off jusqu'à retourner en horaire actif.

Arrivals (arrivées) : donne le nombre d'entités reçues dans le bloc.

Departure (départs) : donne le nombre d'entités qui ont quitté le bloc.

Total Cost (coût total) : le coût total associé à cette activité pour toutes les entités ayant passé ou séjourné dans le bloc.

Onglet Contents

Cet onglet existe pour les blocs Queue, Queue Equation, Resource Item et Activity, pour permettre de connaître soit le contenu du bloc à un instant donné, soit l'historique de toutes les entités qui ont traversé le bloc. Notez que cette fonction ralentit l'exécution et charge la mémoire : elle est conseillée pour la phase de débogage. Un onglet similaire existe pour le bloc Executive, centralisant tous les onglets Contents du modèle.

Enable current contents (activer le suivi du contenu) : en cochant cette case vous activez les options de l'onglet.

Table display (la table montre) : le menu permet de choisir parmi les des modes de fonctionnement de l'onglet.

Current contents mode (contenu actuel) : dans ce mode la table affiche les infos sur les blocs qui résident actuellement dans le bloc. Dans ce mode, les 5 premières colonnes de la table sont fixes. Par le bouton de redimensionnement "+/-" vous pouvez ajouter des colonnes et ainsi suivre d'autres propriétés. Colonne 1, Heure entrée, indique à quelle heure l'entité est entrée dans le bloc. Colonne 2, Arrêt, indique si l'entité est en arrêt (0 si non, 1 si oui). Colonne 3, Horaire, indique si l'entité est en horaire actif si 1, en horaire off si 0. Colonne 4, Durée arrêt, indique le temps total passé en arrêt ou en horaire off. Colonne 5, Heure fin, indique l'heure de fin du traitement. Si cette colonne est vide, c'est que l'entité est encore en traitement. Si une valeur apparaît, c'est que l'entité n'a pas pu quitter le bloc, ce qu'elle cherche à faire quand elle est terminée. Une entité non bloquée ayant fini son traitement ne figure normalement plus dans la table.

Historical log mode (historique) : dans ce mode la table affiche toutes les entités qui sont passées par le bloc durant la simulation. Cela signifie qu'une ligne est consacrée à chaque entité, et que la table peut devenir très grande. Dans ce mode aussi, les 5 premières colonnes de la table sont fixes. Les colonnes Heure entrée, Durée arrêt et Heure fin sont identique à celles du mode *Current contents*. Colonne 4, Heure sortie, affiche l'heure à laquelle l'entité a quitté le bloc. Si une entité a une valeur dans la colonne Heure fin mais pas dans heure sortie, c'est qu'elle est bloquée dans le bloc. Colonne 5, Sortie, décrit la manière dont l'entité a quitté le bloc "Normal" signifie par le connecteur habituel itemOut, "Préemption" signifie que l'entité a été préemptée.

Display contents (afficher contenu) : en mode *Current contents*, un menu apparaît sous la table, permettant de choisir quand afficher le contenu : 1.) durant la simulation, 2.) uniquement sur clic du bouton Instantané, ou 3.) à la toute fin de la simulation.

Control log time (bornes pour historique) : en mode *Historical*, cette case à cocher apparaîtra, permettant de définir une fenêtre de temps durant laquelle l'historique sera enregistré.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Animate H-block object x to y (animer les objets de bloc H x jusque y) : anime la longueur de la file sur plusieurs objets d'animation présents sur le bloc hiérarchique englobant ce bloc. La plage est celle des numéros d'objets d'animation sur l'icône du bloc hiérarchique. Chaque objet dans cette plage animera une entité différente.

Onglet Block Animation

Activity state animation (animation de l'état de l'activité) : cette option existe pour représenter les statistiques de l'état de l'activité sous forme d'un camembert. Les contrôles vous permettent de décider de la position, de la taille et quand mettre à jour le camembert.

Connecteurs

En entrée l'entité à traiter.

Le connecteur de sortie du haut sert pour les entités quittant le bloc après avoir terminé leur traitement, celui du bas sert pour les entités préemptées (il n'apparaît que si *Activate Preemption* est coché dans l'onglet *Preempt.*)

PE : ce connecteur devient continu ou discret en fonction des paramètres de l'onglet *Preempt.* Voir onglet *Préemption.*

SD (entrée) : contrôle les arrêts affectant le bloc.

Connecteurs d'entrée continus : (dans l'ordre d'apparition.)

D : Délai. A priorité, lorsqu'il est connecté, prévaut sur la valeur de *Delay* du dialogue, ou sur l'attribut.

C : Capacité. Prévaut sur le paramètre *Nombre maxi* dans l'activité.

Connecteurs de sortie continus : (dans l'ordre d'apparition.)

F : émet 1 lorsque le bloc est plein, sinon 0.

U : Utilisation. Voir description plus haut.

L : (Longueur) Nombre d'entités dans le bloc.

P : la priorité de l'entité de moindre priorité (plus haute valeur numérique) du bloc. Sert dans les cas de préemption.

CB : (Nombre bloqués) Indique le nombre d'entités bloquées par manque de capacité en aval.

Sd (sortie) : Indique le nombre d'entités en arrêt.

SS : (Horaire) Emet True (1) ou False (0) selon que le bloc utilise ou non un horaire.

D : (Traitement) quantité de temps que la dernière entité ayant quitté le bloc a passé en traitement
N'inclut pas les temps de blocage.

BT : (Bloqué) quantité de temps que la dernière entité ayant quitté le bloc a passé bloquée.

DT : (Durée en arrêt) indique le temps passé en arrêt par la dernière entité ayant quitté le bloc.

AS (Etat de l'activité) émet un nombre indiquant un état (active, en activité, en arrêt, bloquée). Ce chiffre est exploité par le bloc Gantt de la bib. Plotter.

TD (Nombre d'entités en activité) émet un nombre indiquant le nombre d'entité actuellement en cours de traitement dans le bloc.

% Busy (% actif) : intégrale du nombre d'entités en état de traitement / l'intégrale de la capacité de l'activité.

% Idle (% inactif) : intégrale de la capacité non utilisée de l'activité / l'intégrale de la capacité de l'activité.

% Blocked (% bloqué) : intégrale du nombre d'entités dans l'état bloqué / l'intégrale de la capacité de l'activité.

% Shutdown (% en arrêt) : intégrale de la capacité de l'activité passée en état arrêté / l'intégrale de la capacité de l'activité.

% Offshift (% horaire off) : intégrale de la capacité de l'activité passée en état Horaire off / l'intégrale de la capacité de l'activité.

TE : Nombre total d'entités étant entrées dans le bloc.

Animation

L'animation fournit de nombreuses informations sur l'état du bloc. Le bloc affiche une, deux ou trois petites horloges, pour montrer les entités en cours. Si la capacité du bloc est de un, la couleur de l'horloge reflète l'état du bloc. Dans le cas d'une capacité de trois ou plus, les trois horloges ont la signification suivante : une seule horloge indique une seule entité en traitement. Deux horloges, plusieurs entités en traitement. Trois horloges, le nombre maximum d'entités en traitement.

Si une entité est bloquée, la couleur de l'horloge devient rouge.

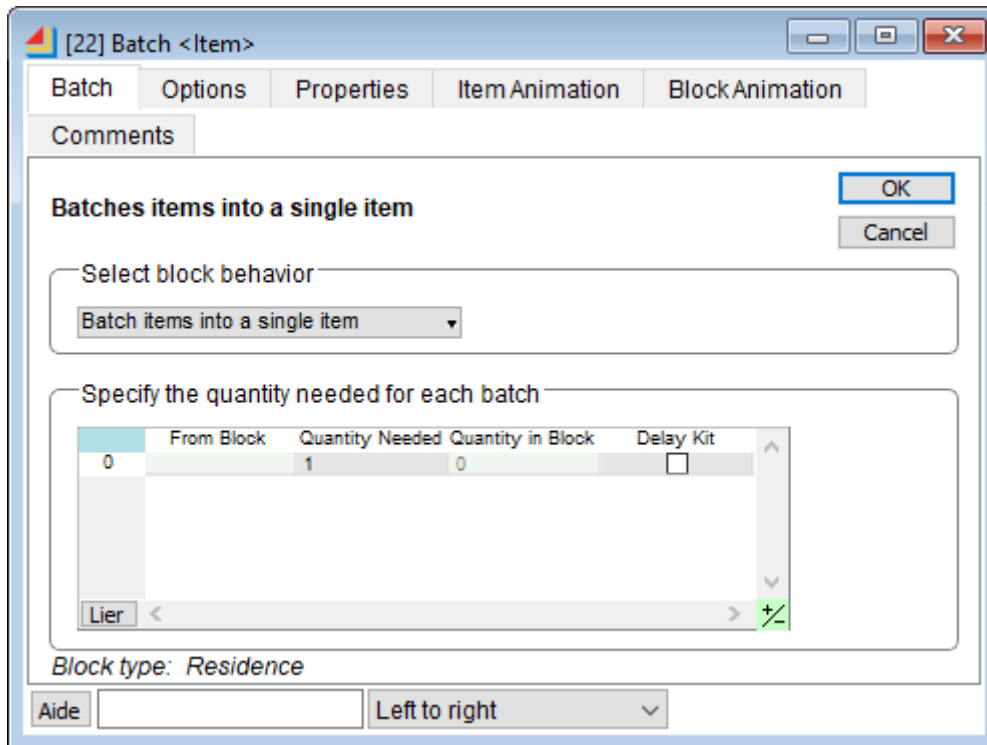
Si le bloc est en arrêt, une croix rouge barre l'icône.

Si le bloc est en horaire inactif, une croix bleue barre l'icône.

Batch



Permet de réunir en une seule entité des entités provenant de différents flux.



Utile pour synchroniser des ressources, et combiner différents sous-ensembles d'un tout. Le dialogue permet de spécifier le nombre d'entités à prélever sur chaque flux afin de produire une entité en sortie. Vous pouvez également préciser qu'on ne doit pas immobiliser les entités d'un ou plusieurs flux tant que les entités d'autres flux ne sont pas réunies.

Le bloc peut permettre aux entités d'accéder au bloc au fur et à mesure de leur arrivée (mode par défaut), ou obliger les entités de certains flux spécifiés à attendre à l'extérieur. Chaque entrée pour laquelle "En dernier" (**Delay kit**) est sélectionné, verra ses entités attendre à la porte du bloc jusqu'à ce que toutes les entités des autres entrées soient parvenues au bloc. Par exemple, si l'option "Delay kit" est sélectionnée pour la branche *a*, les entités sur *b* et *c* sont admises dans le bloc à concurrence des quantités requises, puis les entités de *a* sont immédiatement admises. Au moins un des choix de "Delay kit" doit demeurer non coché. L'exception est si tous les connecteurs doivent recevoir juste une entité : toutes les cases en dernier peuvent alors être cochées.

Dans l'onglet Propriétés vous spécifiez comment se combinent les propriétés des entités constituant le lot. Si vous sélectionnez Individualiser (**Preserve uniqueness**) dans ce bloc et dans le bloc Unbatch correspondant, les entités pourront retrouver leurs propriétés d'origine (attributs et priorités). L'option Individualiser (**Preserve uniqueness**) requiert davantage de mémoire et doit être spécifiée dans les deux blocs.

Le connecteur *demand* permet de contrôler davantage le moment de créer le lot. Une valeur TRUE sur ce connecteur peut déclencher la libération du lot, ou bien le début de la constitution du lot.

Using the Batch block with Activity Based Costing (ABC) (utilisation du Batch avec un calcul de coûts) :

Des coûts et des taux peuvent être définis pour des ressources dans divers blocs ressource. Le coût et le taux d'une ressource sont stockés dans les attributs "_cost" et "_rate" de l'entité ressource. Lorsque des ressources sont mises en lot avec une entité "matière", l'attribut "_cost" de la ressource sera ajouté à l'attribut "_cost" de l'entité. L'information de taux est stockée dans une matrice spéciale qui suit l'entité. Les autres blocs sauront que cette entité est associée à une ressource ayant un coût, et utiliseront cette information pour leurs calculs de coûts. Une fois mise en lot avec une entité, les ressources ayant un coût peuvent être libérées par le bloc Unbatch en sélectionnant l'option "Ce bloc libère les ressources ayant un coût" (**Release cost resources**).

Onglet Batch

Select block behavior (comportement du bloc) : si Mise en lot d'entités (**Batch items into a single item**) est choisie, les entités sont combinées par ordre d'arrivée (les premières d'abord). Si Mise en correspondance d'entités (**Match items into a single item**) est choisie, le bloc essaie d'associer les entités en fonction d'une valeur d'attribut. Passer d'un mode à l'autre change les éléments visibles du dialogue.

Colonne 1 **From Block** (depuis bloc) : identifie le bloc auquel est relié le connecteur correspondant.

Colonne 2 **Quantity needed** (qté requise) : nombre d'entités requises en provenance de ce connecteur.

Colonne 3 **Quantity in block** (qté accumulée) : nombre d'entités présentes dans le bloc en provenance de ce connecteur.

Colonne 4 **Delay kit** (en dernier) : lorsque cette option est sélectionnée, les entités correspondantes ne peuvent accéder au bloc que lorsque toutes les entités des autres flux ont été totalement rassemblées dans le bloc. Si ces options ne sont pas sélectionnées, les entités accèdent au bloc au fur de leur disponibilité. Cette option doit rester inutilisée pour au moins un des flux.

Match on attribute (attribut de correspondance) : cet attribut sert à apparier les entités. Les entités avec la même valeur pour cet attribut sont mises en lot ensemble.

Get batch size from attribute (attribut de taille du lot) : cet attribut sert à déterminer la taille du lot. S'applique au comportement Mise en correspondance.

Onglet Options

Preserve uniqueness (individualiser) : marque les entités constitutives d'un lot pour permettre à un bloc Unbatch de les restaurer spécifiquement, avec l'ensemble de leurs propriétés (attributs et priorité). Cette option utilise de la mémoire. Si elle n'est pas utilisée, le bloc Unbatch crée des entités en fusionnant leurs propriétés, mais en utilisant moins de mémoire. Si vous sélectionnez Preserve uniqueness dans le bloc Batch, vous devez également l'option dans le bloc Unbatch. Attention : si vous sélectionnez l'option et si vous ajoutez de nouvelles propriétés à l'entité mise en lot avant son délottage, ces nouvelles propriétés seront perdues.

Use quantity input connectors (utiliser les connecteurs pour la taille) : si cochée, fait apparaître des connecteurs d'entrée continus à utiliser pour définir les quantités requises pour chaque entrée.

Set batch size (taille définie) : la taille du lot peut être donnée dynamiquement tant que le lot n'est pas terminé (**dynamically as the batch is created**), ou dès la première entité sur chaque connecteur (**by the first item at each connector**). Si dynamiquement tant que le lot n'est pas terminé est sélectionnée, les valeurs contrôlant le nombre d'entités requises sur chaque connecteur peuvent changer dynamiquement tant que le lot n'est pas terminé. L'autre option verrouille la taille du lot à la valeur portée par la première entité entrant par cette branche.

Show connectors for item count (connecteurs de sortie pour contrôle) : cette option fait apparaître des connecteurs de sortie qui indiqueront le compte des entités pour chaque connecteur d'entrée discret.

Allow zero batch size (permettre une taille Zéro) : si cochée, zéro sera accepté sur les connecteurs d'entrée qui déterminent le nombre d'entités à prendre sur chaque connecteur. Cela permet de spécifier qu'aucune pièce d'un certain type n'est requise pour une catégorie de lot, par exemple.

Show demand connector (utiliser un connecteur Demande) : si cochée, fait apparaître un connecteur *dem.* sur l'icône. Il sert à contrôler la création des lots.

When value at demande >=0.5 (quand valeur sur demande >= 0.5) : contrôle le comportement du connecteur *demand.* Si le choix est Commencer un lot, aucune mise en lot ne commence tant qu'une valeur >= 0.5 (TRUE) sur le connecteur *demand.* Si le choix est Créer le lot, le lot est créé lorsque la valeur TRUE arrive.

Dans ce second cas, les entités peuvent entrer dans le lotteur avant que le connecteur *demand* reçoive la valeur TRUE.

Store number of items in batch in attribute (stocker la taille du lot dans l'attribut) : spécifie un attribut porté par l'entité sortant et qui contient le nombre total d'entités qui le constituent.

Onglet Propriétés

Colonne 1 **Property** (propriété) : les propriétés disponibles pour l'entité sont listées ici.

Colonne 2 **Action** : dans cette colonne vous choisissez l'action à effectuer pour chaque propriété de l'entité lors du lot. Votre choix indique comment sera exprimée cette propriété sur l'entité résultante. Les choix sont **Maximum**, **Minimum**, **Average**, **Sum**, **First at con N**, **Last at con N**, et **Count at con N** (Maximum, Minimum, Moyenne, Somme, Première sur N, Dernière sur N et Total sur conn. N). Les options n'apparaissent que pour les propriétés pour lesquelles elles sont pertinentes.

Dans les cas **Maximum** et **Minimum**, la propriété de l'entité mise en lot sera la valeur maximum ou minimum de toutes les entités constituant le lot. Dans le cas **Average**, la propriété résultante est la moyenne de toutes les propriétés constitutives. **Sum** stockera la somme de toutes les valeurs. **First at con N** et **Last at con N** adoptera respectivement la valeur de la première ou la dernière entité reçue sur le connecteur (donc l'ordre d'arrivée sera déterminant). Le dernier cas *Count at con N* donnera à la propriété le total du nombre d'entités arrivées sur ce connecteur.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Onglet Block Animation

Animation

L'icône montre sur quelles entrées sont reçues des entités. L'icône change de taille en fonction du nombre de connecteurs défini.

Catch Item



Ce bloc "reçoit" les entités envoyées par un bloc Throw. Plusieurs blocs Throw peuvent envoyer des entités à un bloc Catch. Les entités sont transmises sans l'aide d'un connecteur. Le lien entre les blocs se fait en spécifiant le label et le numéro du bloc Catch dans le dialogue. L'emploi des blocs Throw et Catch peut remplacer des liens, même d'un niveau hiérarchique à l'autre.

Onglet Catch

Catch item group (groupe de réception) : spécifie le groupe auquel appartient ce bloc Catch. Il existe un menu correspondant dans le bloc Throw qui limite au groupe spécifié la liste des blocs Catch présentés. Cela facilite l'organisation de modèles comportant un grand nombre de blocs Throw et Catch.

Departure (départs) : le nombre d'entités ayant quitté le bloc.

Onglet Throw blocks

Throw blocks associated with this Catch block (blocs Throw liés à ce bloc Catch) : cliquez sur le bouton "Rafraîchir" (**Refresh**) pour afficher la liste des blocs Throw potentiellement émetteurs vers ce bloc Catch.

Colonne 1 **Block label** (label) : labels des blocs Throw.

Colonne 2 **Block number** (numéro bloc) : numéros des blocs Throw.

Colonne 3 **Connection type** (connexion) : il y a trois types de connexions : directe si le bloc Catch est spécifié directement dans le bloc Throw; par attribut si le bloc Catch s'appuie sur un attribut; par numéro de bloc si un attribut est utilisé pour numéro de bloc Catch.

Colonne 4 (**Count** (compteur)) : affiche le nombre total d'entités étant passées par ce bloc Catch.

Refresh (rafraîchir) : met à jour la table des blocs candidats. (Sert si la structure du modèle a changé.)

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Onglet Block Animation

Connecteurs

Le bloc émet l'entité qu'il a reçue par le bloc Throw.

Connecteurs de sortie continus :

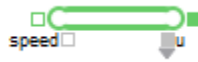
(**Count**) : indique le nombre d'entités passées.

Block number (N° bloc) : indique le numéro de bloc Throw courant. Affecter cette valeur à un attribut peut être utile lorsque *Use attribute value as Catch Item block number* est sélectionné dans le bloc Throw.

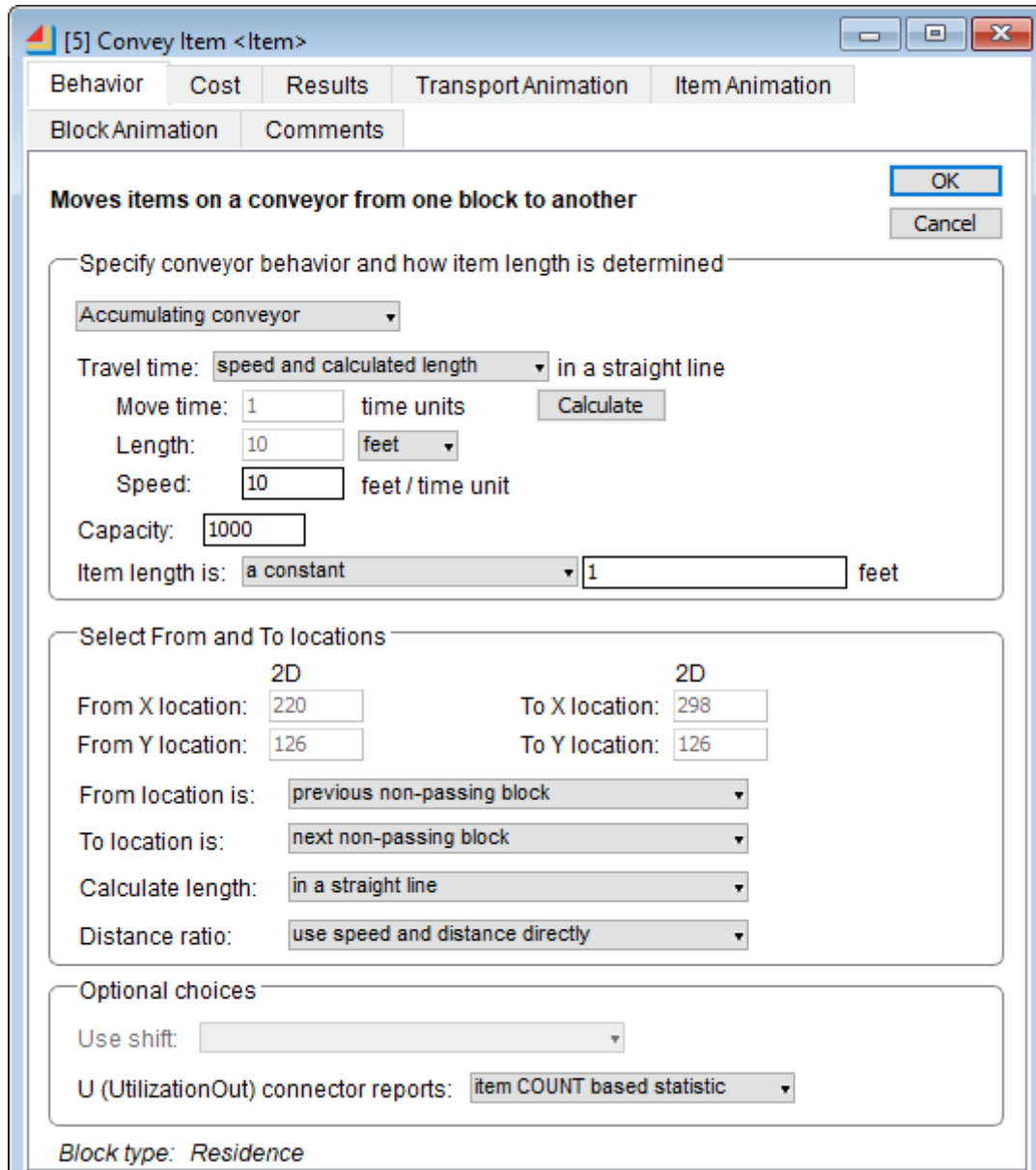
Animation

L'icône montre quand une entité arrive.

Convey Item



Modélise un convoyeur qui transporte des entités d'un point à une autre.



[5] Convey Item <Item>

Behavior Cost Results Transport Animation Item Animation

Block Animation Comments

Moves items on a conveyor from one block to another OK Cancel

Specify conveyor behavior and how item length is determined

Accumulating conveyor

Travel time: speed and calculated length in a straight line

Move time: 1 time units Calculate

Length: 10 feet

Speed: 10 feet / time unit

Capacity: 1000

Item length is: a constant 1 feet

Select From and To locations

2D		2D	
From X location:	220	To X location:	298
From Y location:	126	To Y location:	126

From location is: previous non-passing block

To location is: next non-passing block

Calculate length: in a straight line

Distance ratio: use speed and distance directly

Optional choices

Use shift:

U (UtilizationOut) connector reports: item COUNT based statistic

Block type: Residence

Dans l'onglet Behavior vous choisissez le comportement du convoyeur (accumulation ou non), et diverses options sur sa vitesse et capacité.

Si vous spécifiez que le temps de trajet est la vitesse et distance calculée (**Calculated Length and Speed**), les positions de départ de d'arrivée des entités transportées par le bloc devront être définies dans l'onglet Behavior Cette option sert notamment pour l'animation 2D du bloc.

Un convoyeur avec accumulation signifie que si des entités ne peuvent quitter le bloc à la fin du trajet, le convoyeur continue de faire progresser les entités vers le point de sortie. Sans accumulation, le convoyeur est stoppé dès que les entités en sortie ne peuvent quitter le bloc.

L'animation des entités peut bien entendu se faire en 2D. Si vous sélectionnez l'option Animation 2D montre les mouvements simultanés dans l'onglet Animation convoyeur, vous verrez bouger plusieurs entités le long des liens dans le modèle. Notez que pour activer cette option, vous devez désactiver la commande Animer les liens dans le menu Simulation, car elle interfère avec cette animation.

Les statistiques mesurées sont de deux types. Les unes s'intéressent au nombre d'entités dans le bloc, les autres à la longueur qu'occupent ces entités. Ainsi l'onglet Results présente deux valeurs Arrivées : la première est classiquement le nombre d'entités qui sont entrées dans le bloc. La seconde fait référence à la longueur totale des toutes les entités entrées dans le bloc.

Onglet Behavior

Conveyor behavior (convoyeur avec ou sans accumulation) : ce premier menu définit si le convoyeur sera avec ou sans capacité d'accumulation.

Travel Time (temps trajet) : permet de spécifier comment calculer le temps pris par une entité pour aller du point de départ au point d'arrivée. Si vous spécifiez Temps de transport (**Move time**), le paramètre Temps de transport (**Move time**) déterminera le temps de trajet. Si vous spécifiez distance et vitesse (**speed and length**), le temps de trajet sera calculé à partir de ces deux informations. Si vous spécifiez distance et vitesse calculées (**speed and calculated length**), le temps de trajet sera calculé à partir de la distance (**length**) et la vitesse (**vitesse**) une fois la distance calculée à partir des options au bas de l'onglet.

Move time (durée) : spécifie le temps total pris par l'entité pour aller du point de départ au point d'arrivée. Le menu à sa droite permet de choisir l'unité de temps du Temps de transport.

Calculate (Calculer) : le bouton calcule le temps de déplacement du bloc en fonction des options au bas de l'onglet.

Length (longueur) : spécifie la longueur totale du convoyeur, en pieds ou en mètres.

Speed (vitesse) : spécifie la vitesse des entités transportées.

Capacity (capacité) : spécifie le nombre d'entités maximum que peut recevoir le convoyeur.

Item length is (longueur entité) : ce menu permet d'indiquer comment savoir la place occupée par chaque entité. Si la longueur est lue sur un attribut (**from the attribute**) vous devez spécifier quel attribut utiliser. Si c'est une constante (**a constant**), vous devez donner la valeur de la constante. Si la longueur est fonction de longueur & capacité (**based on Conveyor length & capacity**), c'est le résultat de la longueur du convoyeur divisée par sa capacité.

From X location (position De X) : Position X de départ pour calculer la distance.

From Y location (position De Y) : Position Y de départ pour calculer la distance.

To X location (position Vers X) : Position X d'arrivée pour calculer la distance.

To Y location (position Vers Y) : Position Y d'arrivée pour calculer la distance.

From location is (position De est) : ce menu permet d'indiquer comment sera donnée la position de départ (De) pour le transport de l'entité. Les options sont : valeur donnée pour X et Y; position du bloc; bloc hiérarchique supérieur; bloc précédent et bloc stockeur précédent. La position est donnée par des coordonnées ou définie par les emplacements des blocs.

To location is (position Vers est) : ce menu permet d'indiquer comment sera donnée la position d'arrivée (Vers) pour le transport de l'entité. Les options sont : valeurs données pour X et Y, position du bloc, bloc H supérieur du bloc suivant, bloc suivant, et bloc stockeur suivant. La position est donnée par des coordonnées ou définie par les emplacements des blocs.

Calculate distance (calculer distance) : en suivant les liens (**along connections**) inclura les lignes à angle droit dans le calcul de la distance de transport. En ligne droite (**in a straight line**) ignorera les liens. Notez que cette option affecte aussi la manière dont les entités sont animées, car en suivant les liens fera que les entités suivront les liens en animation 2D.

Distance Ratio (ratio distance) : ce menu permet de spécifier le ratio entre les pixels sur la table de travail, et la longueur du transport. Servira à contrôler comment les longueurs définies dans le dialogue du bloc affecteront l'animation (2D). Les options sont : déduit de vitesse et distance (**Use speed and distance directly**) ; utiliser un ratio de (**Use distance ratio of**). Déduit de vitesse et distance utilisera un ratio de 1. (i.e. un pixel égal un mètre/pied.). Vous pouvez également spécifier votre ratio.

Use Shift (utiliser l'horaire) : sélectionnez l'horaire s'appliquant à ce bloc (de type ON/OFF ou indiquant un nombre maximum d'entité dans le bloc).

U (UtilizationOut) connector reports (connecteur U (UtilizationOut) mesure) : ce menu permet de choisir si le connecteur U s'appuie sur le nombre d'entités ou la longueur pour calculer l'utilisation du convoyeur. Voir l'onglet Results.

Onglet Cost

Define Costs (définir des coûts) : active les options de coût du bloc.

Processing cost (coût de transport) : le coût ajouté à chaque entité par unité de temps passé dans le transport. L'unité de temps définissant ce coût doit être cohérente avec les unités de temps utilisées dans le modèle. Tant que cet élément de ressource est associé à une entité, l'entité accumule le coût indiqué. Le coût est ajouté à l'attribut "_cost" de l'entité. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, cette unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Cost per item (coût par entité) : coût ajouté à l'attribut "_cost" de chaque entité passant par ce bloc. C'est une constante, quelle que soit la durée du transport.

Total cost (coût total) : le coût total accumulé par ce bloc durant la simulation.

Onglet Results

Statistiques relatives au NOMBRE d'entités

Current count (comptage en cours) : le nombre d'entité dans le bloc à cet instant de la simulation. (ou en fin de simulation, si elle est terminée.)

Average count (comptage moyen) : nombre moyen d'entités dans le bloc.

Wait (current, average, ...) (attente (en cours, moyen...)) : le temps passé dans le bloc par l'entité qui vient de le quitter.

Arrivals (count) (arrivées) : nombre d'entités entrées dans le bloc.

Departures (count) (départs) : nombre d'entités ayant quitté le bloc.

Utilization (count) (utilisation) : pourcentage du temps où le bloc a été utilisé sur l'ensemble de la simulation. Dépend de la capacité : par exemple, si la capacité est de 5, et que le bloc transporte 2 entités 100% du temps, l'utilisation est de .40 (40%).

Total cost (coût total) : le coût total d'attente de toutes les entités passées par le bloc.

Statistiques relatives aux LONGUEURS

Length (current, average, ...) (longueur (en cours, moyen...)) : la longueur des entités sur le convoyeur.

Arrivals (arrivées) : la longueur totale de toutes les entités entrées dans le bloc.

Departures (départs) : la longueur totale de toutes les entités ayant quitté le bloc.

Utilization (utilisation) : taux d'utilisation pour la longueur, calculé par la longueur de convoyeur disponible dans le temps.

Onglet Transport Animation

From X location (position X De) : les coordonnées X de la position de départ pour l'entité 2D. Peut être défini directement lorsque "Position De est : valeur donnée pour X et Y", sinon est calculé par le bloc. Cette option est désactivée si "*Travel time is speed and calculated length*" est sélectionné. Dans ce cas, la position De est déterminée par le champ du même nom.

From X location (position Y De) : les coordonnées Y de la position de départ pour l'entité 2D. Peut être défini directement lorsque "Position De est : valeur donnée pour X et Y", sinon est calculé par le bloc. Cette option est désactivée si "*Travel time is speed and calculated length*" est sélectionné. Dans ce cas, la position De est déterminée par le champ du même nom.

To X location (position X Vers) : les coordonnées X de la position d'arrivée pour l'entité 2D. Peut être défini directement lorsque "Position Vers est : valeur donnée pour X et Y", sinon est calculé par le bloc. Cette option est désactivée si "*Travel time is speed and calculated length*" est sélectionné. Dans ce cas, la position Vers est déterminée par le champ du même nom.

To Y location (position Y Vers) : les coordonnées Y de la position d'arrivée pour l'entité 2D. Peut être défini directement lorsque "Position Vers est : valeur donnée pour X et Y", sinon est calculé par le bloc. Cette option est désactivée si "*Travel time is speed and calculated length*" est sélectionné. Dans ce cas, la position Vers est déterminée par le champ du même nom.

From location is (position De est) : ce menu sélectionne la position de départ pour l'animation du transport. Les options sont :

- **Entered X and Y location** (valeur données pour X et Y) : entrez les positions X et Y dans les champs Position X De et Position Y De. Le mouvement des entités commence à ces coordonnées
- **Block location** (position du bloc) : la position du bloc est utilisée pour point de départ de l'animation
- **Enclosing hierarchical block** (bloc hiérarchique supérieur) : la position du bloc hiérarchique englobant ce bloc est utilisée pour point de départ de l'animation. Si le point d'arrivée est calculé (bloc suivant, bloc stockeur suivant), ce bloc doit être le dernier avant que l'entité ne quitte le bloc hiérarchique.
- **Previous block** (bloc précédent) : le point de départ de l'animation est le premier bloc en amont.
- **Previous non-passing block** (bloc stockeur précédent) : le point de départ de l'animation est le premier bloc stockeur ou de décision placé en amont. Cette option sautera tous les blocs passeurs entre ce bloc et le précédent bloc non passeur.

To location is (position Vers est) : ce menu sélectionne la position d'arrivée pour l'animation du transport. Les options sont :

- valeur données pour X et Y : entrez les positions X et Y dans les champs Position X Vers et Position Y Vers Le mouvement des entités se termine à ces coordonnées.
- position du bloc : la position du bloc est utilisée pour point d'arrivée de l'animation.
- bloc hiérarchique supérieur du bloc suivant : a position du bloc hiérarchique englobant ce bloc est utilisée pour point d'arrivée de l'animation. Si le point d'arrivée est calculé (bloc suivant, bloc stockeur suivant), ce bloc doit être le dernier avant que l'entité ne quitte le bloc hiérarchique.
- bloc suivant : le point d'arrivée de l'animation est le premier bloc en aval.
- bloc stockeur suivant : le point d'arrivée de l'animation est le premier bloc stockeur ou de décision placé en aval. Cette option sautera tous les blocs passeurs entre ce bloc et le bloc suivant non passeur.

2D animation shows simultaneous item movement (animation 2D montre les mouvements simultanés) : si cochée, plusieurs entités se déplaceront simultanément dans le modèle 2D. L'animation peut être soit en ligne droite, en partant de la "position De" et se terminant à la "position Vers". Soit en suivant les liens tracés entre ces deux positions. Désactivez la commande Animer les liens du menu Simulation si vous choisissez cette option, sinon l'animation bloc à bloc et l'animation simultanée

tenteront de se dérouler en même temps. Si vous choisissez "*Travel time is speed and calculated length*" dans l'onglet Behavior, l'animation suivra le choix pour "Calculer distance".

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Onglet Block Animation

Connecteurs

En entrée l'entité à transporter.

En sortie les entités inchangées, à la fin de leur transport.

Connecteurs d'entrée continus : (dans l'ordre d'apparition.)

Speed (vitesse) : si relié, prévaut sur la vitesse indiquée dans le dialogue.

Connecteurs de sortie continus : (dans l'ordre d'apparition.)

Entry (entrés) : le nombre d'entités entrées dans le convoyeur.

Exit (sortis) : le nombre d'entités ayant quitté le convoyeur.

L (length) (Longueur) : indique le nombre d'entités présentes dans le bloc.

F (Full) (plein) : émet 1 lorsque le bloc est plein, sinon 0.

SD (shutdown) (Arrêt) : émet TRUE si le bloc est arrêté.

B (blocked) (Blocage) : émet TRUE si le bloc est bloqué.

Accum : Emet TRUE (1) ou FALSE (0) selon que le convoyeur est avec accumulation ou non.

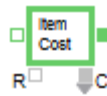
Animation

L'icône montre le nombre d'entités présentes dans le bloc.

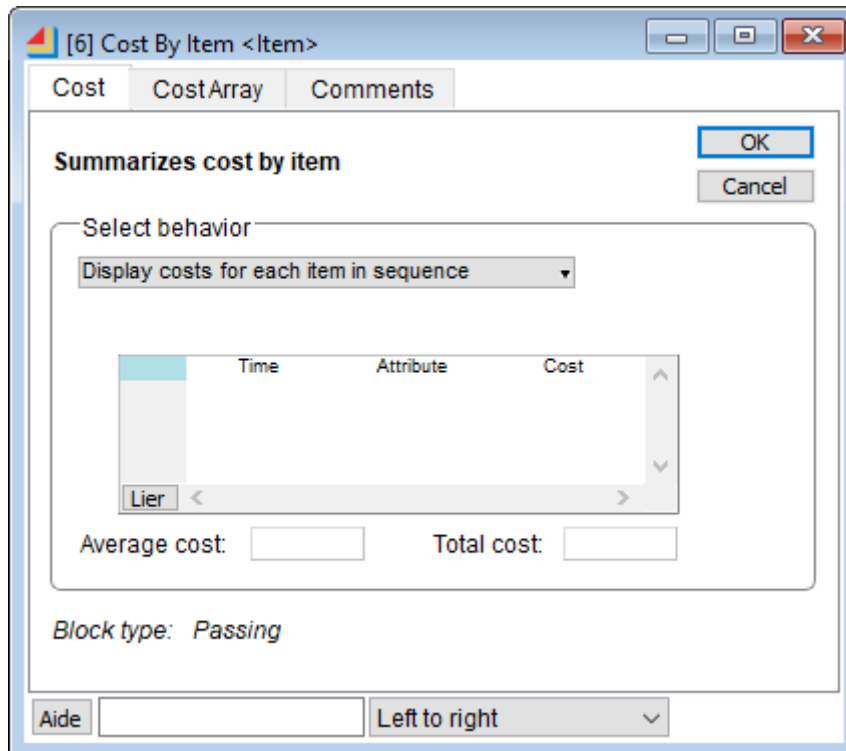
Voir les onglets d'animation pour les spécificités d'animation du bloc.

Si vous sélectionnez l'option **2D animation shows simultaneous item movement** dans l'onglet Transport Animation, vous verrez bouger plusieurs entités le long des liens dans le modèle. Notez que pour activer cette option, vous devez désactiver la commande Animer les liens dans le menu Simulation.

Cost by Item



Affiche les coûts des accumulateurs de coûts passant par le bloc. En utilisant un attribut de tri ou le connecteur R, le nombre de contributeurs, le coût moyen et le coût total peuvent être calculés pour différents types d'entités.



Le bloc possède trois modes, permettant de voir différentes informations de coût.

Display costs for each item in sequence (afficher le coût pour chaque entité en séquence) montre les coûts pour chaque entité singulièrement. **Summarize costs by attribute value** (synthétiser les coûts par valeur d'attribut) classera les coûts par familles. **Summarize costs by connector value row** (synthétiser les coûts par valeur reçue sur le connecteur ligne) spécifie dans quelle ligne de la table il convient d'accumuler les informations de coût.

Note : vous devez relancer le modèle lorsque vous changez de type de tableau dans le dialogue.

Onglet Cost

Display costs for each item in sequence : le coût de chaque entité passant par le bloc est ajouté à la colonne "Coût Total".

Summarize costs by attribute value : l'attribut nommé dans Nom d'attribut sera lu et les informations de coût pour l'entité seront ajoutées pour la ligne dont le numéro est la valeur de cet attribut. Cela permet facilement de connaître des coûts par type d'entité.

Summarize costs by connector value row : la valeur reçue sur le connecteur R sera lue à l'arrivée de l'entité et les informations de coût pour l'entité seront ajoutées dans cette ligne. Attention, la première ligne correspond à la valeur 0.

Attribute name (nom d'attribut) : si **Display costs for each item in sequence** est sélectionné, la valeur de l'attribut nommé dans ce champ sera listée dans le tableau sous le titre "Attribute". Si "N° de ligne = valeur d'attribut" est sélectionné, la valeur de l'attribut nommé dans ce champ sera utilisée pour trier les entités passant par le bloc.

En-têtes de la table Coût

Throughput (contributeurs) : le nombre d'accumulateurs de coût ayant contribué aux statistiques de coût pour cette ligne particulière (uniquement si **Summarize costs by attribute value** est sélectionné)

Average Cost (coût moyen) : le coût moyen des accumulateurs de coût ayant contribué aux statistiques de coût pour cette ligne particulière (uniquement si **Summarize costs by attribute value** est sélectionné).

Total cost (coût total) : le coût total des accumulateurs de coût ayant contribué aux statistiques de coût pour cette ligne particulière (uniquement si **Summarize costs by attribute value** est sélectionné).

Attribute (attribut) : la valeur de l'attribut nommé dans **Attribute name** (nom d'attribut) sera listé dans cette colonne (uniquement si **Display costs for each item in sequence** est sélectionné).

Cost (coût) : le coût de l'accumulateur de coût passé par le bloc au moment indiqué par la colonne **Time** (temps) (uniquement si **Display costs for each item in sequence** est sélectionné).

Time (heure) : l'heure à laquelle l'accumulateur de coût est passé par le bloc (uniquement si **Display costs for each item in sequence** est sélectionné)

Average Cost (coût moyen) : le coût moyen de tous les accumulateurs de coût passés par ce bloc.

Total cost (coût total) : le coût total de toutes les entités passées par ce bloc.

Onglet Cost Array

Table des coûts : tableau contenant les informations de coût et de ressources pouvant être associées à chaque accumulateur de coût. Cette table sert surtout pour du débogage. Elle se compose des colonnes décrites ci-dessous.

Time (heure) : heure de passage de l'entité.

Item Index (index) : numéro d'index de l'entité.

Cost (coût) : valeur de l'attribut "_cost" de l'entité. Cet attribut stocke les coûts accumulés par l'entité.

Rate (taux) : valeur de l'attribut "_rate" de l'entité. Cet attribut stocke pour l'entité les coûts d'attente (stockage) par unité de temps.

Type : si type vaut 1, l'entité est considérée comme un "accumulateur de coût" et accumule des coûts. Si type vaut 2, l'entité est considérée comme une "ressource" et ajoutera son coût à l'entité à laquelle il est associé.

ResRate1 : le coût par unité de temps de la première ressource mise en lot avec l'accumulateur de coût.

ResNum1 : indique le nombre de ressources de coût actuellement mises en lot avec l'accumulateur de coût ainsi que le connecteur (0, 1 ou 2) par lequel ces ressources sont initialement entrées dans le bloc Batch. Cet élément a la forme numéro connecteur* 100 + nombre dans le lot.

ResRate1 : le coût par unité de temps de la seconde ressource mise en lot avec l'accumulateur de coût.

ResNum1 : comme NbRes1, mais stockant les informations pour un autre type de ressource.

PoolRate : coût total par unité de temps pour tous les éléments d'un Pool actuellement utilisés par l'accumulateur de coûts.

AGVRate : coût par unité de temps pour tous les filoguidés actuellement associés à l'entité.

OrigCost : coût d'origine de l'accumulateur de coût mis en lot par un bloc Batch de la bibliothèque **Item**. Sert à déterminer la correcte allocation du coût si les accumulateurs de coûts sont par la suite éclatés par un bloc Unbatch.

Connecteurs

L'entrée est l'entité à observer.

La sortie est la même entité intacte.

R : ligne de la table dans laquelle ajouter les informations de coût pour l'accumulateur de coût courant. Si connecté, l'option **Summarize costs by connector value row** doit être sélectionnée dans le choix du mode de synthèse de l'onglet "Cost".

Connecteurs de sortie continus : (dans l'ordre d'apparition)

C : coût de l'accumulateur de coûts passant par le bloc.

A : coût moyen de tous les accumulateurs de coûts passés par le bloc.

T : coût total de tous les accumulateurs de coûts passés par le bloc.

Create



Fournit des entités (ou des valeurs numériques) à une simulation discrète à des intervalles de temps donnés.

[9] Create <Item>

Create Options Item Animation Block Animation Comments

Creates items and values randomly or by schedule

Select block behavior

Create items randomly Time units: generic*

Configure random time between arrivals (TBA)

Create items using: one random distribution

Specify a distribution for TBA

Exponential Plot Sample

mean: 1

location: 0

Item information

Item quantity (Q): 1

Max items: Infinity

Total created: 0

Total quantity: 0

Total cost:

Block type: Residence *model default

Aide Left to right

Le comportement de base est défini par le premier menu : Crée des entités aléatoirement (**create items randomly**), Crée des entités par programme (**create items by schedule**), Crée des entités infiniment (**create items infinitely**), Crée des valeurs aléatoirement (**create value randomly**) et Crée des valeurs par programme (**create value infinitely**). Le connecteur de sortie s'adapte à ce qui est créé (entité ou valeur). Crée aléatoirement permettra de choisir une distribution qui calculera l'intervalle entre arrivées. Crée par programme affiche une table où vous entrez un programme d'arrivées. Crée des entités infiniment produit un flux infini d'entités.

Create items infinitely doit être utilisé avec prudence. Dans un tel cas, le bloc crée une entité dès qu'elle peut être acceptée en aval. S'emploie lorsque la source de matière n'est pas une contrainte, mais est contrôlée par la capacité aval. Les blocs juste après le Create seront par exemple le bloc Gate (bib. Item) ou Interchange (bib. Rate). Ne prévoyez pas juste en aval une capacité infinie (une file d'attente), car cela provoquerait une boucle infinie.

Le bloc produit des entités selon les intervalles de temps spécifiés. Le bloc est un bloc "pousseur" et doit normalement être suivi d'un bloc Queue ou Resource Item, lorsqu'il sert à fournir des entités au modèle. Dans le cas contraire, vous pouvez perdre certaines des entités générées. Si un débit d'arrivée égal ou inférieur à 0 se produit, les entités sont générées sur le champ (au moment où la valeur 0 est reçue). Vous pouvez également spécifier le nombre d'entités créées à chaque événement, dans le dialogue ou par le connecteur Q.

Les paramètres des temps d'arrivées de la distribution sont établis dans le dialogue. Les distributions aléatoires disponibles sont : Uniforme (entier ou réel), Beta, Binomiale, Cauchy, Khi 2, Constante, Erlang, Exponentielle, Valeur Extrême type 1a, Valeur Extrême type 1b, Gamma, Géométrique, Hyper

Exponentielle, Hyper Géométrique, Normale inverse, Weibull inverse, Johnson SB, Johnson SU, Laplace, Logarithmique, Logistique, Log-Logistique, Binomiale négative, Normale, Pareto, Pearson type V, Pearson type VI, Poisson, Fonction Puissance, Rayleigh, Triangulaire, Weibull, ou Table empirique. On peut également demander une émission d'entités à intervalles constants. La distribution empirique peut être décrite par des points et peut s'interpréter comme discrète, échelonnée, ou interpolée. Les connecteurs d'entrée permettent de modifier les paramètres de la distribution aléatoire en cours de simulation.

Voir plus bas la description des Distributions.

Onglet Create

Select block behavior (choisit le comportement du bloc) : contrôle le comportement général du bloc : Crée des entités aléatoirement; Crée des entités par programme; Crée des entités infiniment; Crée des valeurs aléatoirement; Crée des valeurs par programme. Voir les détails ci-dessus.

Time units (unité de temps) : spécifie l'unité à utiliser dans le bloc. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, une unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Configure les arrivées aléatoires :

Items created using one random distribution (entités créées par une distribution) : vous sélectionnez une distribution pour définir l'intervalle entre deux arrivées d'entités.

Items created using multiple random distribution (entités créées par plusieurs distributions) : avec cette option, n'importe quel nombre de distributions aléatoires peut être à l'origine de la création d'une entité. Vos distributions doivent être spécifiées via une table de base de données ExtendSim où chaque enregistrement représente une distribution différente.

Distribution ID : c'est l'attribut où stocker l'ID de la distribution qui a créé cette entité. La valeur de l'attribut est prise dans le champ ID.

ID Field (champ ID) : c'est identifiant de vos distributions. Cette information est stockée sous forme d'attribut sur l'entité qui vient d'être créée, de sorte que vous puissiez savoir d'où est issue l'entité.

First arrival field (champ 1re arrivée) : champ servant à définir l'heure à laquelle est créée la première entité pour chaque distribution. Ce champ est obligatoire, mais peut être aléatoire ou constant.

TBA field (champ TBA) : champ servant à spécifier la distribution utilisée pour définir l'intervalle entre arrivées. C'est un champ en option, car vous pouvez souhaiter créer la première entité à partir du champ 1re arrivée.

Max arrivals field (champ max arrivées) : champ servant à spécifier le nombre maximum d'entités que chaque distribution peut créer.

Match field (champ filtre) : ce champ s'emploie conjointement avec le suivant pour filtrer des enregistrements dans la table des distributions. Seuls les enregistrements dont le champ filtre a la valeur de filtre indiqué seront utilisés. Ces options servent classiquement lorsque vous avez une grande table contenant toutes les distributions utilisées par plusieurs blocs Create. Chaque Create sera associé à certains enregistrements.

Match value (valeur filtre) : ce champ s'emploie conjointement avec le précédent pour filtrer des enregistrements dans la table des distributions. Seuls les enregistrements dont le champ filtre a la valeur de filtre indiqué seront utilisés.

Specify a distribution for time between arrivals (TBA) (spécifie l'intervalle entre arrivées dans la distribution) : permet de choisir la distribution qui calculera l'intervalle entre arrivées. Les arguments varient en fonction de la distribution. Voir plus bas la description des distributions.

Informations sur l'entité

Item quantity (Q) (quantité) : La quantité associée à chaque entité. Le connecteur Q prévaut sur cette valeur.

Max items (max entités) : le nombre total d'entités qui ont été créées par le bloc

Total quantity (quantité totale) : la somme des quantités des entités qui ont été créées par le bloc. Si toutes les entités avaient une Quantité de 1, ce nombre est égale au Total créées.

Total created (total créées) : le nombre total d'entités qui ont été créées par le bloc

Total cost (coût total) : le coût de toutes les créations (Coût par entité * total créées).

Informations sur la valeur

Value output (V) (valeur émise (V)) : dernière valeur émise.

Num vals created (nb valeurs créées) : le nombre total de valeurs qui ont été créées par le bloc.

Total sum (somme valeurs) : la somme des valeurs qui ont été créées par le bloc.

Programme des heures d'arrivée

	_Create Time	_Item Quantity	_Item Priority	None	None	None
1	0	1	1			

Lier < > +/-

Create times are in: simulation time format

Repeat the schedule every 10 generic* time units

Total cost:

Colonne 1 : **Create time** (heure création) : spécifie l'heure à laquelle créer l'entité ou la valeur associée à cette ligne de la table. Notez que les temps dans cette colonne sont absolus si le connecteur *start* n'est pas relié et que Répéter toutes les N n'est pas cochée. Sinon les temps de création sont relatifs au temps de départ ou au cycle.

Colonne 2 : **Property 1, defaults to Quantity** (propriété 1, quantité par défaut) : spécifie la Quantité à affecter aux entités. Peut servir pour une autre propriété.

Colonne 2 : **Value** (Valeur) : si le programme émet des valeurs, la seconde colonne définit ces valeurs.

Colonne 3 : **Property 2, defaults to Priority** (propriété 2, priorité par défaut) : spécifie la Priorité à affecter aux entités. Peut servir pour une autre propriété.

Colonnes 4, 5, 6 : **Property** (Propriété) : sert à spécifier une propriété (facultative) à l'entité créée.

Create times are in (temps de création en) : format calendaire ou temps de la simulation. Le format calendaire ne peut être choisi que si le modèle utilise globalement les dates calendaires (voir dialogue Paramètres de la simulation).

Repeat the schedule every N (répéter toutes les N) : spécifie si et à quelle fréquence le programme doit être répété. Si activé, les heures du programme sont relatives. N doit être supérieur à la plus grande valeur d'Heure du programme.

Onglet Options

Options pour la création aléatoire

Maximum items generated (nombre maxi d'entités créées) : si la case est cochée, le générateur ne crée que le nombre d'entités spécifié.

Maximum items creation events (nb maxi d'événements de génération) : si la case est cochée, cela impose une limite au nombre d'événements de création. Dans ce cas, la quantité des entités est ignorée.

Interarrival time changes occur immediately (Récalcul immédiat si connecteurs changent) : si la case est cochée, les nouvelles valeurs sur les connecteurs 1, 2, 3, ou V feront que le générateur recalculera immédiatement l'heure de la génération d'une nouvelle entité, en fonction de ces nouvelles valeurs. Sinon, les nouvelles valeurs ne seront prises en compte que lorsque la prochaine entité prévue aura été générée.

Stop simulation if items are unable to leave (arrêter la simulation si les entités sont bloquées) : normalement, si les entités ne sont pas acceptées par l'aval, elles sont supprimées. Si la case est cochée, un message signalera le problème et la simulation sera interrompue.

Timing Attribute (attribut de timing) : crée un attribut pour l'entité, et qui contiendra des informations de timing. Cet attribut reçoit la valeur de `currentTime` au moment où l'entité est créée. Il peut s'employer dans un bloc Information pour calculer un temps de cycle, ou de diverses manières.

Use shift (utiliser l'horaire) : sélectionner l'horaire s'appliquant à ce bloc (de type ON/OFF).

Item quantity (Q) (quantité) : La quantité associée à chaque entité. Le connecteur Q prévaut sur cette valeur.

No item at start time (pas de génération à $t=0$) : si cette case n'est pas sélectionnée, le Create produit sa première entité au temps 0. Si la case est cochée, la première entité est émise strictement d'après la distribution.

No value at start time (pas de valeur à $t=0$) : si ce choix est sélectionné, la première valeur sera émise strictement selon la distribution. Sinon, le bloc émet sa première valeur en début de simulation.

Use block seed (base aléatoire du bloc) : détermine un nombre pour définir la base aléatoire employée par ce bloc. La valeur finale est la base du bloc + le numéro du bloc. Si la case n'est pas cochée, la base est le numéro du bloc (un identifiant unique) +1. Dans la plupart des cas, chaque bloc devant générer des valeurs aléatoires devrait avoir sa propre base aléatoire. Le dialogue Paramètres permet de chercher les bases aléatoires en doubles.

Show next even connector (voir connecteur PE) : si cette case est cochée, le connecteur devient visible dans le coin inférieur droit de l'icône, et émettra l'heure du prochain événement de création prévu par le bloc.

Define Item costs (définir des coûts d'entité) : cocher cette case pour activer les options de coût.

Waiting cost per time units (coût d'attente par unité de temps) : le coût ajouté à chaque entité pour chaque unité de temps passée à attendre. L'unité de temps utilisée pour définir ce taux doit être cohérente avec l'unité de temps utilisée dans le modèle. Le coût est calculé dans les files d'attente pour chaque entité et est ajouté à l'attribut "`_cost`" de chaque entité. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, une unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Cost per item (coût par entité) : le coût ajouté à l'attribut "`_cost`" de chaque entité générée.

Total cost (coût total) : le coût total associé au générateur pour toutes les entités créées (coût par entité * nombre total d'entités créées).

Show connector names (voir noms des connecteurs) : détermine si les noms des connecteurs apparaissent ou non sur l'icône.

Options de création d'entités par programme

Start connector (connecteur départ) : les options de ce menu dépendent de choix divers. Il contrôle le comportement du connecteur *start*, et les options sont les suivantes : *Follow schedule* (Selon programme); *Generate one item per message* (une entité/valeur par message); *Generate one item per event* (une entité/valeur par événement). Selon programme va simplement lancer le programme lorsque le connecteur reçoit une valeur TRUE ou un message de connecteur. Les autres options contrôlent en fait la réponse à cette sollicitation : si vous spécifiez une valeur/entité par événement, il y a une seule création, même si le connecteur reçoit plusieurs messages de connecteur. Si vous spécifiez une valeur/entité par message, plusieurs messages au même événement généreront plusieurs créations.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Onglet Block Animation

Connecteurs

La sortie est discrète ou continue, en fonction du comportement de base du bloc.

Q (connecteur de sortie continu) : émet la valeur de Quantité de la dernière entité créée.

Connecteurs d'entrée continus : (visibles en fonction des options.)

Q : spécifie la quantité de l'entité à créer, prévalant sur l'élément de dialogue Quantité. (Dans le cas de Création des entités uniquement)

V : spécifie la valeur à émettre par le bloc. (Dans le cas de Création de valeurs uniquement)

Start (départ) : sert à déclencher le programme dans le cas d'une Création par programme. (Voir le menu Connecteur *Start* de l'onglet Options)

Match (filtre) : n'apparaît que dans le cas de plusieurs distributions. Sa valeur est comparée à celles du *Match field* dans la table des distributions, et sert à filtrer des enregistrements dans la table.

Les autres connecteurs d'entrée dépendent de la distribution. Dans tous les cas, la valeur sur le connecteur prévaut sur celle donnée dans le dialogue.

Animation

Un petit cercle vert clignote chaque fois qu'une entité est générée.

Lois de distribution

Beta : une distribution continue ayant des limites haute et basse finies. Les deux paramètres 'forme' peuvent servir à générer une grande variété de types de densité bornées. La limite basse est le paramètre Position, la limite haute est le maximum plus le paramètre Position. Cette distribution est souvent utilisée pour une estimation grossière en l'absence de données, pour une distribution d'une proportion aléatoire, ou pour une durée de traitement. La distribution uniforme est un cas particulier de la distribution beta (les deux paramètres 'forme' valent un).

Binomial (binomiale) : nombre de succès en tirant (2) N entités dans une population où (1) Prob est la fraction de la population qui possède la caractéristique désirée et où N est un entier. Cette distribution peut montrer par exemple le nombre de pièces défectueuses dans un lot de N, ou la probabilité qu'un certain nombre de personnes guérissent d'une maladie.

Cauchy : la distribution Cauchy est une distribution continue non bornée ayant un pic au centre et des queues très importantes (bien plus qu'une distribution Normale. Lambda est un paramètre d'échelle.

Chi Squared (Khi 2) : la distribution Khi 2 est une distribution continue bornée à gauche. C'est un sous-ensemble de la distribution Gamma où $\beta = 2$ et $\alpha = \nu/2$. Nu est un paramètre de forme.

Constant (constante) : nombre constant ou intervalle constant entre deux arrivées.

Empirical discrete (empirique discrète) : si vous utilisez les distributions Empiriques, vous devez entrer des valeurs et des probabilités dans la table de données. Ce tableau ne sert que pour les distributions empiriques. Vous pouvez importer ses valeurs par le Presse-Papier, grâce aux commandes du menu Edition, ou les saisir directement. Les probabilités peuvent n'être que relatives : ExtendSim les mettra automatiquement à l'échelle. Intervalle décrit l'intervalle de temps entre deux générations. Probabilité donne la probabilité qu'une valeur se trouve dans cet intervalle.

La table utilisera les probabilités discrètes des valeurs de la colonne Intervalle, ce qui signifie que les valeurs listées dans cette colonne sont les nombres exacts que le bloc utilisera pour l'intervalle entre les arrivées.

Empirical stepped (empirique échelonnée) : les données de la table sont considérées comme les probabilités d'une plage de données. Intervalle définit la limite basse de la classe, la valeur suivante définissant la valeur haute de la même classe. Les distributions échelonnées imposent que le dernier point définisse la limite haute de la distribution. Pour cette raison, la probabilité des deux derniers points doit être la même. Si ce n'est pas spontanément le cas, et si l'interpolation est désactivée, les points seront générés automatiquement pour vous.

Empirical interpolated (empirique interpolée) : la probabilité de la distribution sera interpolée entre les points de données.

Erlang : une distribution utilisée dans le trafic téléphonique et la théorie des files d'attente, spécifiée par sa (1) Moyenne. La valeur entière de l'argument (2) k, ($1 \rightarrow \infty$) doit être inférieur à 1. k doit être un entier. Si k vaut 1, la courbe ressemble à une distribution exponentielle, alors que des valeurs plus importantes tendent à donner une distribution normale.

Exponential (exponentielle) : une distribution ayant la forme d'une exponentielle déclinante, spécifiée par sa (1) Moyenne, un réel positif. S'emploie dans le trafic téléphonique et la théorie des files d'attente. La distribution est biaisée vers la gauche, et donc il est plus probable que le temps entre deux arrivées soit compris entre 0 et la moyenne, qu'entre la moyenne et deux fois la moyenne. Notez que la loi de Poisson est l'inverse d'une loi exponentielle. Pour spécifier une cadence d'arrivée Poisson, utilisez une distribution exponentielle pour l'intervalle entre les arrivées. Pour cela, définissez la moyenne de la loi exponentielle comme étant $1/\text{taux Poisson}$.

Extreme Value Type 1A : Extreme Value 1A est une distribution continue non bornée. On l'appelle aussi distribution de Gumbel. Elle décrit la distribution limitante des plus hautes valeurs de plusieurs types d'échantillons. Beta est un paramètre d'échelle.

Extreme Value Type 1B : Extreme Value 1B est une distribution continue non bornée. On l'appelle aussi distribution de Gumbel. Elle décrit la distribution limitante des plus basses valeurs de plusieurs types d'échantillons. Beta est un paramètre d'échelle.

Gamma : une distribution continue limitée par zéro à gauche, et sans limite à droite. Les lois exponentielle et d'ergang sont des cas spéciaux de la distribution gamma. Très souple, la distribution gamma peut s'employer pour des intervalles entre arrivées, des durées de tâches, etc.

Geometric (géométrique) : une distribution discrète limitée par zéro à gauche, et sans limite à droite. Peut être définie comme le nombre d'échecs avant un succès dans une série d'essais. Sa forme est similaire à une distribution exponentielle. Les emplois traditionnels sont : demande dans un stock, nombre d'entités inspectées avant de trouver un défaut.

HyperExponential (HyperExponentielle) : une distribution utilisée dans le trafic téléphonique et la théorie des files d'attente, spécifiée par sa (1) Moyenne. Elle perturbe la distribution Exponentielle à l'opposé de Erlang. Le second argument, s , ($0 < s < 1$), s'échelonne entre 0 et 0.5 avec 0.5 qui donne une distribution Exponentielle.

Hypergeometric (hypergéométrique) : Hypergéométrique est une distribution discrète bornée par $[0, s]$. Elle décrit le nombre de défauts, X , dans un échantillon de taille S dans une population de taille N contenant au total M défauts.

Inverse Gaussian (normale inverse) : Gauss inverse ou Normale inverse est une distribution continue bornée à gauche. Elle vaut zéro au minimum X et est toujours orientée positivement. Aussi appelée distribution de Wald. Alpha est un paramètre de forme. Beta est un mélange de forme et d'échelle.

Inverse Weibull (Weibull inverse) : Weibull inverse est une distribution continue bornée à gauche. Elle vaut zéro au minimum X , et est toujours orientée positivement. En général, cette distribution convient aux données bornées, avec une très longue queue à droite. Alpha est un paramètre de forme. Beta est un mélange de forme et d'échelle.

Johnson SB : distribution continue bornée avec les limites haute et basse comme la distribution Beta. Avec Lognormale et Johnson SU, ces distributions sont des transformations de la distribution Normale et s'emploient pour décrire des ensembles de données unimodaux d'origine naturelle. Lambda est la plage des X au dessus du minimum. Gamma est un paramètre d'asymétrie. Delta est un paramètre de forme.

Johnson SU : The Johnson SU est une distribution continue non bornée. Avec Lognormale et Johnson SB, ces distributions sont des transformations de la distribution Normale et s'emploient pour décrire des ensembles de données unimodaux d'origine naturelle. Lambda est la plage des X au dessus du minimum. Gamma est un paramètre d'asymétrie. Delta est un paramètre de forme.

Laplace : parfois appelée Double exponentielle, c'est une distribution continue non bornée ayant un pic important au centre, localisé en θ . La distribution est à l'échelle de ϕ .

Logarithmic (logarithmique) : Logarithmique est une distribution discrète bornée par $[1, \dots]$. En général, si les données sont bornées par $[0, \dots]$, une traduction des données avant fitting est nécessaire. Theta est associé à la taille de l'échantillon et à la moyenne.

Logistic (logistique) : distribution continue non bornée symétrique autour de la moyenne. Sa forme est très semblable à celle d'une distribution Normale, sauf qu'elle a des queues plus longues. Alpha est un paramètre de shift. Beta est un paramètre d'échelle.

Log-Logistic (log-logistique) : Si $\text{Forme} = 1$, ressemble à une distribution Exponentielle. Si $\text{Forme} < 1$, tend vers l'infini en Position, et décroît lorsque X augmente. Si $\text{Forme} > 1$, vaut zéro en Position, puis fait un pic et décroît.

LogNormal (LogNormale) : logarithme naturel de la variable qui suit la courbe de gauss avec une (1) Moyenne et un (2) Ecart-type donnés. Cette distribution émet une entité environ chaque Moyenne unité de temps, où le temps entre arrivées est toujours positif et est asymétrique de sorte que la plupart des occurrences sont proches du minimum (asymétrie positive). Lognormale est souvent appropriée

pour des process multiplicatifs, tandis que Normale est préférable pour des process additifs. Très employée dans le domaine financier.

Negative Binomial (binomiale négative) : nombre d'échecs avant le Sème succès. P spécifie la probabilité de succès.

Normal (normale) : courbe de gauss avec une (1) Moyenne et un (2) Ecart-type. Cette distribution émet une entité environ chaque Moyenne unité de temps, où le temps entre arrivées a autant de chances d'être au-dessus qu'en dessous de la moyenne. S'emploie lorsque des événements ont des causes naturelles plutôt qu'humaines. La Moyenne est un réel et l'écart-type un réel positif. Plus grand est, plus grande la variation des valeurs autour de la moyenne.

Pareto : Pareto est une distribution continue bornée à gauche. Elle a une valeur finie au minimum X et décroît de manière monotone lorsque X croît. Une variable aléatoire Pareto est l'exponentielle d'une variable aléatoire Exponentielle, et possède de nombreuses caractéristiques semblables. Minimum est le minimum pour X. Alpha est un paramètre d'échelle.

Pearson type V : sert typiquement à représenter le temps pour effectuer une tâche. Une distribution continue bornée par zéro à gauche et non bornée à droite. La densité prend des formes similaires à lognormale, mais peut avoir une plus grand "spike" quand $X = 0$.

Pearson type VI : sert typiquement à représenter le temps pour effectuer une tâche. Distribution continue bornée à gauche.

Poisson : distribution décrivant typiquement le nombre d'événements par unité de temps en fonction d'un taux moyen. Sert rarement à spécifie un intervalle entre arrivée, car l'on utilise à la place une distribution Exponentielle. La moyenne d'une exponentielle équivaut à $1/\text{moyenne}$ d'une distribution de Poisson.

Power Function (Fonction Puissance) : distribution continue avec des limites haute et basse. C'est un cas spécial de la distribution Beta où $q = 1$. La distribution Uniforme est un cas spécial de Fonction Puissance avec $p = 1$. Alpha est un paramètre de forme.

Rayleigh : distribution continue bornée à gauche. S'emploie souvent pour représenter des durées de vie car son taux de hasard augmente de façon linéaire avec le temps, e.g. la durée de vie de tubes à essai. S'applique aussi dans les modélisations de bruits dans les communications. Sigma est un paramètre d'échelle.

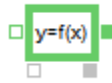
Triangular (triangulaire) : génère une entité toutes les N unités de temps, N étant un nombre réel supérieur ou égal à l'argument choisi pour (1), le minimum, et inférieur ou égal à l'argument choisi pour (2), le maximum, avec la consigne supplémentaire que N tend vers une valeur plus probable. Cette distribution se comportera comme une distribution normale sauf qu'elle pourra être orientée vers la valeur probable et qu'aucune valeur de dépassera les bornes. Notez que la valeur plus probable fait référence au mode, et non à la moyenne.

Uniform (uniforme (entier)) : l'intervalle est un entier supérieur ou égal à l'entier argument 1 et inférieur ou égal à l'entier argument 2. Toutes les valeurs entre le minimum et le maximum ont la même probabilité.

Uniform (uniforme (réel)) : c'est le choix par défaut. L'intervalle est un réel supérieur ou égal à l'entier argument 1 et inférieur ou égal à l'entier argument 2. Toutes les valeurs entre le minimum et le maximum ont la même probabilité.

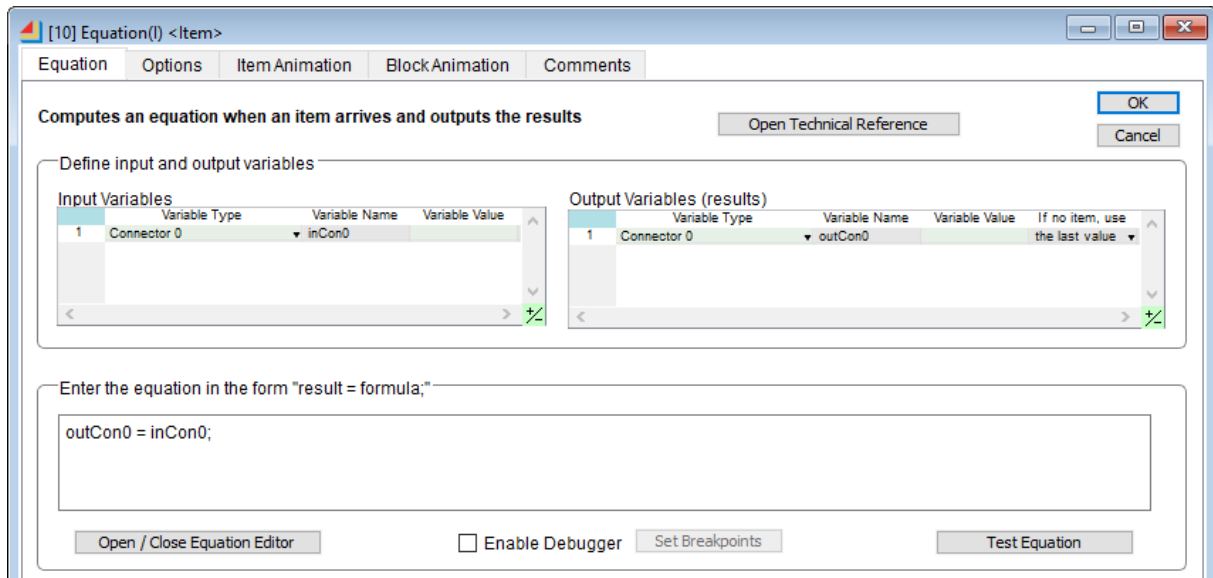
Weibull : distribution utilisée en analyse de fiabilité et basée sur (1) Echelle ou Alpha et (2) Forme ou Beta, des réels positifs. Cette distribution peut adopter les propriétés d'autres distributions (Exponentielle ou Rayleigh), en fonction de ses arguments (1) échelle et (2) forme, qui sont des réels positifs. La courbe de distribution varie considérablement en fonction de la valeur d'Echelle et surtout de Forme. Par exemple, si forme et échelle valent 1, Weibull est quasiment une distribution exponentielle. Avec Echelle 1 et Forme 2, la courbe ressemble à une distribution normale orientée. Dans StatFit, alpha est la forme et beta l'échelle.

Equation(I)



Calcule une équation au passage d'une entité.

Le bloc exécute l'équation chaque fois qu'une entité passe par le bloc. Les résultats de l'équation peuvent modifier les propriétés de l'entité, envoyer de valeurs par les connecteurs, et bien d'autres choses.



Le(s) résultat(s) peuvent en option être affectés à des propriétés de l'entité. Vous pouvez utiliser tous les opérateurs et fonctions ExtendSim, ainsi que tout ou partie des entrées dans la rédaction de l'équation. Il n'y a pas de limite au nombre d'entrées et de sorties : le nombre de connecteurs sur l'icône reflètera la taille de la table (pour les lignes faisant référence à des connecteurs).

Les opérateurs ExtendSim sont : +, -, *, /, ^ (exponentielle), MOD ou % (modulo), AND ou &&, OR ou ||, NOT ou !, == (égal), != ou <> (différent de), <, <=, >, >=

Pour définir plusieurs entrées/sorties, cliquez sur le bouton vert de redimensionnement (+-) dans le coin inférieur droit de la table. Vous pouvez changer le nombre de lignes, copier ou supprimer les lignes sélectionnées.

L'emploi des fonctions ModL peut élargir radicalement ce que vous pouvez effectuer par les équations. Voyez l'aide en ligne ou le manuel du développeur pour connaître ces fonctions.

Chaque entrée doit être nommée dans la table *Input Variables* (Variables d'entrée) pour pouvoir être utilisée dans l'équation. Chaque sortie doit être nommée dans la table *Output Variables* (Variables de sortie).

Vous pouvez utiliser les noms proposés par défaut, ou utiliser les vôtres. ExtendSim vous avertit si des entrées utilisées dans l'équation ne sont pas reliées. Ce n'est pas le cas en sortie : vous pouvez définir des variables en sortie qui ne sont pas utilisées dans l'équation, si cela vous rend service.

Variables en entrée :

En plus des propriétés des entités, les variables en entrée proposées par le système sont les suivantes :

DB read value : lit une valeur à partir d'un emplacement fixe dans une table de BDD. Spécifiez son emplacement dans la colonne "Value".

DB read value using attribute : comme DB read value, mais fait référence à un attribut de l'entité devant comporter une adresse de BDD pour valeur. Lorsque vous faites référence à une variable définie comme DB read value using attribute, vous accédez à la valeur de BDD à laquelle fait référence l'adresse contenue dans l'attribut en question. L'attribut d'emplacement de lecture est spécifié dans la colonne Value.

DB read PRI : lit le 'Parent Record Index' (PRI = Index d'enregistrement parent) à partir d'une cellule dans un champ Enfant. L'emplacement de lecture est fixé et spécifié dans la colonne Value.

DB read PRI using attribute (Parent Record Index) : comme DB read PRI, cette variable reçoit une Index d'enregistrement parent lue dans un champ Enfant. Mais dans ce cas l'emplacement est défini par un attribut Adresse BDD de l'entité. L'attribut d'emplacement de lecture est spécifié dans la colonne Value.

DB address : cette valeur est un pointeur vers une adresse dans la BDD. L'adresse peut pointer sur une base, une table, un champ ou un enregistrement. Elle se compose ainsi de 1 à 4 éléments. Ce type de variable est indispensable car certaines fonctions ModL agissant sur la BDD requièrent une adresse de BDD pour argument.

DB database index - un index unique qui pointe sur une BDD.

DB table index - un index unique qui pointe sur une table.

DB field index - un index unique qui pointe sur un champ.

DB record index - un index unique qui pointe sur un enregistrement.

Static first run init, Static multi run init et Static open model init : les variables statiques sont des variables locales qui conservent leur valeur entre deux calculs. Les variables statiques peuvent être définies comme des variables d'initialisation pour plusieurs simulations ou uniquement pour la première simulation d'une série. **Static first run init** ne prendra sa valeur initiale qu'au début de la première itération de plusieurs simulations, **Static multi run init** au début de chaque itération. **Static open model init** prendra sa valeur initiale à l'ouverture du modèle. Lorsque vous cliquez sur la colonne "**Variable Value**" (Valeur) pour une variable statique, on vous demande de saisir la valeur initiale. Dans la table d'affichage, la valeur à la gauche des " : " est la valeur initiale, celle à la droite la valeur courante.

Connector (connecteur) : la valeur de cette variable est celle du connecteur d'entrée associé.

Variables en sortie :

En plus des propriétés des entités, les types possibles pour les sorties sont les suivants :

DB write value : envoie la valeur de la variable vers une cellule dans la base de données ExtendSim. L'emplacement de la BDD est défini dans la colonne "**Variable Value**" (Valeur).

DB write value using attribute : comme DB write value, cette variable écrit sa valeur à un emplacement dans la BDD. Dans ce cas l'emplacement est variable car défini par un attribut Adresse BDD sur l'entité. Cet attribut est défini dans la colonne "**Variable Name**" (Variable).

DB write PRI (Parent Record Index = index enregistrement parent) : comme DB write value, mais l'emplacement d'écriture doit être un champ enfant, car la valeur écrite est l'index de l'enregistrement parent.

DB write PRI using attribute (Parent Record Index) : comme DB write PRI variable, cette variable écrit sa valeur à un emplacement dans la BDD. Dans ce cas l'emplacement est variable car défini par un attribut Adresse BDD sur l'entité. Cet attribut est défini dans la colonne "**Variable Name**" (Variable).

Connector (connecteur) : associe le nom de la variable à un connecteur de sortie.

L'emploi du bloc Equation(I)est examiné en détail dans le manuel ExtendSim.

Attention : lors de calculs avec des variables entières, les valeurs après la virgule sont perdues.

Suivi des noms dans la BDD (Database Name Tracking)

Notez que les variables de BDD suivantes permettent le suivi des noms : DB read value, DB read PRI, DB write value, DB write PRI, DB address, DB database index, DB table index, DB field index, et DB record index. La fonction de suivi des noms permet que ces variables pointent toujours sur l'emplacement correct dans la base de données, même si celle-ci a subi des modifications dans sa structure après la déclaration de ces variables dans ce bloc. Par exemple, si une variable DB read pointe sur un champ "type voiture", peu importe si le nom ou l'emplacement de ce champ est changé, l'équation saura en faire le suivi.

Onglet Equation

Enter the equation... (entrez l'équation) : Saisissez les instructions. Les variables des tables Variables d'entrée et Variables de sortie doivent être utilisées dans l'équation. Les variables de sortie doivent normalement recevoir une valeur. Toutes les variables d'entrée doivent être utilisées dans l'équation.

Input variables (variables d'entrée) : chaque ligne de la table définit une variable en entrée à utiliser dans l'équation.

Colonne 1 : **Variable Type** (Type) : permet de choisir dans un menu le type de la variable à définir. Voir ci-dessus les différents types.

Colonne 2 : **Variable Name** (Nom Variable) : définit le nom de la variable, à utiliser dans le corps de l'équation.

Colonne 3 : **Variable Value** (Valeur) : variable suivant le type de variable. Pour les variables de connecteur, affiche la dernière valeur présente sur le connecteur. Pour les types DB, affiche la référence de BDD, et permet aussi d'éditer la référence en cliquant sur la cellule. Pour les variables statiques, affiche la valeur d'initialisation (à gauche) et la valeur courante (à droite) de la variable. La valeur d'initialisation peut être éditée en cliquant sur la cellule.

Output Variables (variables de sortie =résultats) : chaque ligne de la table définit une variable en sortie affectée par l'équation.

Colonne 1 : **Variable Type** (Type) : permet de choisir dans un menu le type de la variable à définir. Voir ci-dessus les différents types.

Colonne 2 : **Variable Name** (Nom Variable) : définit le nom de la variable, à utiliser dans le corps de l'équation.

Colonne 3 : **Variable Value** (Valeur) : variable suivant le type de variable. Pour les variables de connecteur, affiche la dernière valeur émise par le connecteur. Pour les types DB Write, affiche la référence de BDD, et permet aussi d'éditer la référence en cliquant sur la cellule.

Colonne 4 : **If no item use** (Si 0 entité...) : permet d'entrer la valeur à utiliser par défaut si aucune entité n'est présente dans le bloc alors que l'équation est calculée.

Open Equation Editor (ouvrir l'éditeur d'équation) : ce bouton ouvre la fenêtre d'éditeur d'équation. Cette fenêtre peut être redimensionnée, et comporte les fonctionnalités de code en couleur, d'indentation automatique ainsi que le remplissage automatique des fonctions.

Enable Debugger (activer Débogueur) : cette case à cocher fait que l'équation est enregistrée sous forme de "code de débogage" pour permettre l'intégration de points d'arrêt. Une équation en format débogage s'exécute plus lentement, et requiert davantage de mémoire. Il est préférable, à la fin du débogage, de décocher cette case.

Set breakpoints (définir points d'arrêt) : ce bouton ouvre la fenêtre "définir des points d'arrêt". Les lignes grises dans la marge de gauche indiquent les seuls endroits où les points d'arrêt peuvent être placés (un point rouge s'affiche alors). Voir le manuel ou l'aide en ligne pour l'utilisation des points d'arrêt.

Test Equation (tester l'équation) : exécute l'équation.

Include files (utiliser des fichiers include) : des fichiers include peuvent être appelés au début de l'équation en cochant cette case. Les fichiers include contiennent du code utilisé dans plusieurs blocs. Les fichiers include simplifient les tâches de programmation qui se répètent dans plusieurs blocs. Ils sont particulièrement utiles quand vous définissez vos propres fonctions pour un bloc et si vous voulez que les autres blocs aient accès à ces fonctions. Par défaut ils sont stockés dans le répertoire Extensions, mais peuvent aussi figurer dans le répertoire du modèle.

Open (ouvrir) : ce bouton ouvre le fichier include mentionné dans cette ligne de la table.

Add Existing (ajouter) : ce bouton ouvre la fenêtre pour sélectionner un fichier include.

Create New (nouveau) : ce bouton crée un nouveau fichier include.

Removed Selected (supprimer sélection) : ce bouton efface de la table les fichiers include sélectionnés.

Prototypes from selected includes (prototypes des includes sélectionnés) : ce menu affiche la liste des prototypes déclarés dans les fichiers Include sélectionnés. Pour qu'un prototype apparaisse dans le menu, il doit avoir été déclaré en haut du fichier Include dans la section prototypes. Lorsque le prototype est affiché dans la zone de texte sous le menu, vous pouvez utiliser un 'glisser-déposer' pour l'inclure dans la fenêtre de l'équation.

Onglet Options

Expand records if necessary (ajouter enregistrement si nécessaire) : si la tentative d'écriture d'une variable 'attribut' va au-delà du nombre d'enregistrements de la table et que cette option est cochée, la table sera agrandie automatiquement. Si l'option n'est pas cochée, cela provoque une erreur.

Convert date time data for DB read and DB write variables (convertir données calendaires pour les variables DB read et DB write) : si une variable DB read pointe sur un champ "Date heure", cocher cette case signifie que la bloc Equation convertira la valeur lue dans le format de l'unité de temps de la simulation avant de calculer l'équation. Pour une variable DB write, la conversion a lieu dans l'autre sens.

Use block seed (base aléatoire) : si la case est cochée et une valeur fournie, cette base aléatoire sera utilisée localement pour les calculs aléatoires faits par le bloc.

Initialize results at start of run to (résultats initialisés à) : spécifie quelle valeur doivent prendre les variables en sortie en début de simulation (avant l'arrivée de la première entité).

Show input connector labels (labels connecteurs d'entrée) : si la case est cochée, tous les connecteurs d'entrée afficheront leur label.

Show input connector values (valeurs connecteurs d'entrée) : si la case est cochée, tous les connecteurs d'entrée afficheront leur valeur.

Show output connector labels (labels connecteurs de sortie) : si la case est cochée, tous les connecteurs de sortie afficheront leur label.

Show output connector values (valeurs connecteurs de sortie) : si la case est cochée, tous les connecteurs de sortie afficheront leur valeur.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Onglet Block Animation

Connecteurs

L'entité entrante est lue puis renvoyée telle quelle en sortie, parfois avec des attributs supplémentaires ou modifiés.

Il y aura un connecteur de sortie continu pour chaque résultat d'équation associé à un connecteur.

Il y aura un connecteur d'entrée continu pour chaque entrée de l'équation associé à un connecteur.

Animation

Un point de couleur peut apparaître sur l'icône lorsque l'équation est sur le point d'être calculée. Pour un calcul en début de simulation, le point est vert; il est bleu pour un calcul toutes les N unités de temps, et rouge dans le cas d'un calcul en fin de simulation. Si aucun point n'apparaît, c'est que le bloc réagit à des messages.

Executive



Ce bloc est le cœur de chaque modèle discret ou flux, et doit être placé à gauche de tout autre bloc du modèle. Il programme les événements et permet de contrôler certains aspects de la simulation des modèles discrets et flux (Rate).

Le dialogue permet de spécifier si la simulation doit s'interrompre selon un critère de temps (end time) ou après qu'un niveau donné a été atteint.

Si vous sélectionnez l'option « **When count connector value >=** » (lorsque la valeur sur Count \geq), et que vous reliez le connecteur, la simulation s'arrêtera lorsque la valeur reçue sera celle indiquée dans le dialogue. Vous pouvez par exemple relier le connecteur Count au connecteur **Num Exited** d'un bloc Exit.

Le champ **Number of item rows allocated** (lignes d'entités allouées) affiche le nombre de lignes de données que le bloc Executive alloue aux entités qui progressent dans le modèle. Lorsque ExtendSim n'a plus de lignes disponibles, il en augmentera le nombre par les paliers indiqué par le champ **Allocate additional items in batches of** (les allouer ensuite par groupes de).

Pour estimer la mémoire occupée par les informations liées aux entités, multipliez le nombre Lignes d'entités allouées par .31. Ce calcul indique la place requise en Koctets. En divisant ce nombre par 1000 vous obtenez le nombre en méga-octets.

Le connecteur "**event**" se met à jour en fonction d'événements systèmes, voyez ses valeurs possibles dans la section Connecteurs. Le connecteur "All events" n'est présent que pour des motifs de

compatibilité. Il est conseillé d'utiliser le connecteur *event* en fonction des événements qui vous intéressent.

L'onglet Item Attributes a deux modes. Le premier mode est sélectionné par l'option **Manage all attributes** (gérer tous les attributs) et permet de renommer et gérer les attributs n'importe où dans le modèle. Vous pouvez voir la liste de tous les blocs du modèle qui utilisent tel attribut.

Dans le second mode, **Declare string attributes values** (déclarer des valeurs d'attribut chaîne), vous gérez le contenu des attributs chaîne du modèle. Ces attributs peuvent être utilisés à divers emplacements du modèle, mais la liste des valeurs possibles de chaque attribut doit être déclarée ici. L'onglet Solver LP est lié à la bibliothèque Rate, et ne doit être modifié qu'à bon escient par des utilisateurs expérimentés.

Onglet Contrôle

Stop simulation (arrêter la simulation) : le choix à **EndTime** (temps de fin) arrête la simulation au temps de fin spécifié dans le dialogue Paramètres. C'est le choix par défaut. SI vous sélectionnez « **When count connector value >=** » (lorsque la valeur sur Count >=) cela fait apparaître le connecteur Count, et la simulation lorsque le nombre reçu sur ce connecteur sera supérieur ou égal à la valeur indiquée.

Initially allocate N items (allouer initialement N entités) : spécifie le nombre initial d'entités allouées en début de simulation. Ce nombre doit être légèrement inférieur au nombre maximum d'entités que vous estimez avoir dans la simulation. Pour déterminer cette valeur, lancez une fois la simulation avec les paramètres par défaut et reprenez la valeur indiquée dans "Lignes d'entités allouées". Cette valeur doit être au moins de 100.

Allocate additional items in batches of (Les allouer ensuite par groupes de ___) : si ExtendSim a utilisé toutes les lignes d'entité allouées et doit générer de nouvelles entités, cette valeur spécifie combien de lignes vont être allouées en complément. La valeur par défaut de 100 est en général appropriée, mais vous pouvez l'augmenter dans de très gros modèles. N'indiquez pas une valeur inférieure à 100 car cela réduirait les performances.

Number of Item rows allocated (Lignes d'entités allouées) : le nombre de ligne que le bloc a alloué aux entités circulant dans le modèle.

Number of Item rows used (nombre de ligne utilisées) : Affiche le nombre maximum d'entité qui ont été dans le modèle à un moment. Sera toujours inférieur ou égal à *Number of Item rows allocated*.

Number of attributes for each item (Nombre d'attributs par entité) : montre combien d'attributs différents l'utilisateur a spécifié dans le modèle.

Report system events on event connector (Indiquer les événements système sur le connecteur Event) : si cochée, montre la valeur du connecteur EventOut sur l'icône du bloc.

Onglet Item Attributes

Select a purpose (Choisit un type d'action) : choisit l'un des deux modes d'action de l'onglet. **Manage all attributes** (gérer tous les attributs) permet de renommer et gérer les attributs n'importe où dans le modèle. **Declare string attributes values** (déclarer des valeurs d'attribut chaîne) gère la liste des valeurs possibles de chaque attribut chaîne.

Show attribute names in Set and Get blocks (Voir noms des attributs au-dessus des blocs) : active l'affichage des noms d'attribut dans les blocs Set et Get du modèle.

Remote blocks track string values (Suivi des valeurs chaîne par les blocs distants) : si cochée, les blocs qui peuvent affecter des attributs chaîne aux entités (ex. Set) utiliseront la valeur chaîne initialement sélectionnée par l'utilisateur pour déterminer quel index entier affecter à l'entité pour valeur d'attribut. Le bloc repérera ainsi une erreur si durant la simulation la valeur chaîne d'origine n'existe plus. Sinon, les blocs utiliseront de manière aveugle la valeur d'index pour affecter des valeurs d'attribut.

Select string attribute (Choisissez un attribut chaîne) : choisissez dans la table la ligne correspondant à l'attribut à éditer.

Colonne 1 **attribute** (attribut) : choisissez dans la table la ligne correspondant à l'attribut à éditer.

Colonne 2 **Type** : cette colonne indique le type des attributs.

Enter string values for the selected attribute table (String Value) (Entrez les valeurs chaîne pour l'attribut sélectionné (table des valeurs chaîne)) : Cette table affiche les valeurs de l'attribut sélectionné.

Colonne 1 (Nom) : noms des blocs utilisant l'attribut sélectionné.

Colonne 2 (Label) : labels des blocs utilisant l'attribut sélectionné.

Colonne 3 (Numéro) : numéros des blocs utilisant l'attribut sélectionné.

Blocks : liste tous les blocs faisant référence à l'attribut sélectionné. Cliquez sur le bouton *Open* pour ouvrir le dialogue du bloc sélectionné.

Delete (Effacer) : efface l'attribut sélectionné. Un attribut ne peut être effacé que s'il n'est référencé dans aucun bloc. Les attributs non référencés sont automatiquement effacés en début de simulation.

Rename (Renommer) : change le nom de l'attribut sélectionné. Cette modification a lieu dans tous les blocs faisant référence à cet attribut. Si l'attribut est mentionné dans une équation (Equation(I)), la modification doit être faite manuellement. Si ExtendSim détecte que l'attribut figure dans une équation, le bloc est ouvert automatiquement.

Onglet Item Contents

L'onglet Contents existe pour les blocs Queue, Queue Equation, Resource Item et Activity, pour permettre de connaître soit le contenu du bloc à un instant donné, soit l'historique de toutes les entités qui ont traversé le bloc. Notez que cette fonction ralentit l'exécution et charge la mémoire : elle est conseillée pour la phase de débogage. Un onglet similaire existe pour le bloc Executive, centralisant tous les onglets Contenu du modèle, et leur activation.

Enable item contents navigation (Activer la visualisation du contenu des entités si coché), active les possibilités de suivi des entités par l'Executive.

Table Filters (Filtre) : ces trois cases à cocher permettent à l'utilisateur de restreindre le nombre de blocs affichant leur contenu dans la table.

Block type (Type de bloc) : active le menu associé qui permet de choisir parmi les blocs Queue, Queue Equation, Resource Item et Activity.

Block label (Label de bloc) : active le menu associé qui permet de choisir parmi les blocs aux labels que vous avez saisis.

H-Block label (Label bloc H) : active le menu associé qui permet de choisir parmi les blocs hiérarchiques aux labels que vous avez saisis.

Open Selected Blocks (Ouvrir blocs sélectionnés) : cliquer sur ce bouton ouvre les dialogues des blocs sélectionnés dans la table. Pour sélectionner un bloc dans le navigateur, cochez la case de la première colonne associée dans la table.

Close Selected Blocks (Ferme blocs sélectionnés) : cliquer sur ce bouton ferme les dialogues des blocs sélectionnés dans la table.

Block navigator (Table) : Chaque ligne de la table représente un bloc du modèle. Le nombre de blocs peut être filtré par les options mentionnées ci-dessus. *Select* (col 1) est une case à cocher permettant de sélectionner des blocs de la table. Un menu dans l'en-tête permet de tout sélectionner/désélectionner, ou d'inverser vos sélections. *Label* (col 2) affiche le label du bloc (ou son nom s'il n'a pas de label) ainsi que son numéro global. *On/Off* (col 3) comporte un menu dans l'en-tête qui permet de d'activer ou de désactiver le suivi du contenu pour les blocs sélectionnés. *Mode* (col 4) permet de choisir le type

de suivi que fera le bloc (contenu actuel, ou historique). Un menu dans l'en-tête permet de choisir l'un ou l'autre mode pour toutes les lignes sélectionnées.

Apply properties to selected blocks (Appliquer les propriétés aux blocs sélectionnés) : cliquer sur ce bouton ajoute les propriétés spécifiées dans la table Propriétés aux blocs sélectionnés dans la table Navigator.

Onglet Discrete Rate

Any rate $\geq N$ is considered infinite (Tout débit $\geq N$ est considéré comme infini) : définit une valeur haute pour les débits importants ; tout nombre au-dessus de cette limite est considéré comme l'infini. Si au moins un débit réel est infini, la simulation s'arrête et donne un message d'erreur.

Any effective rate $\leq N$ is considered to be zero (Tout débit réel $\leq N$ est considéré comme zéro) : définit une valeur basse pour les débits faibles ; tout nombre en-dessous de cette limite est considéré comme zéro. Si au moins un débit réel est à zéro, la simulation s'arrête et donne un message d'erreur.

Blocks update flow status (Les blocs mettent à jour le statut du flux) : **Only when necessary** (uniquement lorsque nécessaire) est l'option par défaut. C'est le plus efficace pour les temps de calcul, car les quantités de flux sont mises à jour uniquement lorsque nécessaire. Si **each block defines how often** (chaque bloc définit sa fréquence) est choisie, le dialogue de chaque bloc propose une option spécifique pour contrôler ses mises à jour. L'option **each block at each step** (chaque bloc à chaque événement) doit être utilisée avec précautions car elle demande beaucoup de temps de calcul.

Valve animates and reports blocking and starving information : (La Vanne indique et anime les états de blocage et pénurie) : ne concerne que le bloc Valve. Si cochée, chaque Vanne fournit pour l'animation davantage d'information sur son statut.

Merge or Diverge blocks (Distributional, Priority, or Sensing mode only) : Mélange et Divergence (modes Distribution, Priorité ou Réactif uniquement) : Ordre de préférence : **defined by Simulation Order** (Défini par la simulation) laisse se faire un calcul interne. **Each block defines its own** (chaque bloc définit son classement) rend possible de modifier l'ordre en sélectionnant des blocs dans la table et en utilisant les boutons << et >>.

Change bias order (Change l'ordre) : en cliquant sur les boutons << ou >> vous changez l'ordre des lignes sélectionnées dans la table.

Colonne 1 **Bias** (Préférence) : montre les valeurs de biais des blocs associés à cette ligne de la table.

Colonne 2 **Label[#]** : montre le label des blocs associés à cette valeur de biais.

Colonne 3 **Name** (Nom) : montre le nom des blocs associés à cette valeur de biais.

Colonne 4 **Mode** : montre le mode des blocs associés à cette ligne de la table.

Colonne 5 **Sim Order** (Ordre simu) : montre l'ordre de la simulation des blocs associés à cette ligne de la table.

Merge blocks (Proportional mode) (Blocs Mélange (mode Proportionnel avec boucle vide)) : trois comportements peuvent être définis pour les blocs Mélange : **Blocks need product simultaneously to push flow** (Toutes les branches entrantes doivent passer le flux), **Blocks push flow even in empty loops** (Le flux passe même avec des boucles vides) et **Each block defines how it will push flow** (Chaque bloc définit comment il pousse le flux). Reportez-vous au Guide de l'utilisateur pour une explication sur ces comportements.

Table **Manage flow units** (Gère les unités de flux)

Colonne 1 (unité de flux) : liste toutes les unités de flux définies pour le modèle.

Add Unit (Ajouter unité) : un dialogue permet de définir une nouvelle unité.

Delete Unit (Effacer unité) : efface l'unité de la ligne sélectionnée.

Rename Unit (Renommer unité) : un dialogue permet de renommer l'unité de la ligne sélectionnée.

Refresh Table (Rafraîchir) : rafraîchit le contenu de la table.

Delete All Units (Tout effacer) : efface toutes les unités de la table.

Onglet Flow Attributes

Select a purpose (Choisit un type d'action) : choisit l'un des deux modes d'action de l'onglet. **Manage all flow attributes** (Gérer tous les attributs flux) permet de renommer et gérer les attributs n'importe où dans le modèle. **Declare string layer attribute values** (Déclarer des valeurs d'attribut couche chaîne) gère la liste des valeurs possibles de chaque attribut flux chaîne.

Remote blocks track string values (Suivi des valeurs chaîne par les blocs distants) : si cochée, les blocs qui peuvent affecter des attributs chaîne aux entités (ex. Set) utiliseront la valeur chaîne initialement sélectionnée par l'utilisateur pour déterminer quel index entier affecter à l'entité pour valeur d'attribut. Le bloc repérera ainsi une erreur si durant la simulation la valeur chaîne d'origine n'existe plus. Sinon, les blocs utiliseront de manière aveugle la valeur d'index pour affecter des valeurs d'attribut.

Select string layer attribute (Choisissez un attribut couche chaîne) : choisissez dans la table la ligne correspondant à l'attribut à éditer.

Colonne 1 (attribut flux) : choisissez dans la table la ligne correspondant à l'attribut à éditer.

Colonne 2 (Type) : cette colonne indique le type des attributs.

Colonne 3 (Initialisation) : valeur initiale pour l'attribut sélectionné.

Enter string values for the selected attribute (Entrez les valeurs chaîne pour l'attribut sélectionné (table des valeurs chaîne)) : Cette table affiche les valeurs de l'attribut sélectionné.

Colonne 1 (Nom) : noms des blocs utilisant l'attribut sélectionné.

Colonne 2 (Label) : labels des blocs utilisant l'attribut sélectionné.

Colonne 3 (Numéro) : numéros des blocs utilisant l'attribut sélectionné.

Blocks : liste tous les blocs faisant référence à l'attribut sélectionné. Cliquez sur le bouton Ouvrir pour ouvrir le dialogue du bloc sélectionné.

Delete (Effacer) : efface l'attribut sélectionné. Un attribut ne peut être effacé que s'il n'est référencé dans aucun bloc. Les attributs non référencés sont automatiquement effacés en début de simulation.

Rename (Renommer) : change le nom de l'attribut sélectionné. Cette modification a lieu dans tous les blocs faisant référence à cet attribut. Si l'attribut est mentionné dans une équation, la modification doit être faite manuellement. Si ExtendSim détecte que l'attribut figure dans une équation, le bloc est ouvert automatiquement.

Change Initial Value (Modifier valeur initiale) : permet d'affecter / modifier la valeur initiale de l'attribut.

Onglet LP Solver

lp_solver version (version de lp_solver) : indique la version de la DLL LP solver utilisée par la bibliothèque Rate.

LP debugging options (Options de débogage) : contactez-nous si vous pensez avoir besoin d'utiliser ces options.

Connecteurs

Count : en reliant ce connecteur à un bloc du modèle et en sélectionnant **Stop simulation when count connector value >=** (Arrêter la simulation lorsque la valeur sur Count >=) la simulation stoppera lorsque le nombre reçu sur ce connecteur sera supérieur ou égal à la valeur indiquée.

event : envoie un message lors d'événements spécifiques de la simulation.

all events : La valeur sur le connecteur identifie l'événement. Les plus importants événements ont leur propre connecteur (le connecteur *all events* reste pour des raisons de compatibilité).

- 1 - Événement de simulation. Est envoyé avant que n'importe quel autre bloc reçoive ses événements.
- 2 - Événement de simulation. Est envoyé après que tous les autres blocs ont traité leur événement
- 3 - Allocation de nouvelles lignes d'entités. SysGlobalInt2 mémorise le nombre de lignes actuellement allouées.
- 4 - Suppression d'une entité. SysGlobalInt7 est l'indice de l'entité détruite.
- 5 - Après traitement du messageur CheckData par le bloc Executive, mais avant que tout autre bloc ait traité ce messageur CheckData
- 6 - Après traitement du messageur InitSim par le bloc Executive, mais avant que tout autre bloc ait traité ce messageur InitSim
- 7 - A l'ouverture du modèle

Before event (Avant l'événement) : envoyé avant que n'importe quel autre bloc reçoive son événement. Vaut 1 lorsque l'événement intervient.

After event (Après l'événement) : envoyé après que tous les autres blocs ont traité leur événement. Vaut 1 lorsque l'événement intervient.

New item rows allocated (Nouvelle ligne d'entité allouée) : envoyé lorsque de nouvelles lignes sont allouées pour des entités. Vaut le nombre de lignes actuellement allouées.

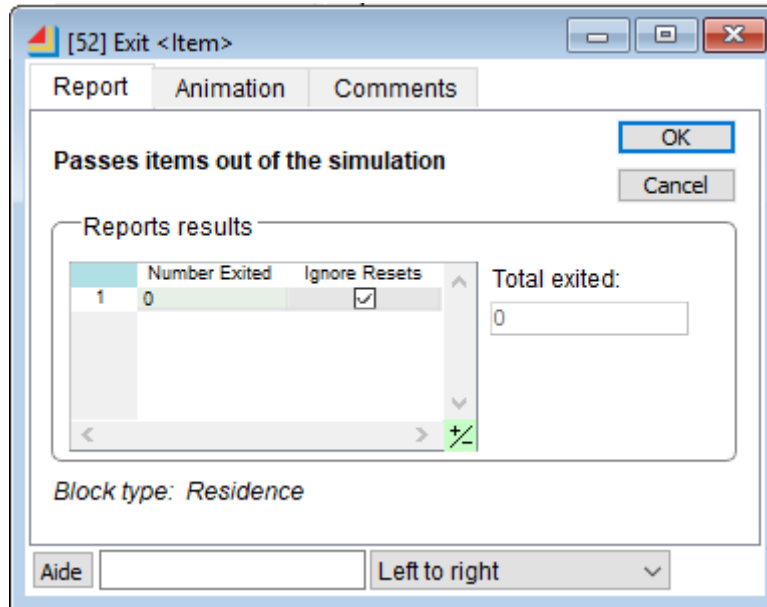
Entité supprimée : envoyé quand une entité est supprimée. Vaut l'index de cette entité.

La case **Report system events on event connector** (Indiquer les événements système sur le connecteur Event) doit être cochée pour que les événements soient visibles.

Exit



Fait sortir des entités de la simulation.



Le nombre total d'entités reçues par le bloc est affiché dans le dialogue et transmis au connecteur *num exited*. Le nombre d'entités reçues par chaque entrée est affiché dans le dialogue et transmis aux connecteurs de sortie continus situés à droite du bloc.

Il est important de faire sortir les entités lorsque plus rien ne doit leur arriver. Les stocker dans une Queue ou un Resource Item ne libérerait pas la mémoire allouée aux entités, ce qui à terme peut devenir un problème.

Onglet Results

Colonne 1 Number exited (Nombre sorties) : nombre d'entités ayant quitté la simulation par le connecteur correspondant.

Colonne 2 Ignore resets (Ignorer reset) : lorsque cette option est cochée, le compteur n'est pas réinitialisé à 0 lorsque le bloc Clear Statistics (bib. Value) demande un effacement.

Total exited (Total sorties) : nombre total d'entités ayant quitté la simulation par ce bloc.

Onglet Animation

Animate exit values (Animer les valeurs de sortie) : active/désactive l'animation du comptage des entités sur les connecteurs de sortie.

Animate H-block object (Animer objet de bloc H) : Indique sur un objet d'animation du bloc H supérieur le nombre d'entités sorties. Indiquer le n° de l'objet d'animation du bloc H parent.

Connecteurs

En entrée, les entités à retirer de la simulation.

Le connecteur de sortie du haut indique le nombre total d'entités absorbées par toutes les entrées. Ceux sur la droite de l'icône indiquent le nombre d'entités reçues par chaque connecteur d'entrée.

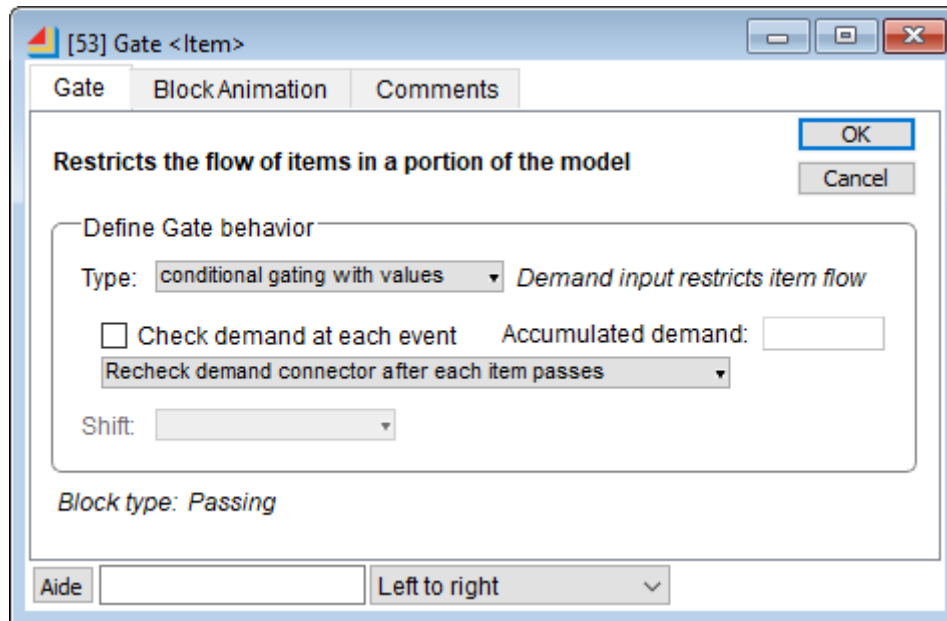
Animation

Le champ texte de l'icône indique le nombre total d'entités absorbées. Chaque entrée est activée lorsqu'une entité la traverse.

Gate



Contrôle l'accès des entités à une portion donnée du modèle.



Fonctionne comme une barrière ouverte ou fermée, ou comme un sas.

Le bloc a quatre modes de fonctionnement, choisis par le menu **Type** : **Conditional gating with items** (barrière pilotée par entités), **conditional gating with values** (barrière pilotée par valeurs), **area gating** (sas sur une portion), **schedule gating with shift** (barrière pilotée par bloc Shift).

En mode **Conditional gating with items**, le bloc accumule la valeur des entités envoyées sur le connecteur *demand* (connecteur discret) et laisse passer le nombre d'entité correspondant à la valeur (quantité) des entités. Par exemple, si une entité de valeur 4 arrive sur le connecteur *demand*, le bloc laisse passer quatre entités avant de refermer sa "porte". Le bloc accumule la demande : la présence de deux entités de valeur respective 4 et 3 sur le connecteur *demand* créera une demande cumulée pour 7 entités sur le connecteur d'entrée. Si les entités désirées ne sont pas immédiatement disponibles, le bloc conserve la quantité demandée et la décrémente lorsque les entités deviennent disponibles et sont tirées.

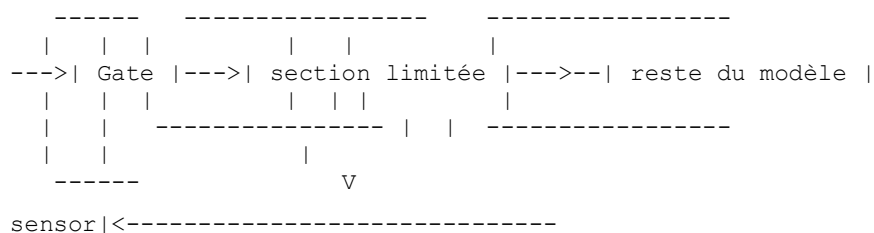
En mode **conditional gating with values**, le bloc laisse passer les entités tant que la valeur transmise à *demande* (connecteur continu) est supérieure à 0.5. Par exemple, une valeur de 4 maintient le bloc en état "passant". Le 4 n'influe pas sur le nombre d'entités traitées : il sert simplement (parce qu'il est >0.5) à maintenir le bloc ouvert. Si le signal reçu par *demande* descend en-dessous de 0.5, le bloc se ferme.

Avec **area gating**, le connecteur *demand* est remplacé par un connecteur *témoin*. Dans ce mode le bloc n'autorise qu'un nombre limité d'entités à accéder simultanément à une portion donnée du modèle. Spécifiez dans le dialogue le nombre d'entités autorisées simultanément dans la zone du modèle. Les premières entités qui arrivent passent sans encombre, tant que le nombre spécifié n'est pas atteint. Les entités suivantes sont alors bloquées, tant qu'une des premières entités passées n'a pas quitté la zone délimitée.

En mode **schedule gating with shift**, l'horaire permet qu'un nombre illimité d'entités passent en horaire actif et aucune en horaire inactif. Note : avec des horaires numériques, la porte sera toujours ouverte si la valeur est > 0. Dans ce mode, il ne faut utiliser ni le connecteur *Sensor* ni le connecteur *demand*.

Le bloc sait qu'une entité a quitté la zone à accès limité lorsque le connecteur *Sensor* reçoit un signal de la sortie du dernier bloc de la zone surveillée. (Le connecteur *Sensor* n'absorbe pas les entités). Ainsi le dernier bloc de la zone doit avoir deux sorties connectées en parallèle : une en direction de *Sensor*, l'autre vers le bloc suivant. Il peut y avoir plusieurs connecteurs *Sensor* si les entités quittent la zone de sas par plus d'une voie.

La façon dont le bloc est connecté détermine les limites de la section observée, selon le schéma ci-dessous. La porte est le premier bloc de la section gérée. Le dernier bloc de la zone observée adresse à la fois le bloc suivant et le connecteur *Sensor* du bloc *Gate*.



Le bloc peut représenter une attente conditionnelle, par exemple : plusieurs commandes dans un restaurant, clients faisant la queue, avions à l'atterrissage.

Attention : avec le bloc *Gate*, il est possible de bloquer les événements de façon prématurée au cas où un bloc de ressource (par exemple, *Resource Item* de la bib. *Item*) est relié au *Gate* et que le *Gate* est contrôlé par un bloc *Lookup Table*. Si le premier nombre émis par la table de correspondance est un zéro, et qu'il n'y a pas d'autre source d'événements dans le modèle, le bloc *Executive* va aller directement en fin de simulation.

Onglet *Gate*

Accumulated demand (Demande accumulée) : Indique les demandes non satisfaites faites par le connecteur *demand*. Si émet 0, le bloc ne laisse pas passer les entités.

Check demand connector at each event : normalement, le connecteur *demand* est 'consulté' lorsqu'une entité tente de traverser le bloc, ou si un message est envoyé par un bloc lié au connecteur *demand* (ce qui a lieu en général lorsque la valeur du connecteur change). C'est le comportement efficace et correct dans la plupart des cas. Mais il peut être nécessaire de contrôler le connecteur à chaque événement, notamment lorsque l'on fait appel à une condition sur le temps (par exemple l'équation : $if(currentTime \geq 10) result = TRUE$). Aucun événement n'est associé à cette évaluation. En vérifiant à chaque événement, la condition est évaluée plus souvent (pas forcément au temps 10 cependant, mais à l'événement juste après assurément). Une autre solution étant de lier un bloc discret ou non qui publie des événements.

Items allowed in area (Entités autorisées) : nombre maxi d'entités qui peuvent se trouver simultanément dans la zone délimitée par le sas.

Items currently in area (Entités dans la zone) : nombre d'entités actuellement dans la zone.

Recheck demand connector after each item passes (Vérifier Demande après le passage de chaque entité) : contrôle le comportement du bloc vis à vis de plusieurs entités cherchant à passer au même événement. En mode **Recheck demand connector after each item passes**, le bloc interroge le connecteur *demand* chaque fois qu'une entité passe. De nouveaux messages sont générés, mais cela permet de prendre la décision entité après entité. En mode **When Gate is open, pass as many items as possible** (Quand barrière ouverte, passer autant d'entités que possible), le bloc fait passer instantanément autant d'entités que possible. S'emploie essentiellement lorsque demande est un connecteur continu dont la valeur à un moment ouvre la barrière.

Use item quantity for area count (Tenir compte de la quantité de l'entité) : si la case est cochée, la 'quantité' d'une entité est utilisée dans le calcul du nombre autorisé dans le sas, en entrée comme en sortie.

Use Shift (Utiliser l'horaire) : sélectionnez l'horaire applicable à ce bloc. L'horaire doit être ON/OFF. Un horaire peut être utilisé pour contrôler le bloc Gate de plusieurs façons. En mode "**schedule gating with shift**", l'horaire sélectionné sera seul responsable de l'ouverture et de la fermeture de la porte. Mais l'horaire peut aussi servir dans les autres modes. Ainsi la valeur du connecteur *demand* est ignorée lorsque l'horaire de référence passe à off, et le comportement du bloc Gate reprend normalement lorsque l'horaire devient On. Le connecteur *sensor* peut similairement être sensible à un horaire.

Connecteurs

En entrée, les entités à filtrer.

En sortie, les entités admises dans la zone de sas.

AD : demande accumulée. Indique la demande accumulée à faire passer.

: le nombre d'entités dans la zone surveillée.

demand : tant que le signal reçu est supérieur à 0,5, ou lorsqu'une entité arrive sur *demande*, le connecteur d'entrée tire une entité. Permet au bloc d'accumuler une demande pour plusieurs entités lorsqu'un connecteur discret est relié à *demande*. Le connecteur est discret ou continu en fonction du choix **Type** dans l'onglet Gate.

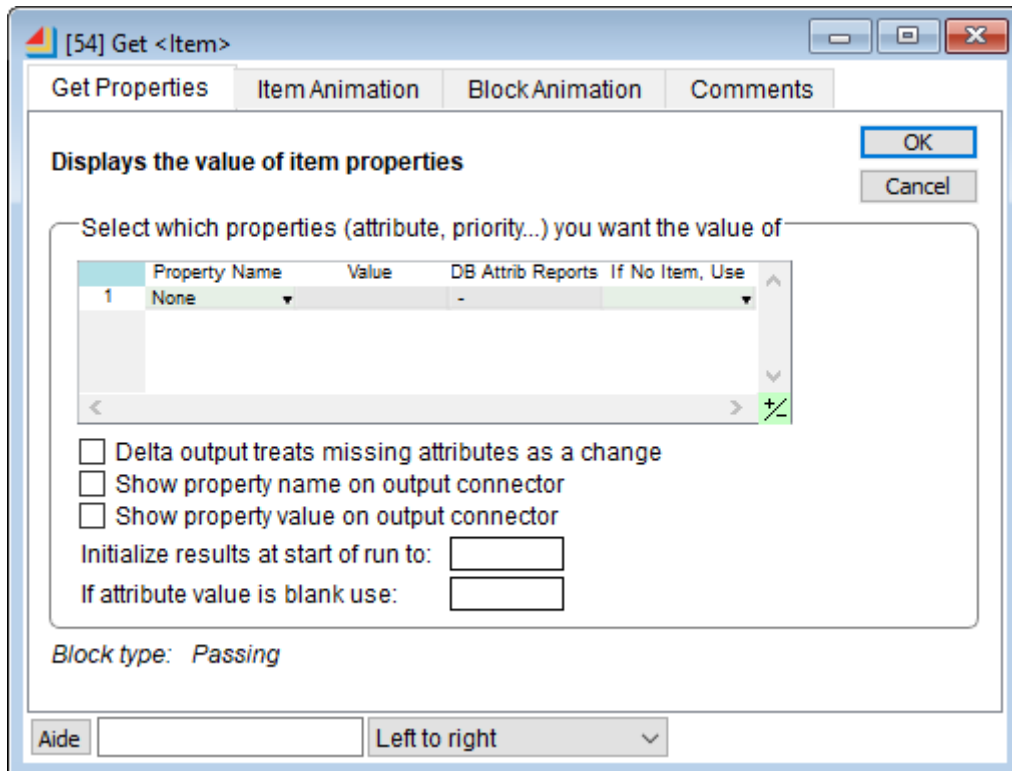
Animation

La barrière (barre rouge) s'ouvre et se ferme en fonction de l'état du bloc.

Get



Affiche les propriétés des entités avant de les faire passer. La valeur des propriétés s'affiche dans le dialogue et sur les connecteurs de sortie.



Plusieurs propriétés peuvent être lues, ce qui augmentera le nombre de connecteurs de sortie exploitables dans le modèle.

Les propriétés à lire peuvent être des attributs, la priorité, et la quantité (valeur) de l'entité. Les attributs peuvent être des attributs chaîne ou numériques, et des attributs liés à la base de données (DB Address). Si vous lisez un attribut chaîne, la valeur émise sur le connecteur sera celle de l'index de la valeur chaîne (par ex. si l'attribut contient ROUGE et BLEU et que vous affichez la valeur BLEU, le connecteur émettra un 2.)

Si vous lisez un attribut DB Address, vous devrez sélectionner dans la table ce qui sera lu dans la base de données (voir plus bas.)

Quand aucune entité n'est présente dans le bloc, la valeur émise sera celle spécifiée dans la colonne 4 : la dernière valeur, novalue ou un nombre spécifique.

Le connecteur Delta émet la valeur 1 si la valeur de l'attribut a changé depuis la précédente lecture (sinon zéro).

Onglet Get Properties

Table

Colonne 1 Property Name (Propriété) : spécifie la propriété à lire sur l'entité. Le menu permet de choisir parmi des propriétés existantes.

Colonne 2 Value (Valeur) : affiche la valeur de la propriété spécifiée pour la dernière entité passée par le bloc.

Colonne 3 DB attrib reports (Attrib BDD =) : cette colonne spécifie ce qui est lu lorsque l'attribut spécifié est une adresse de base de données (DB Address). Chaque cellule propose les options suivantes : **db index**, **table index**, **field index**, **record index**, et **db address**. Les quatre premières options retourneront la valeur d'index sélectionnée pour la valeur d'adresse de BDD donnée. La dernière option la valeur d'adresse de BDD elle-même (cette valeur n'est pas un nombre utilisable directement, et ne servira que dans certains contextes, par exemple pour transmettre la valeur à un bloc Equation faisant appel à une fonction ayant besoin de cette adresse.)

Colonne 4 If missing use (Si absente, utiliser) : spécifie quelle valeur émettre lorsqu'aucune entité n'est présente. Trois options sont proposées : **last value (la dernière valeur)** affichera la dernière valeur correcte trouvée. **Novalue** retournera noValue. La dernière option permet d'indiquer un nombre à retourner.

Delta output treats missing attributes as a change ("delta" traite l'absence d'attribut en changement) : le connecteur delta (D) est modifié si l'attribut n'est pas présent sur une entité. L'absence de l'attribut équivaut à une différence d'attribut.

Show property name on output connector (Nom sur le connecteur de sortie) : affiche le nom de la propriété près du connecteur de sortie.

Show property value on output connector (Valeur sur le connecteur de sortie) : affiche la valeur de la propriété près du connecteur de sortie.

Initialize results at start of run to (En début de simulation, valeur initialisée à) : permet une initialisation à une valeur particulière.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Connecteurs

L'entité entrante est lue puis renvoyée telle quelle en sortie.

Connecteur(s) de sortie continus : valeur de la propriété, ou bien noValue, ou bien valeur par défaut spécifiée.

Delta : émet la valeur 1 si la valeur de l'attribut a changé depuis la précédente lecture (sinon zéro). Le connecteur delta est à côté du petit triangle rouge sur l'icône.

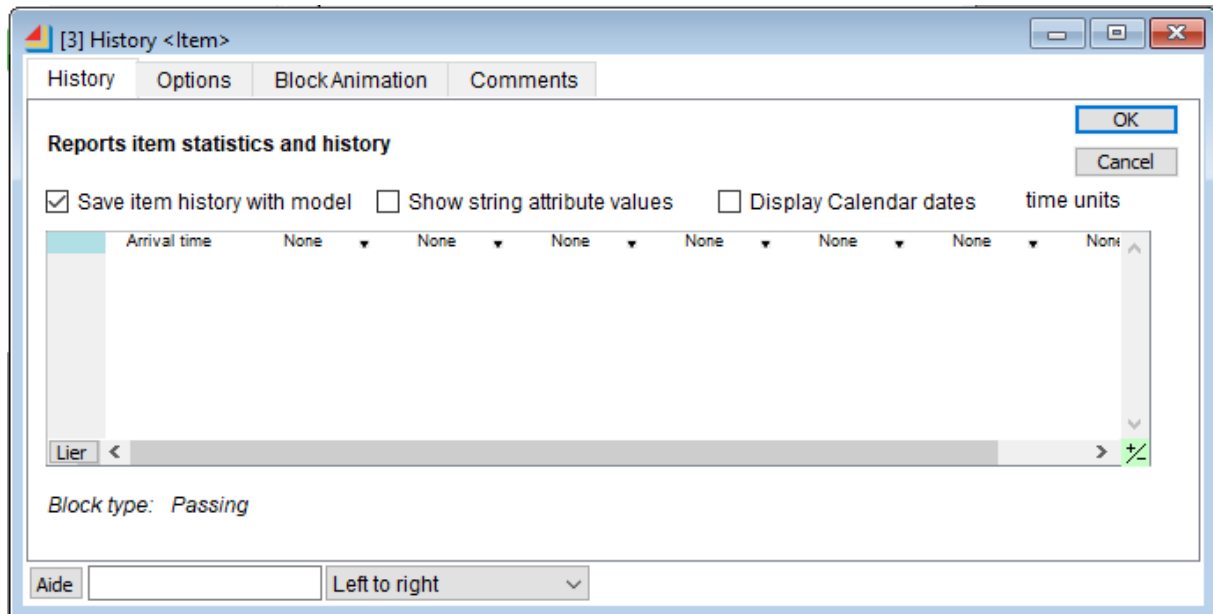
Animation

L'arrivée des entités anime l'icône du bloc.

History



Recherche et affiche des informations sur les entités qui passent.



La première colonne indique l'heure de passage, les autres colonnes étant des valeurs de propriétés de l'entité. Vous spécifiez les propriétés à observer dans un menu déroulant dans chaque colonne.

Le bloc peut être placé de deux manières dans un modèle. Par défaut il a deux connecteurs discrets standard, et s'insère en série dans le flux. Il peut aussi être ajouté au modèle en cliquant du bouton droit sur un connecteur de sortie d'un bloc discret : un menu propose alors d'ajouter le bloc en mode "observation". Dans ce mode le bloc n'a qu'un connecteur d'entrée, et se relie en parallèle du flux. Il ne fera passer aucune entité, mais les observera seulement. Les informations observées sont les mêmes dans les deux modes.

Onglet History

Save item history with model (Enregistrer avec le modèle) : si cochée, le contenu de la table sera enregistré avec le modèle. Sinon, fermer le modèle effacera la table.

Show string attribute values (Voir les attributs chaîne) : si cochée, la table affichera les valeurs chaîne associées aux attributs chaîne.

Display Calendar dates (Dates calendaires) : si cochée, les heures d'arrivée seront affichées au format calendaire.

Colonne 1 Arrival time (Heure arrivée) : affiche l'heure d'arrivée de chaque entité dans le bloc.

Colonne 2-8 Property (Propriété) : vous sélectionnez la propriété à afficher en cliquant sur le menu déroulant.

Onglet Options

Store history in database at end of run (Stocker l'historique dans la BDD en fin de simulation) : si cochée, les données rassemblées par le bloc History sont conservées dans une table de BDD en fin de simulation.

Data collection window (Voir contenu de la table) : affiche la table où est stocké l'historique en mode Edition.

Control log time (Plage de recueil) : le cadre temporel de recueil des données. Si cochée, le bloc ne recueillera d'information que durant la période indiquée.

Pause when item passes (Pause à chaque passage) : si cochée, le modèle fait une pause chaque fois qu'une entité entre dans le bloc.

Disable recording (memory saving) (Désactiver le bloc (économie de mémoire)) : si cochée, le bloc n'enregistrera plus le passage des entités. Cela permet d'économiser de la mémoire si les informations ne sont plus utiles pour la simulation, sans forcément supprimer le bloc. Une croix orange sur l'icône du bloc indique que le bloc est inactif.

Onglet Block Animation

Show item information when animation is on (Voir information entité si animation active) : anime l'information sur l'entité sur l'icône si l'animation est activée.

Connecteurs

Le connecteur d'entrée discret au milieu est celui qui fait entrer les entités. Celui du haut est un connecteur témoin, présent quand le bloc est en mode observation.

L'entité entrée sort telle quelle. Le connecteur de sortie est présent en standard. En mode observation, le bloc possède uniquement le connecteur témoin à gauche de l'icône.

Animation

L'icône montre si une entité est présente, et en option les valeurs des attributs de la dernière entité observée.

Information



Fournit des statistiques sur les entités qui le traversent.

Le bloc information donne des informations sur les entités à un certain point du modèle. En plus de valeurs sur l'intervalle entre deux entités et le nombre d'entités observées, le bloc sert à calculer des temps de cycle pour parcourir une certaine portion du modèle.

Vous pouvez déterminer le temps de cycle en utilisant l'option Attribut de timing du bloc Create (onglet Options). Placez un bloc Information plus loin dans le modèle et sélectionnez le même attribut de timing.

L'attribut porte initialement l'heure de création de l'entité. Lorsqu'elle arrive au bloc Information, la valeur de cet attribut est soustraite du temps courant. La différence est le temps de cycle. Il n'est pas obligé de calculer ce temps depuis un bloc Create : vous pouvez utiliser un bloc Set ou Equation(I) n'importe où dans le modèle pour affecter l'attribut de timing.

Onglet Statistics

Item Count Comptage

Number of items (Nombre d'entités) : le nombre total d'entités passées par le bloc.

Ignore item quantity (Ignorer la valeur des entités) : si cochée, une entité de valeur (quantité) supérieure à un comptera pour un uniquement. Sinon, le comptage sera augmenté de cette quantité.

Add N to the # (CountOut) output (Ajouter N au connecteur #) : cette constante sera ajoutée au connecteur de sortie countOut.

Reset item count every N items (Remettre à zéro toutes les N entités) : permet de faire un reset toutes les N entités

Throughput rate (Débit) : affiche le débit du bloc, en calculant le nombre moyen d'entités passées par unité de temps.

Calculate TBI and Cycle Time statistics (Calculs d'intervalle et temps de cycle) : si cochée, le bloc calcule aussi les statistiques d'intervalle et temps de cycle.

TBI (Intervalle) : calcule les valeurs courante, moyenne, mini et maxi.

Cycle Time (Temps de cycle) : calcule les valeurs courante, moyenne, mini et maxi.

Timing attribute (Attribut de timing) : permet de spécifier quel attribut sert à calculer le temps de cycle. Il est appelé attribut de Timing, et contient l'heure prise au début du cycle.

Time units (Unité de tps) : spécifie l'unité de temps utilisée pour tous ces calculs.

Onglet Options

Do not reset when statistics are cleared (Insensible au Reset des statistiques) : si cochée, le bloc ne remet pas à zéro les statistiques lorsqu'une commande de reset affecte les blocs information.

Reset when (Reset quand) : permet de choisir quand le bloc va contrôler la possibilité d'un reset. Les options sont : **'r' (reset) input receives a value > 0** ('r' (reset) reçoit une valeur > 0) ; **before current item is counted** (avant de compter l'entité courante) ; **after current item is counted** (après avoir compté l'entité courante). Notez que ces options ont effet quand le connecteur reset va vérifier s'il est à TRUE, et pas forcément lorsque le bloc est remis à zéro.

L (Length) output connector sends (Connecteur L (Longueur) émet) : **TRUE**, ou **TRUE puis FALSE** la première option donne à L une valeur TRUE, et envoie un message juste après ; la seconde option envoie une valeur TRUE et remet juste après le connecteur à FALSE. Cela peut être nécessaire si le connecteur relié requiert une valeur TRUE, mais un retour à FALSE entre deux messages TRUE.

Report xxx time between items at 'i' output : (Indique sur 'i' l'intervalle xxx entre 2 entités) : le menu permet de choisir si le connecteur i émet l'intervalle moyen ou l'intervalle courant.

Report xxx cycle time at 'ct' output (Indique sur 'ct' le temps de cycle xxx) : le menu permet de choisir si le connecteur ct émet le temps de cycle moyen ou le temps de cycle courant.

Onglet Block Animation

Show item information when animation is on (Voir information entité si animation active) : anime l'information sur l'entité sur l'icône si l'animation est activée.

Connecteurs

L'entité entrée sort telle quelle.

r (reset) : remet à zéro les statistiques du bloc lorsque reçoit une valeur **TRUE**.

Connecteurs de sortie continus : (dans l'ordre d'apparition.)

L (longueur) : indique le nombre d'entités dans le bloc. C'est soit zéro, soit un.

(nombre) : indique le nombre d'entités passées par le bloc.

I : intervalle de temps écoulé entre l'arrivée de l'entité actuelle et de la précédente.

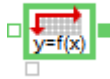
ct : indique la valeur de temps de cycle.

TP : indique le débit.

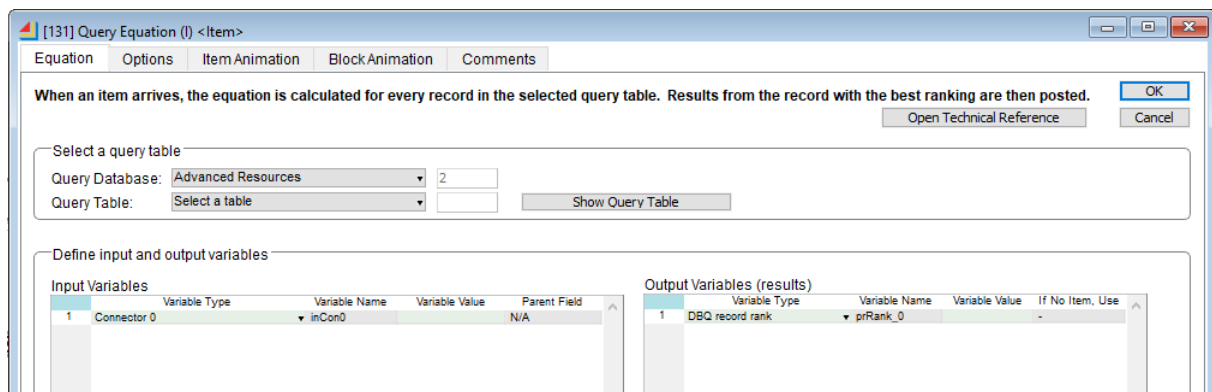
Animation

L'icône montre si une entité est présente, et le nombre d'entités passées par le bloc.

Query Equation (I)



Le bloc Query Equation (I) est utilisé pour classer les enregistrements dans une table de BDD ExtendSim et sélectionner intelligemment un enregistrement, en fonction de règles de classement. Une équation définie par l'utilisateur dans le dialogue du bloc est calculée une fois pour chaque enregistrement dans la table ; les résultats servent à assigner un rang à chaque enregistrement. L'enregistrement ayant le plus haut rang est celui qui est sélectionné.



Le bloc Query Equation (I) s'emploie quand une BDD contient des informations requises pour prendre des décisions dans un modèle. L'évaluation et le classement pourraient être accomplis en utilisant les blocs Equation simples, mais les équations seraient longues et compliquées. En outre, le bloc Query Equation (I) fournit des informations spécifiques permettant d'écrire des requêtes plus complexes.

Principes de fonctionnement

Un cycle de requête est le point temporel où un bloc Query Equation (I) exécute son équation pour choisir le prochain enregistrement " gagnant " (l'onglet Options propose des choix pour contrôler l'initiation du cycle de requête).

Il est important de savoir que seuls les résultats des équations pour l'enregistrement gagnant sont utilisés pour un cycle de requête particulier. Par exemple, si la table de requête contient 10 enregistrements, l'équation sera calculée 10 fois et 10 ensembles individuels de résultats d'équation seront collectés. Cependant, seuls les résultats de l'enregistrement le mieux classé seront utilisés et les résultats des 9 autres sont oubliés.

Les types de variables d'entrée et de sortie des blocs sont spécialement conçus pour aider les modélisateurs à écrire des équations qui vont classer les enregistrements correctement et sélectionner l'enregistrement gagnant. L'équation définie par l'utilisateur est calculée une fois pour chaque enregistrement afin que chacun obtienne un classement.

Entités mémo (spawn)

Une fonction particulière au bloc Query Equation (I) est la possibilité de créer de nouvelles entités. Ces entités appelées « spawn » (on les appelle 'mémos') ont généralement un rôle spécial dans le modèle, étant indépendantes et parallèles aux entités qui circulent, et que l'on appelle 'entités de passage' car elles ne font que passer par le bloc Query Equation (I). Les entités mémo créées en fonction de la logique de l'équation et de ce qui a été trouvé dans la base de données. Si une entité mémo est créée pour un enregistrement, les données de l'enregistrement peuvent être transférées à l'entité mémo sous forme d'attributs.

Généralités sur les blocs Equation

Les opérateurs ExtendSim sont : +, -, *, /, ^ (puissance), MOD ou % (modulo), AND ou &&, OR ou ||, NOT ou !, == (égal), != ou <> (différent), <, <=, >, >=

Pour définir plusieurs entrées et sorties, redimensionnez la table appropriée à la taille de votre choix en cliquant sur le bouton de redimensionnement (l'icône vert +/- situé dans le coin inférieur droit de la table).

Les fonctions ModL de l'environnement de développement ExtendSim peuvent être utilisées dans les équations et permettent d'augmenter considérablement les capacités des équations. Pour plus d'informations sur les fonctions ModL, reportez-vous au manuel de programmation ExtendSim, ou l'aide en ligne ExtendSim (disponible dans le menu Aide).

Les options proposées dans l'onglet Options vous permettent de choisir à quel moment le bloc doit calculer l'équation. Cela peut être à l'initialisation de la simulation, à la fin de la simulation et à d'autres moments. Par défaut, l'équation sera calculée autant que nécessaire durant la simulation.

Chaque entrée doit être nommée dans la table *Input Variables* (Variables d'entrée) pour pouvoir être utilisée dans l'équation. Chaque sortie doit être nommée dans la table *Output Variables* (Variables de sortie). Vous pouvez utiliser les noms par défaut ou spécifier d'autres noms.

Si une entrée est utilisée dans l'équation mais qu'elle n'est pas reliée, ExtendSim vous avertit. Cette condition ne concerne pas les sorties. Si vous le souhaitez, vous pouvez définir une variable de sortie qui n'est pas utilisée dans l'équation.

Le bloc Equation est examiné en détail dans le manuel ExtendSim ; reportez-vous y pour plus d'informations.

Attention : lors de calculs avec des variables entières, les valeurs après la virgule sont perdues.

Variables en entrée

Les variables en entrée proposées par le système sont : DB read value, DB read value using attribute, DB read PRI, DB read PRI using attribute, DB address, DB database index, DB table index, DB field index, DB record index, Static first run init, Static multi run init, Static open model init, Connector. Les variables prévues pour les requêtes dans la BDD commencent par le préfixe "DBQ". DBQ query field value, DBQ query field PRI, DBQ start record, DBQ num records, DBQ current record, DBQ static query init, DBQ current best rank result, DBQ num non-blank ranks, DBQS current best rank result, DBQS num non-blank ranks.

DB read value : lit une valeur à partir d'un emplacement fixe dans une table de BDD. Spécifiez son emplacement dans la colonne "Value".

DB read value using attribute : comme DB read value, mais fait référence à un attribut de l'entité devant comporter une adresse de BDD pour valeur. Lorsque vous faites référence à une variable définie comme **DB read value using attribute**, vous accédez à la valeur de BDD à laquelle fait référence l'adresse contenue dans l'attribut en question. L'attribut d'emplacement de lecture est spécifié dans la colonne Value.

DB read PRI : lit le 'Parent Record Index' (PRI = Index d'enregistrement parent) à partir d'une cellule dans un champ Enfant. L'emplacement de lecture est fixé et spécifié dans la colonne Value.

DB read PRI using attribute (Parent Record Index) : comme *DB read PRI*, cette variable reçoit une Index d'enregistrement parent lue dans un champ Enfant. Mais dans ce cas l'emplacement est défini par un attribut *DB Address* de l'entité. L'attribut d'emplacement de lecture est spécifié dans la colonne Value.

DB address : cette valeur est un pointeur vers une adresse dans la BDD. L'adresse peut pointer sur une base, une table, un champ ou un enregistrement. Elle se compose ainsi de 1 à 4 éléments. Ce type de variable est indispensable car certaines fonctions ModL agissant sur la BDD requièrent une adresse de BDD pour argument.

DB database index - un index unique qui pointe sur une BDD.

DB table index - un index unique qui pointe sur une table.

DB field index - un index unique qui pointe sur un champ.

DB record index - un index unique qui pointe sur un enregistrement.

Static first run init, Static multi run init, and Static open model init : les variables statiques sont des variables locales qui conservent leur valeur entre deux calculs. Les variables statiques peuvent être définies comme des variables d'initialisation pour plusieurs simulations ou uniquement pour la première simulation d'une série. *Static first run init* ne prendra sa valeur initiale qu'au début de la première itération de plusieurs simulations, *Static multi run init* au début de chaque itération. *Static open model init* prendra sa valeur initiale à l'ouverture du modèle. Lorsque vous cliquez sur la colonne "Value" pour une variable statique, on vous demande de saisir la valeur initiale. Dans la table d'affichage, la valeur à la gauche des " :" est la valeur initiale, celle à la droite la valeur courante.

Connector : cette variable prend la valeur du connecteur d'entrée associé.

DBQ query field value : l'utilisateur doit choisir un champ dans la table de requête. Au fur et à mesure du balayage des enregistrements, cette variable prend automatiquement la valeur trouvée pour ce champ pour l'enregistrement courant. (Plus simple que l'appel à `DBDataGetAsNumber()` pour faire référence à l'information localisée dans la table de requête pour l'enregistrement courant.)

DBQ query field PRI : la valeur Parent Record Index (= index enregistrement parent) du champ spécifié pour l'enregistrement courant. Voir note ci-dessous.

DBQ start record : l'enregistrement où commencer le cycle de requête. (Par défaut commence à l'enregistrement 1.)

DBQ num records : le nombre d'enregistrements dans la table de requête.

DBQ current record : l'enregistrement courant actuellement évalué dans le cycle de requête courant.

DBQ static query init : une variable statique initialisée au début de chaque cycle de requête.

DBQ current best rank result : l'actuel meilleur rang d'enregistrement dans le cycle de requête courant.

DBQ num non-blank ranks : nombre d'enregistrements non vides évalués à ce point du cycle de requête courant.

Variables en sortie

Les variables de sortie prédéfinies sont : DB write value, DB write value using attribute, DB write PRI, DB write PRI using attribute, et Connector. Les variables prévues pour les requêtes dans la BDD commencent par le préfixe "DBQ". DBQ record rank, DBQ halt query, DBQ next record, DBQS record rank, DBQS attribute, DBQS item quantity, DBQS item priority, DBQS 3D object ID.

DB write value : envoie la valeur de la variable vers une cellule dans la base de données ExtendSim. L'emplacement de la BDD est défini dans la colonne "Value".

DB write value using attribue : comme DB write value, cette variable écrit sa valeur à un emplacement dans la BDD. Dans ce cas l'emplacement est variable car défini par un attribut Adresse BDD sur l'entité. Cet attribut est défini dans la colonne "Variable".

DB write PRI (Parent Record Index = index enregistrement parent) : comme DB write value, mais l'emplacement d'écriture doit être un champ enfant, car la valeur écrite est l'index de l'enregistrement parent.

DB write PRI using attribute (Parent Record Index) : comme DB write PRI variable, cette variable écrit sa valeur à un emplacement dans la BDD. Dans ce cas l'emplacement est variable car défini par un attribut Adresse BDD sur l'entité. Cet attribut est défini dans la colonne "Variable"

Connector : associe le nom de la variable à un connecteur de sortie.

DBQ record rank : résultat de rang pour l'enregistrement courant.

DBQ halt query : une valeur TRUE stoppe le cycle de requête courant

DBQ next record : indique au bloc l'enregistrement suivant à interroger.

DBQS current best rank result : Enregistrement mémo de meilleur rang dans le cycle de requête courant.

DBQS num non-blank ranks : Nombre d'enregistrements mémo classés à ce stade dans le cycle de requête courant.

DBQS attribute : Pour affecter des valeurs d'attribut à une entité mémo. Si le rang de l'enregistrement mémo est assez bon pour mériter la création d'une entité, les résultats de cet enregistrement seront associés à l'entité mémo sous forme d'attributs DBQS.

DBQS item quantity : Quantité de l'entité à donner à l'entité mémo. Voir ci-dessus DBQS attribute,

DBQS item priority : Priorité de l'entité à donner à l'entité mémo. Voir ci-dessus DBQS attribute.

Suivi des noms dans la BDD (Database Name Tracking)

Notez que les variables de BDD suivantes permettent le suivi des noms : DB read value, DB read PRI, DB write value, DB write PRI, DB address, DB database index, DB table index, DB field index, et DB record index. La fonction de suivi des noms permet que ces variables pointent toujours sur l'emplacement correct dans la base de données, même si celle-ci a subi des modifications dans sa structure après la déclaration de ces variables dans ce bloc. Par exemple, si une variable DB read pointe sur un champ "type voiture", peu importe si le nom ou l'emplacement de ce champ est changé, l'équation saura en faire le suivi.

Onglet Equation

Enter the equation... (entrez l'équation) : Saisissez les instructions. Les variables des tables Variables d'entrée et Variables de sortie doivent être utilisées dans l'équation. Les variables de sortie doivent normalement recevoir une valeur. Toutes les variables d'entrée doivent être utilisées dans l'équation.

Input variables (variables d'entrée) : chaque ligne de la table définit une variable en entrée à utiliser dans l'équation.

Colonne 1 : **Variable Type** (Type) : permet de choisir dans un menu le type de la variable à définir. Voir ci-dessus les différents types. En cliquant sur une cellule, un menu déroulant avec la liste des types possibles apparaît.

Colonne 2 : **Variable Name** (Nom Variable) : définit le nom de la variable, à utiliser dans le corps de l'équation.

Colonne 3 : **Variable Value** (Valeur) : cette colonne affiche différentes choses pour différents types de variables. Pour les variables *Connector*, elle affichera la dernière valeur lue sur le connecteur. Pour les types DB Value, DB Attribute et DB Index, elle affichera une référence à une base de données et peut aussi être utilisée pour éditer cette référence en cliquant sur la cellule. Pour les variables statiques, elle affiche à la fois la valeur de la variable statique à l'initialisation et la valeur actuelle de la variable. Le nombre à gauche de la colonne est la valeur à l'initialisation et le nombre à droite de la colonne est la valeur actuelle. Vous éditez la valeur à l'initialisation en cliquant sur la cellule.

Colonne 4 : **Parent field** (Champ parent) : cette colonne n'a de sens que si la variable de la ligne courante est du type "DBQ read PRI". Dans ce cas la colonne 4 peut au besoin spécifier une "indirection". Cela signifie de ne pas utiliser la table de requête pour effectuer la lecture, mais plutôt la table parent de la variable. Dans ce cas nous utilisons la valeur d'index enregistrement parent comme index d'enregistrement. Il faut également spécifier le champ parent, soit dans une liste, soit en entrant un index, soit via un attribut, soit par un connecteur.

Output Variables (variables de sortie = résultats) : chaque ligne de la table définit une variable en sortie affectée par l'équation.

Colonne 1 : **Variable Type** (Type) : permet de choisir dans un menu le type de la variable à définir. Voir ci-dessus les différents types.

Colonne 2 : **Variable Name** (Nom Variable) : définit le nom de la variable, à utiliser dans le corps de l'équation.

Colonne 3 : **Variable Value** (Valeur) : variable suivant le type de variable. Pour les variables de connecteur, affiche la dernière valeur émise par le connecteur. Pour les types DB Write, affiche la référence de BDD, et permet aussi d'éditer la référence en cliquant sur la cellule.

Colonne 4 : **If no item use** (Si 0 entité...) : permet d'entrer la valeur à utiliser par défaut si aucune entité n'est présente dans le bloc alors que l'équation est calculée.

Open Equation Editor (ouvrir l'éditeur d'équation) : ce bouton ouvre la fenêtre d'éditeur d'équation. Cette fenêtre peut être redimensionnée, et comporte les fonctionnalités de code en couleur, d'indentation automatique ainsi que le remplissage automatique des fonctions.

Enable Debugger (activer Débogueur) : cette case à cocher fait que l'équation est enregistrée sous forme de "code de débogage" pour permettre l'intégration de points d'arrêt. Une équation en format débogage s'exécute plus lentement, et requiert davantage de mémoire. Il est préférable, à la fin du débogage, de décocher cette case.

Set breakpoints (définir points d'arrêt) : ce bouton ouvre la fenêtre "définir des points d'arrêt". Les lignes grises dans la marge de gauche indiquent les seuls endroits où les points d'arrêt peuvent être placés (un point rouge s'affiche alors). Voir le manuel ou l'aide en ligne pour l'utilisation des points d'arrêt.

Test Equation (tester l'équation) : exécute l'équation.

Include files (utiliser des fichiers include) : des fichiers include peuvent être appelés au début de l'équation en cochant cette case. Les fichiers include contiennent du code utilisé dans plusieurs blocs. Les fichiers include simplifient les tâches de programmation qui se répètent dans plusieurs blocs. Ils sont particulièrement utiles quand vous définissez vos propres fonctions pour un bloc et si vous voulez que les autres blocs aient accès à ces fonctions. Par défaut ils sont stockés dans le répertoire Extensions, mais peuvent aussi figurer dans le répertoire du modèle.

Open (ouvrir) : ce bouton ouvre le fichier include mentionné dans cette ligne de la table.

Add Existing (ajouter) : ce bouton ouvre la fenêtre pour sélectionner un fichier include.

Create New (nouveau) : ce bouton crée un nouveau fichier include.

Removed Selected (supprimer sélection) : ce bouton efface de la table les fichiers include sélectionnés.

Prototypes from selected includes (prototypes des includes sélectionnés) : ce menu affiche la liste des prototypes déclarés dans les fichiers Include sélectionnés. Pour qu'un prototype apparaisse dans le menu, il doit avoir été déclaré en haut du fichier Include dans la section prototypes. Lorsque le prototype est affiché dans la zone de texte sous le menu, vous pouvez utiliser un 'glisser-déposer' pour l'inclure dans la fenêtre de l'équation.

Onglet Options

Record rank settings (Paramètres définissant le rang) : Ces paramètres permettent de définir le "meilleur" rang d'enregistrement. Si "plus haut rang" (**highest**) est choisi, c'est l'enregistrement de plus haute valeur qui gagne. Si "plus faible rang" (**lowest**) est choisi, c'est l'enregistrement de plus basse valeur qui gagne. Si "premier rang = TRUE" (**first**) est choisi, l'équation est calculée pour chaque enregistrement jusqu'à trouver un enregistrement dont le résultat soit TRUE ou qu'il n'y ait plus d'enregistrements à évaluer.

Set attribute on pass-through item specifying the record with the best pass-through record rank (Donner un attribut aux entités de passage spécifiant l'enregistrement de meilleur rang) : si cochée,

chaque entité de passage reçoit un attribut indiquant quel enregistrement de la table de requête a eu le meilleur rang.

Records ranked with XX are equals (Résultats de +/- considérés égaux) : cette option permet de définir une marge d'erreur pour les valeurs calculées de *Result rank*. Les valeurs à l'intérieur de la plage sont considérées comme égales.

Enable the spawning of items (translate some subset of records into items) *Activer les entités Mémo (transformer des enregistrements en entités)* : cochez cette case pour activer la création d'entités mémo. Un second connecteur de sortie discret d'ajoute alors sur l'icône, et de nouvelles variables (préfixe DBQS pour database query spawn), deviennent disponibles. Les résultats de l'équation pour les entités de passage sont calculés indépendamment de ceux des entités mémo. Par exemple, les résultats pour une entité de passage pourraient être pris dans l'enregistrement 6 (parce que 6 avait le meilleur rang des entités de passage), alors que les entités mémos utiliseraient les données de l'enregistrement 2 (parce que 2 avait le meilleur rang des entités mémo). Lorsqu'un mémo est créé, il représente un enregistrement de la table de requête. Vous pouvez considérer cette nouvelle entité comme un "enregistrement mémo".

Spawn records with (Créer des entités Mémo pour) : Ces paramètres contrôlent quels enregistrements deviennent des mémos. 'le rang d'enregistrement Mémo le plus élevé' : c'est l'enregistrement de meilleur rang mémo qui devient entité mémo. 'le rang d'enregistrement Mémo le plus bas' : c'est l'enregistrement de plus mauvais rang mémo qui devient entité mémo. 'le premier enreg. Mémo de rang True;' : c'est le premier enregistrement qui reçoit un rang mémo TRUE qui devient entité mémo. 'tous les enregistrements Mémo True' : tous les enregistrements de rang mémo TRUE deviennent des entités mémo.

Pass-through item (L'entité de passage) : Paramètres qui contrôlent le destin de l'entité de passage. Si "passe toujours" est choisi, l'entité sera toujours autorisée à quitter le bloc Query Equation (I). Si "ne passe jamais" est choisi, l'entité sera toujours détruite. Si "passe si aucune entité Mémo créée" est choisi, l'entité sera détruite si au moins une entité mémo est créée. Si "passe si au moins 1 entité Mémo créée" est choisi, l'entité sera détruite si aucune entité mémo n'est créée.

Copy attributes from pass-through item to spawned items (Copier les attributs de l'entité de passage sur les entités Mémo) : si cochée, toutes les valeurs d'attribut de l'entité de passage sont transmises à chaque mémo.

Set attribute on spawn specifying the number of spawns generated by the pass-through item (Donner aux entités Mémo un attribut contenant le nb de mémos générés par l'entité de passage) : si cochée, chaque mémo reçoit un attribut indiquant combien de mémos ont été créés en réponse à l'entité de passage courante.

Set attribute on spawn specifying the record for which the spawn was created (Donner aux entités Mémo un attribut indiquant l'enregistrement dont est issu le mémo) : si cochée, chaque mémo reçoit un attribut indiquant quel enregistrement de la table de requête il représente.

Move Batch ID from pass-through item to the spawned item (Transférer le 'Batch ID' de l'entité de passage à l'entité Mémo) : si cochée, l'identification de lot (Batch ID) de l'entité de passage est donnée à l'entité mémo.

Convert date time data for DB read and DB write variables (Convertir données calendaires pour les variables DB read et DB write) : si une variable DB read pointe sur un champ "Date Time", cocher cette case signifie que le bloc Equation convertira la valeur lue dans le format de l'unité de temps de la simulation avant de calculer l'équation. Pour une variable DB write, la conversion a lieu dans l'autre sens.

Show input connector labels : affiche le label sur les connecteurs d'entrée.

Show input connector values : affiche la valeur sur les connecteurs d'entrée.

Show output connector labels : affiche le label sur les connecteurs de sortie.

Show output connector values : affiche la valeur sur les connecteurs de sortie.

Use block seed (Utiliser la base aléatoire du bloc) : cette option permet de définir une base aléatoire pour l'équation. Cela n'est utile que si le bloc a accès aux fonctions de nombre aléatoire.

Expand records if necessary (Ajouter enregistrement si nécessaire) : si la tentative d'écriture d'une variable 'attribut' va au-delà du nombre d'enregistrements de la table et que cette option est cochée, la table sera agrandie automatiquement. Si l'option n'est pas cochée, cela provoque une erreur.

Initialize results at start of run to (Résultats initialisés à) : spécifie quelle valeur doivent prendre les variables en sortie en début de simulation (avant l'arrivée de la première entité).

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Connecteurs

L'entité entrante est lue puis renvoyée telle quelle en sortie, parfois avec des attributs supplémentaires ou modifiés.

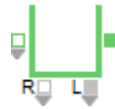
Il y aura un connecteur de sortie continu pour chaque résultat d'équation associé à un connecteur.

Il y aura un connecteur d'entrée continu pour chaque entrée de l'équation associé à un connecteur.

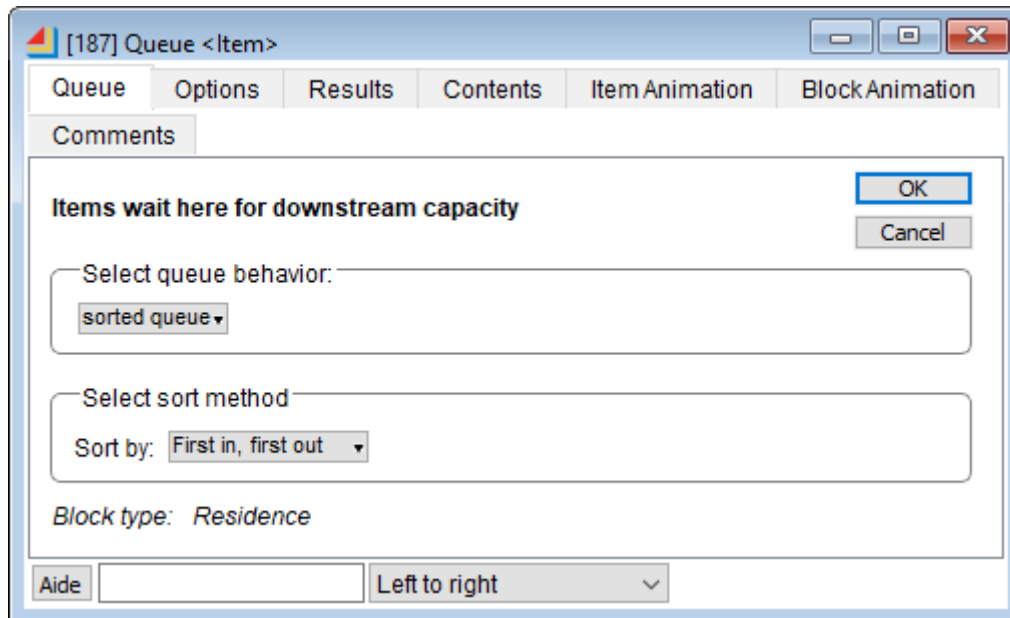
Animation

Un point de couleur peut apparaître sur l'icône lorsque l'équation est sur le point d'être calculée. Pour un calcul en début de simulation, le point est vert ; il est bleu pour un calcul toutes les N unités de temps, et rouge dans le cas d'un calcul en fin de simulation. Si aucun point n'apparaît, c'est que le bloc réagit à des messages.

Queue



Reçoit des entités en file d'attente et les libère selon l'algorithme choisi.



L'onglet Queue permet de choisir le fonctionnement global de la file, qui est soit une file triée (**sorted queue**) par un algorithme, soit une file faisant appel à des ressources : **Ressource pool queue** faisant appel aux ressources d'un bloc Resource Pool, pour les associer aux entités. Ou **Advanced Resources**, pour la gestion de ressources complexes via le bloc Resource Manager.

L'algorithme d'une file triée est choisi parmi **File triée par attribut; File FIFO; File LIFO** et **File selon priorités..** Les files selon attribut trient les entités en fonction d'un attribut défini. Premier-entré-premier-sorti (first-in-first-out = FIFO) est la file d'attente simple de base. Dernier-entré-premier sorti (Last in, first out = LIFO) emploie l'ordre inverse, aussi appelé pile. La file selon priorités utilise la priorité portée par chaque entité pour déterminer l'ordre de sortie. Pour utiliser des algorithmes plus complexes, faites appel au bloc Queue Equation.

Si vous choisissez une file ressource, vous devrez spécifier les ressources nécessaires dans la portion « Select resource pools and set quantity » (Sélectionner pools et quantités) du dialogue. Le choix *Take from any resource pool* (Utiliser n'importe lequel des pools) sur la droite spécifie que les éléments de ressources peuvent provenir de n'importe lequel des pools listés dans la table. Le choix *All resource pools are required* (Tous les pools sont requis) spécifie que la liste exacte des ressources est requise pour libérer l'entité. Dans le premier cas, le nombre utilisé peut varier, par exemple un élément du pool 1, ou deux éléments du pool 2.

Dans la table qui liste les ressources utilisées, remarquez la petite flèche descendante à la droite de la colonne Quantity. En cliquant dessus, vous remplissez la colonne Quantity à l'imitation de la valeur de la première ligne.

Pour l'utilisation des ressources complexes, reportez-vous à l'aide du bloc Resource Manager.

Connecteurs et éléments de dialogue changent en fonction de vos choix. Par exemple, si l'option **Renège** (Abandon) est cochée, un second connecteur de sortie discret apparaît à la droite de l'icône.

Onglet Queue

Select queue behavior (Comportement de la file) : vous choisissez principalement si la file sert à associer des ressources (par le système des Pools ou éventuellement par celui des ressources complexes), ou si c'est une file d'attente triée. Dans les trois cas, vous devrez préciser le comportement sur d'autres éléments du dialogue.

File classée

Sort by (Tri par) : sélectionnez l'algorithme de tri de la file. Les choix possibles sont : Valeur d'attribut, FIFO, LIFO et Priorité (voir en introduction). Dans le cas de l'attribut, vous devez préciser l'attribut concerné, et l'ordre du tri (ascendant ou descendant).

File Ressource

Tableau des ressources

Colonne 1 (Pool) : sélectionnez dans cette colonne quel(s) pool(s) seront requis pour permettre à l'entité de quitter la file.

Colonne 2 (Quantity) : sélectionnez dans cette colonne le nombre d'éléments de chaque ressource requis.

Take from any resource pool (Utiliser n'importe lequel des pools) : seule l'une des ressources spécifiée est requise pour que l'entité quitte la file. Si cette option est sélectionnée, le nombre de ressource effectivement utilisé peut varier en fonction de la valeur dans la colonne quantité de la ligne en cours. Le bloc cherchera à allouer les éléments des pools en commençant par celui du haut. Si un attribut est sélectionné dans le menu **Record Resource Pool on attribute** (Mémoriser le Pool dans l'attribut), le pool alloué sera mémorisé. Le second choix, **All resource pools are required** (Tous les pools sont requis), spécifie que le nombre requis pour chaque pool doit être disponible pour que l'entité puisse quitter la file.

Resource quantity from (Quantité lue sur) : choisissez une des options : **Table**, **Connector** et **Attribute** pour définir d'où provient l'information de quantité.

Show block numbers in popup (N° de bloc dans menu) : si cochée, le numéro du bloc Resource Pool sera affiché avec son nom dans le menu déroulant.

Show resources on icon (Voir Pools sur l'icône) : si cochée, affiche les noms des blocs Pool pour lesquels une entité attend des éléments.

If resource pool names are duplicated, find only closest (Si noms de Pools en double, prendre le plus proche) : cochez cette case pour éviter tout nom en double dans le menu de choix du Pool. Si deux pools ont le même nom, seul celui du bloc le plus proche sera affiché. Sert notamment pour ne prendre en compte que les blocs dans un même bloc hiérarchique.

Onglet Options

Maximum queue length (Longueur maximum) : capacité maxi de la file. Détermine le niveau de saturation de la file. La case 'infini' donne une capacité infinie.

Reneg items (Abandon) : si cette case est cochée, un connecteur de sortie s'ajoute à l'icône, et le bloc pourra faire sortir des entités par abandon en fonction des autres paramètres de l'onglet.

Reneg items after (après) : spécifie le délai associé au comportement d'abandon. Lorsque ce temps est dépassé, l'entité sort par le connecteur de sortie du bas ; sinon elle sort par le connecteur habituel du milieu.

Reneg items immediately (dès que : le connecteur Abandon vaut TRUE). Si cette option est sélectionnée, les entités abandonnent (quittent le bloc par le connecteur d'abandon) lorsque la valeur du connecteur d'abandon (A) est TRUE (≥ 0.5).

Calculate waiting costs (Calculer les coûts d'attente) : si sélectionné, la file accumule un coût d'attente en fonction de l'attribut "_rate" de chaque entité. Ce coût est ajouté à l'attribut "_cost" existant

sur chaque entité lorsqu'elle quitte la file. La somme de tous les coûts d'attente figure dans le champ "Total cost" du dialogue.

Record Resource Pool on attribute (Mémoriser le Pool dans l'attribut) : sélectionnez l'attribut dans lequel sera stocké le nom du pool utilisé par l'entité.

Record resource quantity on attribute (Mémoriser la quantité dans l'attribut) : Sélectionnez l'attribut dans lequel sera stocké la quantité d'éléments du pool utilisés par l'entité. Cette option n'est accessible que lorsque **Take from any resource pool** est sélectionné dans l'onglet Queue. Utilisez cette option si vous faites appel à plusieurs ressources alternatives, pour garantir que les éléments reviennent correctement à leur pool respectif ensuite.

Wait time based on Shift (Attente en fonction de bloc Shift) : si un horaire on/off est sélectionné, cette option spécifie que les statistiques d'Attente pour la file ne seront pas accumulées lorsque l'horaire est Off.

Minimum wait time (Attente minimum) : si cochée, les entités passant par la file attendront au minimum le temps indiqué. Permet de modéliser un transport ou convoyeur simple.

Onglet Results

Current queue length (Longueur en cours) : la longueur de la file à cet instant de la simulation. (ou en fin de simulation, si elle est terminée.)

Average queue length (Longueur moyenne) : nombre moyen d'entités dans la file.

Maximum queue length (Longueur maxi) : nombre maximum d'entités observé.

Current queue wait (Attente en cours) : attente de la dernière entité ayant quitté la file.

Average queue wait (Attente moyenne) : moyenne des durées d'attente.

Maximum queue wait (Attente maxi) : durée maximale passée par une entité.

Arrivals (Arrivées) : nombre d'entités entrées dans la file.

Departures (Départs) : nombre d'entités ayant quitté la file.

Renegs (Abandons) : nombre d'entités ayant abandonné la file.

Utilization : pourcentage du temps où le bloc a été utilisé sur l'ensemble de la simulation. Intègre le temps passé par les entités dans la file parce qu'aucun bloc aval n'était capable de les absorber.

Total cost (Coût total) : le coût total d'attente de toutes les entités passées par la file.

Inclure les entités dans la file en fin de simu : si cochée, les entités qui demeurent dans la file sont ajoutées dans le calcul des statistiques.

Onglet Contents

Cet onglet existe pour les blocs Queue, Queue Equation, Resource Item et Activity, pour permettre de connaître soit le contenu du bloc à un instant donné, soit l'historique de toutes les entités qui ont traversé le bloc. Notez que cette fonction ralentit l'exécution et charge la mémoire : elle est conseillée pour la phase de débogage. Un onglet similaire existe pour le bloc Executive, centralisant tous les onglets Contents du modèle.

Enable current contents (activer le suivi du contenu) : en cochant cette case vous activez les options de l'onglet.

Table display (la table montre) : le menu permet de choisir parmi les des modes de fonctionnement de l'onglet.

Current contents mode (contenu actuel) : dans ce mode la table affiche les infos sur les blocs qui résident actuellement dans le bloc. Dans ce mode, les 5 premières colonnes de la table sont fixes. Par le bouton de redimensionnement "+/-" vous pouvez ajouter des colonnes et ainsi suivre d'autres

propriétés. Colonne 1, Heure entrée, indique à quelle heure l'entité est entrée dans le bloc. Colonne 2, Arrêt, indique si l'entité est en arrêt (0 si non, 1 si oui). Colonne 3, Horaire, indique si l'entité est en horaire actif si 1, en horaire off si 0. Colonne 4, Durée arrêt, indique le temps total passé en arrêt ou en horaire off. Colonne 5, Heure fin, indique l'heure de fin du traitement. Si cette colonne est vide, c'est que l'entité est encore en traitement. Si une valeur apparaît, c'est que l'entité n'a pas pu quitter le bloc, ce qu'elle cherche à faire quand elle est terminée. Une entité non bloquée ayant fini son traitement ne figure normalement plus dans la table.

Historical log mode (historique) : dans ce mode la table affiche toutes les entités qui sont passées par le bloc durant la simulation. Cela signifie qu'une ligne est consacrée à chaque entité, et que la table peut devenir très grande. Dans ce mode aussi, les 5 premières colonnes de la table sont fixes. Les colonnes Heure entrée, Durée arrêt et Heure fin sont identiques à celles du mode *Current contents*. Colonne 4, Heure sortie, affiche l'heure à laquelle l'entité a quitté le bloc. Si une entité a une valeur dans la colonne Heure fin mais pas dans heure sortie, c'est qu'elle est bloquée dans le bloc. Colonne 5, Sortie, décrit la manière dont l'entité a quitté le bloc. "Normal" signifie par le connecteur habituel itemOut, "Préemption" signifie que l'entité a été préemptée.

Display contents (afficher contenu) : en mode *Current contents*, un menu apparaît sous la table, permettant de choisir quand afficher le contenu : 1.) durant la simulation, 2.) uniquement sur clic du bouton Instantané, ou 3.) à la toute fin de la simulation.

Control log time (bornes pour historique) : en mode *Historical*, cette case à cocher apparaîtra, permettant de définir une fenêtre de temps durant laquelle l'historique sera enregistré.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Animate H-block objects X to Y (Animer les objets de bloc H x jusque y) : anime la longueur de la file sur plusieurs objets d'animation présents sur le bloc hiérarchique englobant ce bloc. La plage est celle des numéros d'objets d'animation sur l'icône du bloc hiérarchique. Chaque objet dans cette plage animera une entité différente.

Connecteurs

En entrée l'entité discrète qui ressort après avoir attendu dans la file ou non.

Si l'option **Renegé** est cochée, un second connecteur de sortie discret apparaît à la droite de l'icône. Il sert pour les entités qui quittent la file par abandon.

Si la file est une file ressource et que l'option *Resource quantity from connector* est choisie, une matrice de connecteurs apparaît permettant d'entrer les quantités pour les ressources.

Connecteurs d'entrée continus : (dans l'ordre d'apparition.)

R : (Abandon) Signal pour provoquer l'abandon d'une entité.

L : (Longueur) Définit la longueur maximum de la file.

Connecteurs de sortie continus : (dans l'ordre d'apparition.)

L : (Longueur) Nombre d'entités dans le bloc.

W : (Attente : Wait) La durée d'attente de l'entité sortie en dernier du bloc.

F : (Plein : Full) Emet 1 si le bloc est plein, sinon émet 0.

P : (Priorité) Priorité de l'entité sortie en dernier du bloc.

Ave L : Longueur moyenne.

Ave W : (Attente : Wait) Attente moyenne.

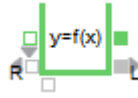
TD : Total des entités sorties.

Animation

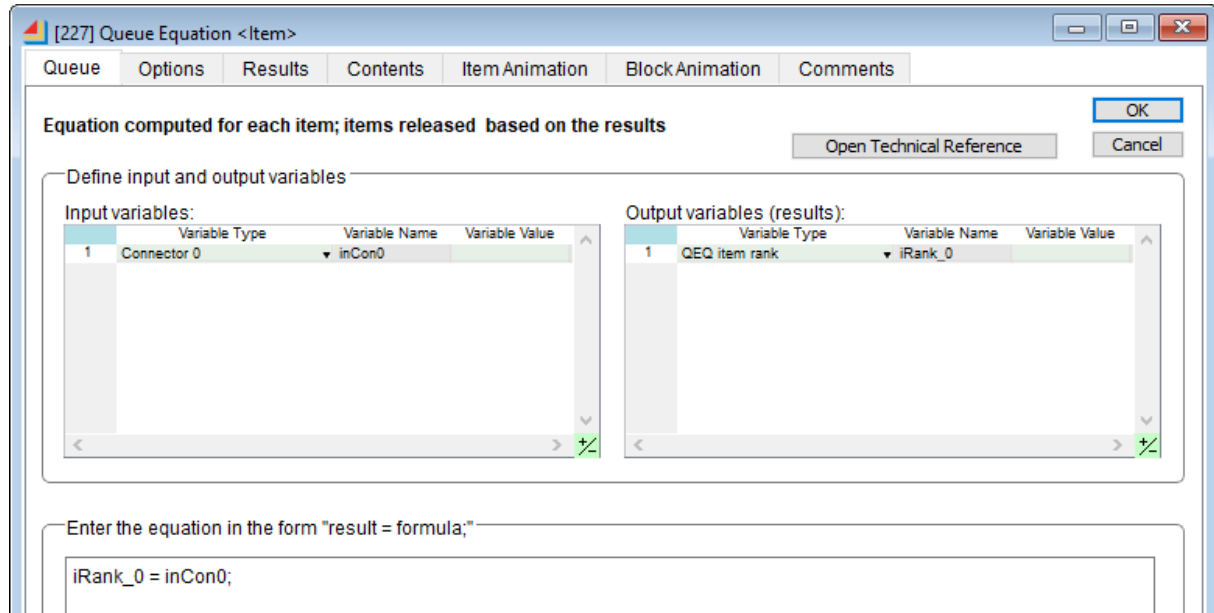
Les entités accumulées dans le bloc s'animent par leur symbole 2D. Jusqu'à 4 entités peuvent figurer sur automatiquement l'icône du bloc. Au-delà, un nombre apparaît pour indiquer le nombre total d'entités.

Le bloc reflète s'il est en mode file classée ou file ressource.

Queue Equation



Cette file d'attente libère des entités en fonction du résultat d'une équation.



Le bloc s'appuie sur les résultats d'une équation utilisateur pour déterminer quelle entité va quitter la file d'attente. Chaque fois que la file doit être réévaluée, l'équation est calculée pour chaque entité, et celle qui donne le meilleur résultat sort de la file. L'équation doit définir au moins une variable de type *QEItem rank* (celle déterminant le classement) dans la table des variables en sortie. Cette variable *QEItem rank* servira à résoudre l'algorithme de tri. En fonction de l'option choisie dans le menu déroulant "Libérer les entités...", l'entité ayant le plus haut, le plus bas, ou le premier résultat TRUE (≥ 0.5) est libérée. Si vous définissez plus d'une variable *QEItem rank*, elle servira pour départager en cas d'égalité. (i.e., si la première variable *QEItem rank* ne détermine rien, la seconde sera considérée, etc.) Si la valeur de *_Result rank* est BLANK (vide) ou NOVALUE, l'entité ne sera pas libérée.

Un exemple simple d'une équation imitant le comportement d'une file FIFO serait $qRank0 = ArriveTime0$.

Dans le menu **Libérer les entités (Release)** il faudrait choisir **lowest ranked item** (plus bas résultats d'abord) pour obtenir le comportement FIFO.

Par défaut la file se réévalue (= recalcule l'équation pour toutes les entités) chaque fois qu'une entité entre ou sort, lorsqu'un message arrive sur un connecteur d'entrée continue, et lorsqu'un bloc en attente d'obtenir une entité. Cependant plusieurs options de l'onglet Options permettent de contrôler ce qui déclenche un cycle de calcul. Comme les temps de calcul peuvent devenir très importants si la file comporte de nombreuses entités, lorsqu'une entité de quantité >1 entre dans la file, elle n'est pas décomposée en autant d'entités individuelles, mais conserve sa quantité initiale, contrairement à ce qui se passe avec une file classique.

Le(s) résultat(s) peuvent en option être affectés à des propriétés de l'entité. Vous pouvez utiliser tous les opérateurs et fonctions ExtendSim, ainsi que tout ou partie des entrées dans la rédaction de l'équation. Il n'y a pas de limite au nombre d'entrées et de sorties : le nombre de connecteurs sur l'icône reflètera la taille de la table (pour les lignes faisant référence à des connecteurs).

Les opérateurs ExtendSim sont : +, -, *, /, ^ (exponentielle), MOD ou % (modulo), AND ou &&, OR ou ||, NOT ou !, == (égal), != ou <> (différent de), <, <=, >, >=

Pour définir plusieurs entrées/sorties, cliquez sur le bouton vert de redimensionnement (+-) dans le coin inférieur droit de la table.

L'emploi des fonctions ModL peut élargir radicalement ce que vous pouvez effectuer par les équations. Voyez l'aide en ligne ou le manuel du développeur pour connaître ces fonctions.

Le bloc exécute l'équation une fois pour chaque entité, à chaque fois que l'ordre de la file doit être reconsidéré.

Chaque entrée doit être nommée dans la table *Input Variables* (Variables d'entrée) pour pouvoir être utilisée dans l'équation. Chaque sortie doit être nommée dans la table *Output Variables* (Variables de sortie). Vous pouvez utiliser les noms par défaut ou spécifier d'autres noms. ExtendSim vous avertit si des entrées utilisées dans l'équation ne sont pas reliées. Ce n'est pas le cas en sortie : vous pouvez définir des variables en sortie qui ne sont pas utilisées dans l'équation, si cela vous rend service.

Variables en entrée : En plus des propriétés des entités, les variables en entrée proposées par le système sont les suivantes :

DB read value : lit une valeur à partir d'un emplacement fixe dans une table de BDD. Spécifiez son emplacement dans la colonne "Value".

DB read value using attribute : comme *DB read value*, mais fait référence à un attribut de l'entité devant comporter une adresse de BDD pour valeur. Lorsque vous faites référence à une variable définie comme *DB read value using attribute*, vous accédez à la valeur de BDD à laquelle fait référence l'adresse contenue dans l'attribut en question. L'attribut d'emplacement de lecture est spécifié dans la colonne Value.

DB read PRI : lit le 'Parent Record Index' (PRI = Index d'enregistrement parent) à partir d'une cellule dans un champ Enfant. L'emplacement de lecture est fixé et spécifié dans la colonne Value.

DB read PRI using attribute (Parent Record Index) : comme *DB read PRI*, cette variable reçoit une Index d'enregistrement parent lue dans un champ Enfant. Mais dans ce cas l'emplacement est défini par un attribut *BD Address* de l'entité. L'attribut d'emplacement de lecture est spécifié dans la colonne Value.

DB address : cette valeur est un pointeur vers une adresse dans la BDD. L'adresse peut pointer sur une base, une table, un champ ou un enregistrement. Elle se compose ainsi de 1 à 4 éléments. Ce type de variable est indispensable car certaines fonctions ModL agissant sur la BDD requièrent une adresse de BDD pour argument.

DB database index - un index unique qui pointe sur une BDD.

DB table index - un index unique qui pointe sur une table.

DB field index - un index unique qui pointe sur un champ.

DB record index - un index unique qui pointe sur un enregistrement.

Static first run init, Static multi run init et Static open model init : les variables statiques sont des variables locales qui conservent leur valeur entre deux calculs. Les variables statiques peuvent être définies comme des variables d'initialisation pour plusieurs simulations ou uniquement pour la première simulation d'une série. *Static first run init* ne prendra sa valeur initiale qu'au début de la première itération de plusieurs simulations, *Static multi run init* au début de chaque itération. *Static open model init* prendra sa valeur initiale à l'ouverture du modèle. Lorsque vous cliquez sur la colonne "Valeur" pour une variable statique, on vous demande de saisir la valeur initiale. Dans la table d'affichage, la valeur à la gauche des " : " est la valeur initiale, celle à la droite la valeur courante.

Connector : la valeur de cette variable est celle du connecteur d'entrée associé.

QEQ arrival time : l'heure à laquelle l'entité est entrée dans la file.

QEQ FIFO position : la position FIFO de l'entité dans la file. Cette position détermine l'ordre dans lequel des entités subissent l'équation.

QEQ current best item rank : le meilleur (plus haut ou plus bas) rang de classement d'entité pour le cycle de calcul en cours.

QEQ attrib last item to exit : indique la valeur d'un attribut donné de la dernière entité qui a quitté le bloc.

QEQ num items in queue : le nombre d'entités actuellement dans la file.

QEQ static calc cycle init : une variable statique initialisée à sa valeur de départ au début de chaque cycle de calcul. Changes to this variable remain fixed (static) across equation calculations during the current calculation cycle.

Variables en sortie : En plus des propriétés des entités, les types possibles pour les sorties sont les suivants :

DB write value : envoie la valeur de la variable vers une cellule dans la base de données ExtendSim. L'emplacement de la BDD est défini dans la colonne "Value".

DB write value using attribute : comme DB write value, cette variable écrit sa valeur à un emplacement dans la BDD. Dans ce cas l'emplacement est variable car défini par un attribut Adresse BDD sur l'entité. Cet attribut est défini dans la colonne "Variable".

DB write PRI (Parent Record Index = index enregistrement parent) : comme DB write value, mais l'emplacement d'écriture doit être un champ enfant, car la valeur écrite est l'index de l'enregistrement parent.

DB write PRI using attribute (Parent Record Index) : comme DB write PRI variable, cette variable écrit sa valeur à un emplacement dans la BDD. Dans ce cas l'emplacement est variable car défini par un attribut Adresse BDD sur l'entité. Cet attribut est défini dans la colonne "Nom variable".

Connector : associe le nom de la variable à un connecteur de sortie.

QEQ item rank : définit le rang (position) de chaque entité dans la file. Au moins une des variables en sortie doit être de ce type. D'autres variables du même type serviront en cas d'égalité sur la première.

QEQ halt calculation cycle : si vaut True, cette variable interrompra avant l'heure le cycle de calcul courant.

L'emploi des blocs à équation est examiné en détail dans le manuel ExtendSim.

Attention : lors de calculs avec des variables entières, les valeurs après la virgule sont perdues.

Suivi des noms dans la BDD (Database Name Tracking)

Notez que les variables de BDD suivantes permettent le suivi des noms : DB read value, DB read PRI, DB write value, DB write PRI, DB address, DB database index, DB table index, DB field index, et DB record index. La fonction de suivi des noms permet que ces variables pointent toujours sur l'emplacement correct dans la base de données, même si celle-ci a subi des modifications dans sa structure après la déclaration de ces variables dans ce bloc. Par exemple, si une variable DB read pointe sur un champ "type voiture", peu importe si le nom ou l'emplacement de ce champ est changé, l'équation saura en faire le suivi.

Onglet Queue

Input variables (Variables d'entrée) : chaque ligne de la table définit une variable en entrée à utiliser dans l'équation.

Colonne 1 (Type) : permet de choisir dans un menu le type de la variable à définir. Voir ci-dessus les différents types.

Colonne 2 (Nom Variable) : définit le nom de la variable, à utiliser dans le corps de l'équation.

Colonne 3 (Valeur) : en fonction du type de variable. Pour les variables de connecteur, affiche la dernière valeur présente sur le connecteur. Pour les variables de types DB, affiche la référence de BDD, et permet aussi d'éditer la référence en cliquant sur la cellule. Pour les variables statiques, affiche la valeur d'initialisation (à gauche) et la valeur courante (à droite) de la variable. La valeur d'initialisation peut être éditée en cliquant sur la cellule.

Output variables (Variables de sortie = résultats) : chaque ligne de la table définit une variable en sortie affectée par l'équation.

Colonne 1 (Type) : permet de choisir dans un menu le type de la variable à définir. Voir ci-dessus les différents types.

Colonne 2 (Nom Variable) : définit le nom de la variable, à utiliser dans le corps de l'équation.

Colonne 3 (Valeur) : variable suivant le type de variable. Pour les variables de connecteur, affiche la dernière valeur émise par le connecteur. Pour les types DB Write, affiche la référence de BDD, et permet aussi d'éditer la référence en cliquant sur la cellule.

Enter the equation (Entrez une équation..) : tapez une équation dans la zone de texte. Les variables utilisées dans les tables *Input Variables* et *Output Variables* doivent être utilisées dans l'équation. Des valeurs doivent être attribuées aux variables de sortie. Toutes les variables d'entrée définies doivent être utilisées dans l'équation.

Open Equation Editor (Ouvrir l'éditeur d'équation) : ce bouton ouvre la fenêtre d'éditeur d'équation. Cette fenêtre peut être redimensionnée, et comporte les fonctionnalités de code en couleur, d'indentation automatique ainsi que le remplissage automatique des fonctions.

Enable Debugger (Activer Débogueur) : cette case à cocher fait que l'équation est enregistrée sous forme de "code de débogage" pour permettre l'intégration de points d'arrêt. Une équation en format débogage s'exécute plus lentement, et requiert davantage de mémoire. Il est préférable, à la fin du débogage, de décocher cette case.

Set Breakpoints (Définir points d'arrêt) : ce bouton ouvre la fenêtre "Définir des points d'arrêt". Les lignes grises dans la marge de gauche indiquent les seuls endroits où les points d'arrêt peuvent être placés (un point rouge s'affiche alors). Voir le *Guide d'utilisation* ou l'aide en ligne pour l'utilisation des points d'arrêt.

Tester l'équation : exécute l'équation.

Use include files (Utiliser des fichiers include) : des fichiers include peuvent être appelés au début de l'équation en cochant cette case. Les fichiers include contiennent du code utilisé dans plusieurs blocs. Les fichiers include simplifient les tâches de programmation qui se répètent dans plusieurs blocs. Ils sont particulièrement utiles quand vous définissez vos propres fonctions pour un bloc et si vous voulez que les autres blocs aient accès à ces fonctions. Par défaut ils sont stockés dans le répertoire Extensions, mais peuvent aussi figurer dans le répertoire du modèle.

Open (Ouvrir) : ce bouton ouvre le fichier include mentionné dans cette ligne de la table.

Add existing (Ajouter) : ce bouton ouvre la fenêtre pour sélectionner un fichier include.

Create new (Nouveau) : ce bouton crée un nouveau fichier include.

Remove Selected (Supprimer sélection) : ce bouton efface de la table les fichiers include sélectionnés.

Prototypes from selected includes (Prototypes des include sélectionnés) : ce menu affiche la liste des prototypes déclarés dans les fichiers include sélectionnés. Pour qu'un prototype apparaisse dans le menu, il doit avoir été déclaré en haut du fichier include dans la section prototypes. Lorsque le prototype est affiché dans la zone de texte sous le menu, vous pouvez utiliser un 'glisser-déposer' pour l'inclure dans la fenêtre de l'équation.

Onglet Options

Comportement

Maximum queue length (Longueur maximum) : capacité maxi de la file. Détermine le niveau de saturation de la file. La case 'infini' donne une capacité infinie.

Calculate waiting costs (Calculer les coûts d'attente) : si sélectionné, la file accumule un coût d'attente en fonction de l'attribut "_rate" de chaque entité. Ce coût est ajouté à l'attribut "_cost" existant sur chaque entité lorsqu'elle quitte la file. La somme de tous les coûts d'attente figure dans le champ "Coût total" du dialogue.

Release (Libérer les entités) : ce menu permet de contrôler la manière dont les résultats de l'équation déterminent la libération des entités de la file. Les options sont : **highest ranked item** (plus haut résultat d'abord) - l'entité dont la valeur calculée par l'équation est la plus haute est libérée en premier ; **lowest ranked item** (plus bas résultats d'abord) - l'entité dont la valeur calculée par l'équation est la plus basse est libérée en premier. **first True ranked item** (Premier résultat TRUE d'abord) - la première entité pour laquelle l'équation calcule une valeur TRUE est libérée en premier. Avec la dernière option, **All True ranked items** (tous résultats TRUE), toutes les entités qualifiées seront présentées au départ en ordre FIFO. Une entité présentée au départ peut immédiatement sortir de la file et ne sera plus jamais évaluée par l'équation.

Results within +/- N are equal (Résultats de +/- considérés égaux) : cette option permet de définir une marge d'erreur pour les valeurs calculées de *QEQ item rank*. Les valeurs à l'intérieur de la plage sont considérées comme égales.

Définit quand calculer l'équation

When receive input value connector msgs (A réception d'un message de connecteur d'entrée continu) : cette option permet que l'équation se calcule chaque fois qu'un message est reçu sur un connecteur d'entrée continu. Elle est complétée par deux autres : l'une pour englober tous les connecteurs d'entrée, ou un seul ; l'autre pour réagir **immédiatement** ou après un **événement de durée zéro**. Si l'option de réponse immédiate est choisie, la file n'essaiera d'envoyer qu'une seule entité en sortie. Si l'option après un événement de durée zéro est choisie, la file tentera d'envoyer le plus d'entités possible en sortie.

When receive database link alert msgs (A réception d'un message d'alerte de lien BDD) : cette option permet que l'équation se calcule si au moins l'une des variables d'entrée est de type DB read value ou DB read PRI et si les données associées à cette variable ont changé. La case à cocher est complétée par deux autres : l'une pour réagir à une alerte de liens pour **une seule ou toutes les variables** DB read, l'autre pour réagir immédiatement ou après un événement de durée zéro. Si l'option de réponse immédiate est choisie, la file n'essaiera d'envoyer qu'une seule entité en sortie. Si l'option après un événement de durée zéro est choisie, la file tentera d'envoyer le plus d'entités possible en sortie.

When item enters (A l'entrée de l'entité) : si cette case est cochée, l'équation se calcule une fois pour chaque entité présente, lorsqu'une nouvelle entité entre.

When item exits (A la sortie de l'entité) : si cette case est cochée, l'équation se calcule une fois pour chaque entité présente, lorsqu'une nouvelle entité sort.

When downstream block tries to pull an item (Lorsque le bloc aval demande une entité) : si cette case n'est pas cochée, toute tentative par un bloc aval de tirer une entité dans la file sera ignorée.

Use the "Calc event time" output variable (Utiliser la variable de sortie "Calc event time") : si cette case est cochée, un nouveau type de variable de sortie apparaît dans la liste proposée. Si cette variable est utilisée dans l'équation, sa valeur sera utilisée pour publier un futur événement de recalcul.

Respond to value connector message (Réagir aux messages de connecteur de sortie) : par défaut l'option propose 'sous condition' ce qui signifie qu'un message sur un connecteur de sortie ne déclenchera un cycle de calcul que si le message provient d'un bloc bloqueur en aval (ex. Select Item

Out). L'autre option, "Jamais", force la Queue Equation à ignorer tous les messages reçus sur des connecteurs de sortie continus.

Autres options

Use random seed (Base aléatoire) : si la case est cochée et une valeur fournie, cette base aléatoire sera utilisée localement pour les calculs aléatoires faits par le bloc.

Expand records if necessary (Ajouter enregistrement si nécessaire) : si la tentative d'écriture d'une variable 'attribut' va au-delà du nombre d'enregistrements de la table et que cette option est cochée, la table sera agrandie automatiquement. Si l'option n'est pas cochée, cela provoque une erreur.

Convert date time data for DB read and DB write variables (Convertir données calendaires pour les variables DB read et DB write) : si une variable DB read pointe sur un champ "Date Time", cocher cette case signifie que le bloc Equation convertira la valeur lue dans le format de l'unité de temps de la simulation avant de calculer l'équation. Pour une variable DB write, la conversion a lieu dans l'autre sens.

Labels connecteurs d'entrée : si la case est cochée, tous les connecteurs d'entrée afficheront leur label.

Valeurs connecteurs d'entrée : si cochée, tous les connecteurs d'entrée afficheront leur valeur

Labels connecteurs de sortie : si cochée, tous les connecteurs de sortie afficheront leur label.

Valeurs connecteurs de sortie : si cochée, tous les connecteurs de sortie afficheront leur valeur.

Enable Advanced Resources (AR) : ce cadre concerne les Ressources complexes, voir bloc Resource Manager.

Onglet Results

Current queue length (Longueur en cours) : la longueur de la file à cet instant de la simulation. (ou en fin de simulation, si elle est terminée.)

Average queue length (Longueur moyenne) : nombre moyen d'entités dans la file.

Maximum queue length (Longueur maxi) : nombre maximum d'entités observé.

Current queue wait (Attente en cours) : attente de la dernière entité ayant quitté la file.

Average queue wait (Attente moyenne) : moyenne des durées d'attente.

Maximum queue wait (Attente maxi) : durée maximale passée par une entité.

Arrivals (Arrivées) : nombre d'entités entrées dans la file.

Departures (Départs) : nombre d'entités ayant quitté la file.

Renegs (Abandons) : nombre d'entités ayant abandonné la file.

Utilization : pourcentage du temps où le bloc a été utilisé sur l'ensemble de la simulation. Intègre le temps passé par les entités dans la file parce qu'aucun bloc aval n'était capable de les absorber.

Total cost (Coût total) : le coût total d'attente de toutes les entités passées par la file.

Onglet Contents

Cet onglet existe pour les blocs Queue, Queue Equation, Resource Item et Activity, pour permettre de connaître soit le contenu du bloc à un instant donné, soit l'historique de toutes les entités qui ont traversé le bloc. Notez que cette fonction ralentit l'exécution et charge la mémoire : elle est conseillée pour la phase de débogage. Un onglet similaire existe pour le bloc Executive, centralisant tous les onglets Contents du modèle.

Enable current contents (activer le suivi du contenu) : en cochant cette case vous activez les options de l'onglet.

Table display (la table montre) : le menu permet de choisir parmi les des modes de fonctionnement de l'onglet.

Current contents mode (contenu actuel) : dans ce mode la table affiche les infos sur les blocs qui résident actuellement dans le bloc. Dans ce mode, les 5 premières colonnes de la table sont fixes. Par le bouton de redimensionnement "+/-" vous pouvez ajouter des colonnes et ainsi suivre d'autres propriétés. Colonne 1, Heure entrée, indique à quelle heure l'entité est entrée dans le bloc. Colonne 2, Arrêt, indique si l'entité est en arrêt (0 si non, 1 si oui). Colonne 3, Horaire, indique si l'entité est en horaire actif si 1, en horaire off si 0. Colonne 4, Durée arrêt, indique le temps total passé en arrêt ou en horaire off. Colonne 5, Heure fin, indique l'heure de fin du traitement. Si cette colonne est vide, c'est que l'entité est encore en traitement. Si une valeur apparaît, c'est que l'entité n'a pas pu quitter le bloc, ce qu'elle cherche à faire quand elle est terminée. Une entité non bloquée ayant fini son traitement ne figure normalement plus dans la table.

Historical log mode (historique) : dans ce mode la table affiche toutes les entités qui sont passées par le bloc durant la simulation. Cela signifie qu'une ligne est consacrée à chaque entité, et que la table peut devenir très grande. Dans ce mode aussi, les 5 premières colonnes de la table sont fixes. Les colonnes Heure entrée, Durée arrêt et Heure fin sont identique à celles du mode *Current contents*. Colonne 4, Heure sortie, affiche l'heure à laquelle l'entité a quitté le bloc. Si une entité a une valeur dans la colonne Heure fin mais pas dans heure sortie, c'est qu'elle est bloquée dans le bloc. Colonne 5, Sortie, décrit la manière dont l'entité a quitté le bloc ; "Normal" signifie par le connecteur habituel itemOut, "Préemption" signifie que l'entité a été préemptée.

Display contents (afficher contenu) : en mode *Current contents*, un menu apparaît sous la table, permettant de choisir quand afficher le contenu : 1.) durant la simulation, 2.) uniquement sur clic du bouton Instantané, ou 3.) à la toute fin de la simulation.

Control log time (bornes pour historique) : en mode *Historical*, cette case à cocher apparaîtra, permettant de définir une fenêtre de temps durant laquelle l'historique sera enregistré.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Animate H-block objects X to Y (Animer les objets de bloc H x jusque y) : anime la longueur de la file sur plusieurs objets d'animation présents sur le bloc hiérarchique englobant ce bloc. La plage est celle des numéros d'objets d'animation sur l'icône du bloc hiérarchique. Chaque objet dans cette plage animera une entité différente.

Connecteurs

En entrée l'entité discrète qui ressort après avoir attendu dans la file ou non.

Il y aura un connecteur de sortie continu dans la matrice des connecteurs de sortie pour chaque résultat d'équation associé à un connecteur.

Il y aura un connecteur d'entrée continu dans la matrice des connecteurs d'entrée pour chaque entrée d'équation associée à un connecteur.

R : si le bloc s'intègre au système des ressources complexes, indique index d'enregistrement dans la table des besoins du besoin de l'entité considérée.

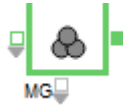
Connecteurs de sortie continus : (dans l'ordre d'apparition.)

L : (Longueur) Nombre d'entités dans le bloc.

W : (Attente : Wait) La durée d'attente de l'entité sortie en dernier du bloc.

F : (Plein : Full) Emet 1 si le bloc est plein, sinon émet 0.

Queue Matching



Généralités

Le bloc Queue Matching comporte un certain nombre de files d'attentes contenant des groupes d'entités différentes. Elle libère tout un groupe lorsqu'il y a la disponibilité en aval et que les conditions liées au groupe sont réunies. Elle sert à assembler ou appairer un type d'entité avec un autre.

Chaque connecteur d'entrée représente l'entrée dans une file interne du bloc. Chacune de ses files internes est elle-même divisée en "bacs" dans lesquels les entités entrantes sont rangées suivant la valeur d'un attribut. Cet attribut est commun à toutes les files internes et un groupe est donc la somme de tous les bacs des files internes définis par la même valeur de l'attribut en question.

Dans ce schéma il y a 5 files (5 connecteurs d'entrée), et 3 groupes (car l'attribut d'assemblage prend les 3 valeurs x, y et z). En tout 15 'bacs' d'entités distincts.

	Groupe 1 Valeur = x	Groupe 2 Valeur = y	Groupe 3 Valeur = z		Besoins
File 1	bac (G1,F1)				a
File 2					b
File 3		...			c
File 4					d
File 5			Bac(G3,F5)		e

Le besoin ('requirement') définit le nombre d'entités minimum dans chaque bac d'un groupe pour qu'un assortiment soit constitué.

Une entité d'un type particulier (x, y ou z) ne peut s'apparier qu'à une entité du même type (les entités du groupe 2 ne correspondront jamais à des entités du groupe 3). Quand les exigences en termes de quantité requise sont atteintes pour tous les bacs d'un même groupe (le besoin), une correspondance peut avoir lieu et les entités du groupe en question seront relâchées. Cf ci-dessous pour connaître plus en détail la façon dont les quantités requises sont traitées.

Quelques particularités de la Queue Matching :

- un assortiment ('match') dans un groupe ne peut être atteint tant que le minimum requis pour chaque bac composant le groupe n'a pas été atteint.
- le nombre de groupes peut être fixe ou variable. S'il est variable, il est défini durant la simulation quand les entités entrent dans le bloc avec différentes valeurs d'attribut de correspondance.
- si le nombre de groupes est fixé à 1, il n'est pas nécessaire de définir un attribut de correspondance. Sinon, c'est obligatoire et toute entité qui entre dans le bloc doit avoir une valeur pour cet attribut d'assemblage, sinon elle est inclassable.

- si l'option de mise en lot n'est pas activée, les files internes sont indépendantes : une entité ressortira par le connecteur équivalent à celui par lequel elle est entrée.

Exemple : assemblage d'une voiture

Le bloc peut ainsi représenter l'accumulation de toutes les pièces requises pour produire différentes voitures. Chaque type de pièce (essieux, pneus, moteurs, bloc air conditionné, etc.) est stocké dans sa propre file. Chaque type de voiture requiert des pièces particulières (le groupe), et un nombre spécifique de chacune de ces pièces (la taille de l'assortiment). Le groupe de pièces pour tel type de voiture ne sera libéré que lorsque toutes les pièces seront disponibles.

Le tableau suivant représente le contenu à l'instant t de cette Queue Matching :

	Economie	Sport
Pneus	4	2
Essieux	1	3
Moteurs	1	4
A/C	3	3

Dans cet exemple, chaque catégorie de pièce pour voiture (Essieu, pneu, moteur, A/C) a sa propre file, où l'on distingue les entités pour voitures économiques et pour voitures de sport. Lorsqu'un moteur arrive sur la file Moteurs, elle sera placée dans le bac Economie ou Sport en fonction de son attribut d'assemblage. Lorsque le modèle tourne, les pièces sont stockées dans la file en attendant qu'un lot complet soit disponible. Dans l'exemple, le groupe Economie comporte 1 essieu, 4 pneus, 1 moteur et 3 unités d'air conditionné. Il manque un essieu, et quand il arrivera, les pièces suivantes quitteront la file : 2 essieux, 4 pneus, 1 moteur, 1 A/C.

Que se passe-t-il si les deux types de voitures ont des pièces en commun ? Par exemple elles ont la même antenne radio. On pourrait imaginer une autre file dans le bloc, pour les antennes, mais dans quel groupe mettre les entités antennes qui arrivent ? La solution est de déclarer un "Groupe universel" dont les entités sont disponibles pour TOUS les groupes. Une valeur d'attribut particulière distingue les entités de ce groupe.

	Economie	Sport	Universel
Pneus	4	3	
Essieux	1	0	
Moteurs	1	2	
A/C	3	1	
Antennes	0	0	2

Notez que le nombre de files dans le groupe universel n'est pas limité. On pourrait décider que les unités d'air conditionné sont les mêmes pour les deux types de voitures. L'unique modification au modèle, dans ce cas, serait de donner aux entités A/C la valeur du groupe universel.

Onglet Match items

Match Items | Batch | Contents | Results | Item Animation | Block Animation | Comments

Sorts items into Groups by match attribute value. Items from a particular Group are held until the requirements for each Bin in that Group have been met.

Set matching behavior

Table view: requirements

	Queue Names	Max Q Length	Requirement Options	Required Items
1	Q1	Infinity	Constant	1

Grouping options

The number of groups is fixed at 1 groups.

Requirements options

When requirement is met: release only matched items from Group

Store effective requirement as an attribute on released items: None

Queueing options

Sort by: First in, first out Show L (length) connectors Calculate waiting costs

Block type: Residence Current Group:

Le nombre de files d'attente que contiendra le bloc est défini par le nombre de lignes dans la table, ou plus facilement en déployant le connecteur d'entrée du bloc.

Table

Queue names (*Noms des files*) : vous avez la possibilité de donner un nom descriptif à chaque file (ligne).

Max Q Length (*Longueur maxi*) : spécifie la capacité maximum de chaque file. Par défaut, la capacité est infinie.

Requirements Options (*options des Besoins*) : Vous choisissez la "méthode" pour connaître le besoin en entité pour chaque file.

Constant : vous entrez le nombre souhaité. Connector : le nombre est la valeur reçue sur le connecteur d'entrée continu qui se colle au connecteur d'entrée discret pour chaque file. Attribute : c'est un attribut de l'entité qui arrive qui indiquera le nombre d'entité souhaité. Vous devez donner le nom de l'attribut et la valeur par défaut en cas d'absence (0 ou non défini).

Required items (*Nbre d'entités*) : le nombre d'entités requis pour former un assortiment (nombre défini par l'une des trois méthodes de la colonne précédente).

Grouping options (Options d'assemblage)

S'il n'y a qu'un groupe, il n'est pas besoin d'attribut d'assemblage ni d'ID de groupe. Mais vous aurez souvent plus d'un groupe, et devrez définir un attribut d'assemblage ainsi qu'un ID de groupe dans l'entête de chaque colonne de groupe.

The number of groups is fixed (Nombre de groupes fixe) : saisissez le nombre de groupe (le nombre de colonnes correspondantes s'ajoute à la table). Choisissez immédiatement l'attribut d'assemblage, et la valeur qui le caractérise pour chaque groupe. Le texte en rouge vous guidera.

Grouping options

The number of groups is variable Show each group's current requirement
Slot items into groups using the attribute: pom (required)
 Declare 'Common Group' match attribute value

The number of groups is variable (Nombre de groupes variable) : deux raisons peuvent faire choisir l'option 'variable' : vous ne savez pas combien de différents types d'entités entreront dans le bloc, ou bien le nombre de groupe est très important et vous voulez éviter de spécifier l'ID de groupe pour chaque groupe.

Sort items into groups using the attribute XX (Définir des groupes par l'attribut XX d'assemblage) : il s'agit de l'attribut servant à l'assemblage des entités. Il est indispensable d'en sélectionner un. Les valeurs de cet attribut servent à définir les groupes.

Declare 'Common Group' match attribute value (Déclarer une valeur d'attribut 'Groupe universel') : en cochant cette case vous définissez un groupe commun où les entités qui possèdent cette valeur d'attribut pourront servir pour un assortiment avec tel ou tel groupe (voir l'exemple des voitures ci-dessus).

Requirements options (Options des besoins)

Les options proposées dépendent de la méthode choisie pour les besoins.

Connector requirements (Les besoins par connecteur) : **use current connector value** (utilisent la valeur courante du connecteur) : avec cette option, chaque fois qu'un message est reçu sur un connecteur de besoin, les valeurs courantes des connecteurs sont utilisées pour tenter de faire un assemblage ; **use value at connector when first item enters bin** (utilisent la valeur du connecteur quand la première entité arrive : la valeur du connecteur de besoin correspondant lorsque qu'arrive la première entité dans un bac vide est enregistrée et utilisée pour l'assemblage. **When last item enters bin** (quand la dernière entité arrive) : la valeur du connecteur de besoin correspondant lorsque qu'arrive chaque entité (la dernière) dans un bac est enregistrée et utilisée pour l'assemblage.

Attribute requirements are set when when first item enters bin (Besoins par attribut définis quand la première entité arrive) : la valeur l'attribut de besoin enregistrée et utilisée est celle de la première entité arrivant dans un bac vide. **When last item enters bin** (quand la dernière entité arrive) : la valeur l'attribut de besoin enregistrée et utilisée est celle de l'entité courante (la dernière).

When requirement is met (Lorsque l'assortiment est prêt) : permet de contrôler ce qui est libéré lorsque les conditions d'assemblage sont réunies. Par défaut le bloc va libérer uniquement les entités formant un assortiment, mais vous pouvez aussi libérer toutes les entités du groupe ('release all items in matched Group'), même celles stockées à l'avance et n'ayant pas trouvé un assortiment complet.

Store effective requirement as an attribute on released items (Stocker les besoins dans un attribut des entités sortantes) : pour chaque entité sortant du bloc, cet attribut recevra pour valeur la taille réelle de l'assemblage.

Queuing options (Options des files)

Sort by (Trier par) : sélectionnez l'algorithme de tri de la file. Les choix possibles sont : Valeur d'attribut, FIFO, LIFO et Priorité. Dans le cas de l'attribut, vous devez préciser l'attribut concerné, et l'ordre du tri (ascendant ou descendant).

Show L (length) connectors (Voir les connecteurs L (longueur)) : si cochée, les connecteurs L correspondant aux files contenues dans le bloc. Des options complémentaires s'affichent :

L reports (L indique) : permet de contrôler ce qu'indiqueront les connecteurs L pour chaque file. Les options sont : batched items (entités avec correspondance); unbatched items (entités sans correspondance) et unbatched, by Group (sans correspondance, par groupe). Elles spécifient quelles entités présentes dans les files seront comptées. La dernière option vous demande d'entrer la valeur du groupe que vous souhaitez suivre par L.

Show values on L connectors (Voir valeurs sur les connecteurs L) : affiche les valeurs des connecteurs L sur l'icône du bloc.

Calculate waiting costs (Calculer les coûts d'attente) : calcule les informations de coût pour chaque file.

Onglet Batch

Si la case "**Enable batching**" (Activer la mise en lot) est cochée alors une seule entité ressortira du bloc quel que soit le nombre de files internes qui auront été définies. De plus, TOUTES les entités faisant partie de l'assortiment seront rassemblées en une seule et unique entité. La vue de l'icône du bloc s'adapte à cette case à cocher.

Dans l'exemple de la voiture, au lieu d'avoir en sortie 8 entités (2 essieux, 4 pneus, 1 moteur, 1 A/C), il n'y en aura qu'une seule représentant la voiture complète.

Le tableau "**Item property transfer protocol**" (Transfert des propriétés des entités) permet de gérer le transfert des valeurs d'attribut depuis les entités initiales vers l'entité composée.

Voir le bloc Batch pour la notion "Preserve uniqueness".

Onglet Contents

Cet onglet existe pour les blocs Queue, Queue Equation, Resource Item et Activity, pour permettre de connaître soit le contenu du bloc à un instant donné, soit l'historique de toutes les entités qui ont traversé le bloc. Notez que cette fonction ralentit l'exécution et charge la mémoire : elle est conseillée pour la phase de débogage. Un onglet similaire existe pour le bloc Executive, centralisant tous les onglets Contents du modèle. Pour le bloc Queue Matching, l'onglet est spécifique.

Track item in each queue (activer le suivi du contenu) : en cochant cette case vous activez les options de l'onglet.

Table display (la table montre) : le menu permet de choisir ce qu'affiche la table. Les options sont : entités sans correspondance dans chaque ou le total des arrivées pour chaque file.

User filter (Filtrer) : en cochant cette case vous pouvez entrer des bornes pour les valeurs de l'attribut d'assemblage pour limiter les entités montrées dans la table.

Blocked batches (Lots bloqués) : si la mise en lots est activée, ce champ indique combien d'entités mises en lot sont prêtes à quitter le bloc mais ne le peuvent pas à cause d'un blocage en aval.

Rotate table 90 degrees (Pivoter la table de 90 degrés) : fait pivoter les données de la table. Ce changement de format est nécessaire si vous voulez associer ces données à une BDD par des options ci-dessous.

At start of simulation, create new DB table and write item information to it (En début de simulation, créer une nouvelle table de BDD et y écrire ces informations) : spécifie que les données de la table doivent être écrites dans une table.

Database name (Nom BDD) : nom de la base données cible.

Table name (Nom table) : nom de la table cible.

Onglet Results

Colonne 1 (Longueur moyenne) : nombre moyen d'entités dans la file.

Colonne 2 (Longueur maxi) : nombre maximum d'entité ayant séjourné dans la file.

Colonne 3 (Attente moyenne) : durée moyenne passée dans la file.

Colonne 4 (Attente maxi) : durée maximum passée dans la file.

Colonne 5 (Utilisation) : pourcentage du temps où le bloc a été utilisé sur l'ensemble de la simulation. Intègre le temps passé par les entités dans la file parce qu'aucun bloc aval n'était capable de les absorber.

Colonne 6 (Coût total) : le coût total d'attente de toutes les entités passées par la file.

Onglet Item Animation

Cet onglet ne présente pas de choix global. Une table permet de spécifier les choix pour chaque file interne :

Colonne 1 (Animation Option) : spécifie le choix d'animation pour chaque file interne. **No Change** "Tel quel" pour garder l'objet initial et **Change to** "changer en" pour pouvoir spécifier de nouveaux choix.

Colonne 2 (2D picture) : spécifie le symbole 2D associé à cette valeur.

Connecteurs

Chaque file interne a son connecteur d'entrée discret.

Les entités sortent par leur connecteur de sortie lorsque les conditions d'assemblage ont été réunies.

Les connecteurs continus près des connecteurs d'entrée discrets sont les connecteurs pour les besoins, *MatchRequirementIn*. Voir ci-dessus *Options des besoins*.

Les connecteurs continus près des connecteurs de sortie discrets sont les connecteurs L (Longueur). Voir ci-dessus *L reports*.

Connecteurs d'entrée continus : (dans l'ordre d'apparition.)

MG : correspondance uniquement sur le groupe indiqué. Si ce connecteur est relié, le bloc ne cherchera à faire de correspondance que sur le groupe spécifié par ce connecteur. A la différence du connecteur *Demand*, un message reçu sur MG peut provoquer plusieurs correspondances pour le groupe indiqué. Notez que ce connecteur ne modifie pas le nombre d'entités requises dans une file pour effectuer une correspondance, mais contrôle l'assemblage que le bloc tente de constituer.

DG : vider les entités d'un seul groupe. La valeur sur ce connecteur définit le groupe à vider. Notez que c'est au moment où un message est reçu sur le connecteur MG que le bloc vide toutes les entités du groupe.

DA : vider les entités de tous les groupes. Une valeur TRUE (> 0.5) déclenche de vidage de tous les groupes.

LO : L indique la longueur des groupes. La valeur de cette entrée prévaut sur la valeur de groupe indiquée par l'élément de dialogue *L reports*.

Order : si relié, le bloc n'essaie d'assembler les entités que lorsqu'il reçoit un message sur ce connecteur. De plus, la valeur sur le connecteur définit le groupe que le bloc essaie d'assembler. Donc ainsi, le bloc retient toutes les entités jusqu'à ce qu'arrive un message de *Order*, et le bloc cherche alors à servir le groupe indiqué par la valeur sur *Order*. La tentative d'assemblage n'est faite qu'une fois par message.

Connecteurs de sortie continus : (dans l'ordre d'apparition.) :

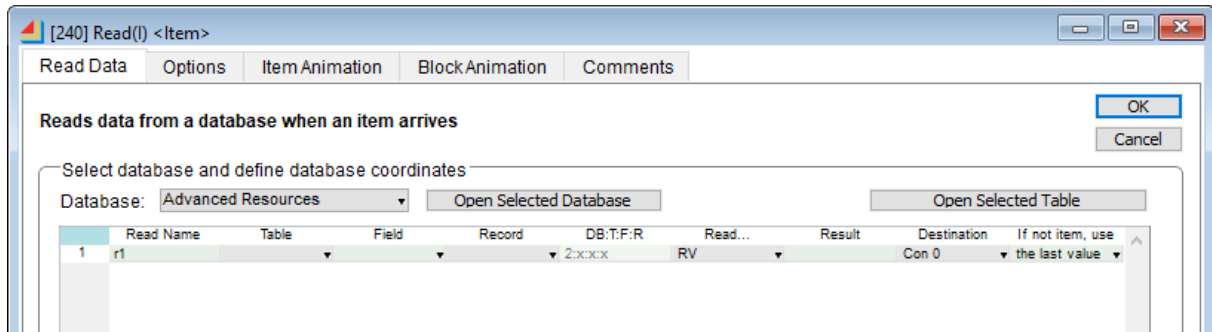
B : émet l'index de BDD de la base cible.

T : émet l'index de table de la table cible.

Read(I)



Lit des données dans la BDD à l'arrivée d'une entité.



Vous définissez un certain nombre de lectures à effectuer au passage de l'entité, chaque lecture remplissant une ligne de la table de données. Dans la table vous définissez où lire les données et s'il faut les stocker dans un attribut.

Des connecteurs d'entrée peuvent spécifier le nom de chaque lecture, ou encore les valeurs à lire.

Vous devez sélectionner une base de données dans le dialogue, ou par un connecteur qui change en cours de simulation.

Des connecteurs apparaîtront si l'option **Use a connector** (Utiliser un connecteur) est sélectionnée dans les colonnes Table, Field (Champ) ou Record (Enregistrement). Si les connecteurs existent, ils doivent obligatoirement être reliés.

Le nombre de lignes dans la table peut être modifié en cliquant dans la case +/- dans le coin inférieur droit de la table. Vous pouvez changer le nombre de lignes, copier ou supprimer les lignes sélectionnées.

Onglet Read Data

Database (BDD) : sélectionne la base de données à référencer.

Open selected database (Ouvrir cette BDD) : ouvre la BDD en mode Edition.

Open selected table (Ouvrir cette table) : ouvre le mode Edition pour la table sélectionnée.

Tableau

Colonne 1 Read name (Nom) : donne un nom aux données lues par cette ligne. L'utilisateur pourra ainsi savoir de quoi il s'agit, et l'information pourra aussi être affichée sur les connecteurs du bloc.

Colonne 2 Table : sert à référencer la table. La cellule contient un menu déroulant avec les options suivantes : **Select from list** (Dans une liste), **Type an index** (Entrer un index), **Use a connector** (Utiliser un connecteur), **Get from attribute** (Utiliser un attribut) et **Use previous result** (Utiliser résultat précédent). **Select from list** affiche une liste de tables pour y faire un choix. **Type an index** permet de saisir une valeur d'index. **Use a connector** affecte un connecteur d'entrée continu à cette ligne afin que les valeurs de table soient lues sur un connecteur. **Get from attribute** permet de spécifier un attribut dans lequel lire la valeur. **Use previous result** permet, à partir de la ligne 2, d'utiliser le résultat de la ligne précédente pour définir l'index de table.

Colonne 3 Field (Champ) : permet de choisir le champ cible dans un menu. Les options du menu sont similaires à celle de sélection de table.

Colonne 4 Record (Enreg.) : ces options du menu sont similaires à celle de sélection de table, avec quatre options supplémentaires. **Use run number** (Numéro simu) indique que l'index d'enregistrement doit être le numéro de simulation + 1. S'emploie dans le cas de simulations multiples. **Item count** (Comptage entités) utilise le nombre d'entités qui ont passé par le bloc pour index d'enregistrement. **Record search using attribute** (Utiliser un attribut) propose de rechercher un enregistrement dont le contenu corresponde à celui d'un attribut de l'entité de passage. Le premier enregistrement qui correspond sera retenu. **Record search using connector** (Utiliser un connecteur) fait de même, mais à partir de la valeur lue sur un connecteur.

Colonne 5 (DB :T :F :R) : affiche l'Adresse BDD référencée par cette ligne. Si un "x" est affiché à un emplacement de l'adresse, cela signifie que cet élément de l'adresse est variable.

Colonne 6 Read (Lire...) : change en fonction des choix des colonnes précédentes. Les choix sont les suivants : **record value RV** (Valeur d'enregistrement), **parent record value PRV** (Valeur d'enr. Parent), **parent record index PRI** (Index d'enr. Parent) et **oldest parent record index PRIO** (Index d'enr. +vieux parent). **record value** va simplement prendre la valeur contenue dans l'enregistrement, et implique que le champ ne soit pas enfant. Si le champ cible est un champ enfant, 2 choix sont possibles (qui reflètent la structure des champs enfants qui contiennent en fait deux informations : la valeur prise chez le parent, et l'index de l'enregistrement d'origine chez ce parent) : **parent record value** prendra la valeur telle que définie dans l'enregistrement parent (c'est l'équivalent du RV). **parent record index** va lire chez l'enfant l'index de l'enregistrement, qui doit correspondre à un enregistrement parent (≥ 1 et \leq au nb d'enregistrement de la table parent). **oldest parent record index** remontera la relation parent-enfant jusqu'à l'ancêtre racine.

Colonne 7 Result (Résultat) : affiche la valeur effectivement lue par le bloc.

Colonne 8 Destination : permet de spécifier une propriété ou un connecteur où placer la valeur lue.

Colonne 9 If no item, use (0 entité, prendre) : spécifie quelle valeur prendre si aucune entité n'est présente. Les choix sont les suivants : **The last value** (dernière valeur), **novalue** et **la valeur=**. **last value** retourne la dernière valeur correcte. **novalue** retourne une noValue. Vous pouvez également spécifier un nombre.

Onglet Options

Initialize results at start of run to (résultats initialisés à) : spécifie quelle valeur doivent prendre les variables en sortie en début de simulation (avant l'arrivée de la première entité).

Show input connector labels (labels connecteurs d'entrée) : si la case est cochée, tous les connecteurs d'entrée afficheront leur label.

Show input connector values (valeurs connecteurs d'entrée) : si la case est cochée, tous les connecteurs d'entrée afficheront leur valeur.

Show output connector labels (labels connecteurs de sortie) : si la case est cochée, tous les connecteurs de sortie afficheront leur label.

Show output connector values (valeurs connecteurs de sortie) : si la case est cochée, tous les connecteurs de sortie afficheront leur valeur.

If reading from a Date Time field, read values reported in (Si lecture d'un champ Date Heure, donnée en sortie est dans l'unité de temps) : une donnée de type date sera transformée dans l'unité de temps de la simulation, ou gardée dans un format de date calendaire.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Connecteurs

En entrée l'entité qui déclenche la lecture dans la base de données.

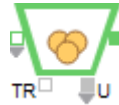
En sortie la même entité, avec éventuellement une propriété modifiée.

DB : spécifie la BDD à référencer. Prévaut sur les choix du dialogue.

Connecteur d'entrée continu : les lignes de la table qui font appel à un connecteur pour le choix de la table/champ devront être reliées. Également, les données à écrire dans la BDD si elles doivent être spécifiées par un connecteur.

Connecteur de sortie continu : les résultats lus peuvent être émis sur des connecteurs.

Resource Item



Ce bloc contient et fournit des entités (automobiles, commandes, opérateurs, etc.) au modèle. Il peut intervenir dans des systèmes ouverts ou fermés.

Property Name	Set Value	Table	Field	Record
1 None		N/A	N/A	N/A

A la différence du générateur, le bloc ne pousse pas ses entités. Si vous l'utilisez pour fournir des entités au modèle, assurez-vous d'avoir un nombre initial suffisant pour alimenter le modèle durant toute la simulation.

Le bloc ressemble à une file d'attente, dans laquelle le bloc en aval puise des entités (tant que son contenu reste positif).

Le connecteur TR (Total ressources) permet de modifier le nombre d'entités disponibles (s'emploie généralement dans un système fermé). Le nombre envoyé sur le connecteur TR spécifie ce que devrait être le nombre d'entités dans la ressource. Si ce nombre est supérieur au nombre antérieur, de nouvelles entités seront créées ; si ce nombre est inférieur, des entités seront supprimées.

Vous pouvez ajouter des propriétés aux entités traversant le bloc, ou leur retirer des attributs. Si le bloc est intégré dans une boucle fermée et si vous supprimez les propriétés des entités entrantes, les propriétés par défaut deviennent les propriétés de l'entité. Si vous ne supprimez pas les propriétés des entités (cas d'une boucle ouverte), les propriétés par défaut sont simplement ajoutées aux propriétés de l'entité.

Onglet Items

Initial number of items (Nombre initial d'entités) : nombre initial d'entités dans la ressource.

Database (optional) (Base de données -facultatif) : spécifie la BDD référencées par l'attribut DB Address. N'a besoin d'être défini que pour les propriétés liées à la base de données.

Strip properties from any incoming items (closed system) (Effacer propriétés des entités entrantes - systèmes fermés) : remplace les propriétés présentes sur les entités.

Use Shift (Utiliser l'horaire) : sélectionnez l'horaire s'appliquant à ce bloc. Un horaire peut être soit du type ON/OFF, soit indiquer un nombre maximum de d'entités dans le bloc.

Tableau

Colonne 1 Property name (Propriété) : La propriété donnée peut être autonome ou liée à la base de données. Sélectionnez le type de propriété que vous voulez donner à l'entité.

Colonne 2 Set value (Valeur) : Spécifiez la valeur de la propriété dans cette colonne.

Colonne 3 (Table) : spécifie la table associée à cette propriété. Cette colonne et les deux suivantes n'ont besoin d'être définies que pour les propriétés liées à la base de données.

Colonne 4 Field (Champ) : spécifie le champ associé à cette propriété.

Colonne 5 Record (Enreg.) : spécifie l'enregistrement associé à cette propriété.

Onglet Cost

Define Activity Based Costs (Définit le coût des entités) : active les éléments de dialogue liés aux coûts.

Provides items that calculate costing as (Les entités sont des) : cet élément de dialogue ne s'applique que si le modèle calcule des coûts automatiquement. (Pour cela, un paramètre de coût en entrée doit être supérieur à zéro, pour au moins un bloc dans le modèle.) Si vous choisissez **accumulators** (accumulateurs de coût), chaque entité créée ou passant par le bloc mémorisera les coûts accumulés dans le modèle dans l'attribut "_cost". Si vous choisissez **resources** (ressources), chaque entité mémorisera des taux "par unité de temps" et "par entité" (voir plus bas). Lorsque la ressource est mise en lot avec une entité, cette information servira à calculer le coût associé à l'utilisation de la ressource, coût ajouté à l'attribut "_cost" de l'entité.

Waiting cost per time unit (Coût d'attente par unité de tps) : le coût par unité de temps pour l'utilisation de cette ressource. L'unité de temps utilisée pour définir ce taux doit être cohérente avec l'unité de temps utilisée dans le modèle. Tant que la ressource est associée à une entité (mise en lot), l'entité accumule de coût correspondant. Le coût ajouté à l'attribut "_cost" de l'entité concernée. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, une unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Cost per use (Coût à l'utilisation) : coût ajouté à l'attribut "_cost" de chaque entité mise en lot avec une entité provenant du bloc Resource Item.

Total Cost (Coût total) : Si les entités sont des ressources, le coût total est le coût de toutes les ressources utilisées durant la simulation. Si les objets sont des accumulateurs de coûts, le coût total représente les coûts associés à la création de l'entité (coût à l'utilisation) plus les coûts associés aux temps d'attente (coût par unité de temps (attente)).

Onglet Results

Available Items (Entités disponibles) : nombre d'entités présentes dans la ressource.

Items in us (Entités utilisées) : nombre d'entités issues de ce bloc utilisées dans le modèle. C'est le nombre d'entités ayant quitté le bloc, moins le nombre y étant entré.

Total items (Total entités) : nombre total dans la ressource. Dans un système fermé, ce devrait être la somme de **Available Items** et de **Items in us**, à moins que des entités doivent encore être affectées par la valeur reçue sur le connecteur **TR**.

Utilisation : le pourcentage de temps d'utilisation des entités (hors du bloc) sur le temps total simulé. N'a de signification que dans un système fermé.

Total cost (Coût total) : voir onglet Cost.

Onglet Contents

Cet onglet existe pour les blocs Queue, Queue Equation, Resource Item et Activity, pour permettre de connaître soit le contenu du bloc à un instant donné, soit l'historique de toutes les entités qui ont traversé le bloc. Notez que cette fonction ralentit l'exécution et charge la mémoire : elle est conseillée pour la phase de débogage. Un onglet similaire existe pour le bloc Executive, centralisant tous les onglets Contents du modèle.

Enable current contents (activer le suivi du contenu) : en cochant cette case vous activez les options de l'onglet.

Table display (la table montre) : le menu permet de choisir parmi les des modes de fonctionnement de l'onglet.

Current contents mode (contenu actuel) : dans ce mode la table affiche les infos sur les blocs qui résident actuellement dans le bloc. Dans ce mode, les 5 premières colonnes de la table sont fixes. Par le bouton de redimensionnement "+/-" vous pouvez ajouter des colonnes et ainsi suivre d'autres propriétés. Colonne 1, Heure entrée, indique à quelle heure l'entité est entrée dans le bloc. Colonne 2, Arrêt, indique si l'entité est en arrêt (0 si non, 1 si oui). Colonne 3, Horaire, indique si l'entité est en horaire actif si 1, en horaire off si 0. Colonne 4, Durée arrêt, indique le temps total passé en arrêt ou en horaire off. Colonne 5, Heure fin, indique l'heure de fin du traitement. Si cette colonne est vide, c'est que l'entité est encore en traitement. Si une valeur apparaît, c'est que l'entité n'a pas pu quitter le bloc, ce qu'elle cherche à faire quand elle est terminée. Une entité non bloquée ayant fini son traitement ne figure normalement plus dans la table.

Historical log mode (historique) : dans ce mode la table affiche toutes les entités qui sont passées par le bloc durant la simulation. Cela signifie qu'une ligne est consacrée à chaque entité, et que la table peut devenir très grande. Dans ce mode aussi, les 5 premières colonnes de la table sont fixes. Les colonnes Heure entrée, Durée arrêt et Heure fin sont identique à celles du mode *Current contents*. Colonne 4, Heure sortie, affiche l'heure à laquelle l'entité a quitté le bloc. Si une entité a une valeur dans la colonne Heure fin mais pas dans heure sortie, c'est qu'elle est bloquée dans le bloc. Colonne 5, Sortie, décrit la manière dont l'entité a quitté le bloc. "Normal" signifie par le connecteur habituel itemOut, "Préemption" signifie que l'entité a été préemptée.

Display contents (afficher contenu) : en mode *Current contents*, un menu apparaît sous la table, permettant de choisir quand afficher le contenu : 1.) durant la simulation, 2.) uniquement sur clic du bouton Instantané, ou 3.) à la toute fin de la simulation.

Control log time (bornes pour historique) : en mode *Historical*, cette case à cocher apparaîtra, permettant de définir une fenêtre de temps durant laquelle l'historique sera enregistré.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Animer les objets de bloc H x jusque y : anime la longueur de la file sur plusieurs objets d'animation présents sur le bloc hiérarchique englobant ce bloc. La plage est celle des numéros d'objets d'animation sur l'icône du bloc hiérarchique. Chaque objet dans cette plage animera une entité différente.

Connecteurs

En entrée des entités à mettre à disposition par la ressource.

En sortie les entités demandées par le modèle.

TR : (Total ressources) modifie le nombre d'entités dans le bloc. Par exemple si le nombre disponible précédent était 10, et que la valeur reçue est 9, si toutes les entités sont en utilisation, la prochaine qui revient dans le bloc sera supprimée.

Connecteurs de sortie continus : (dans l'ordre d'apparition.)

U : (Utilisation) taux d'utilisation du bloc - voir plus haut.

Avail (dispo) : nombre de pièces disponibles dans le bloc.

In use (en usage) : nombre de pièces en utilisation dans le modèle n'étant pas retournées dans le bloc.

Animation

L'icône indique quand une entité arrive ou part, et le nombre d'entité présentes dans le bloc.

Resource Manager

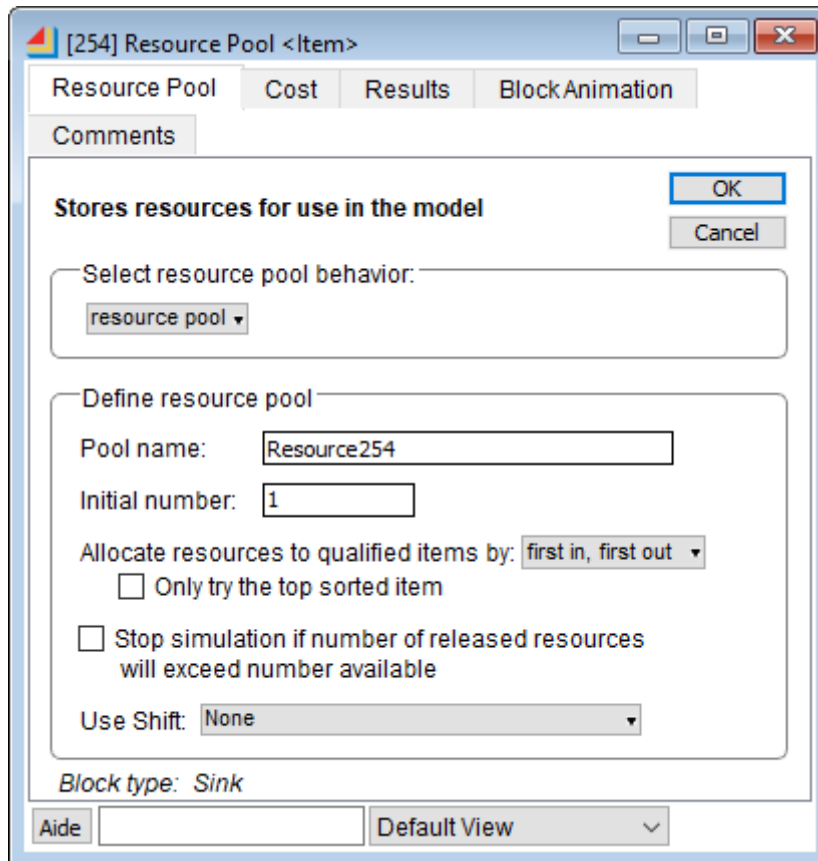


(documentation à part)

Resource Pool



Ce bloc est un "pool" de ressources utilisables dans une simulation. Les éléments de la ressource peuvent limiter la capacité d'une section d'un modèle. Par exemple, elles peuvent représenter un nombre limité de tables dans un restaurant, ou des outils dans un atelier.



Le comportement de base du bloc se définit dans le menu principal : Pool standard ou Pool ressources complexes.

A la différence du bloc Resource Item, les éléments d'un pool ne sont pas des entités. Il s'agit de variables qui indiquent l'aspect quantitatif d'un facteur limitant. Le bloc Resource Pool s'emploie avec le bloc *Queue* en mode Ressource pour allouer les éléments du pool à des entités, et avec le bloc *Resource Pool Release* pour libérer les éléments du pool. Des entités peuvent attendre pour les éléments d'un pool depuis n'importe quel nombre de blocs Queue. Le bloc Resource Pool détermine comment les éléments sont alloués aux entités : soit l'entité qui est arrivée la première dans le bloc Queue, soit l'entité ayant la plus haute priorité (plus faible valeur numérique de priorité). Si la case **Only try the top sorted item** (*Allouer uniquement à l'entité la mieux classée*) est cochée, une seule entité sera considérée pour l'allocation d'un élément du pool. Si elle n'est pas cochée, le pool passe en revue toutes les entités en attente jusqu'à trouver à la fois la première entité pouvant quitter un bloc Queue, et le nombre suffisant d'éléments du pool disponibles.

Le connecteur *c* change le nombre d'éléments disponibles dans le pool, en fonction de la valeur de l'entité envoyée. Une nouvelle valeur écrase l'ancienne valeur.

Onglet Resource Pool

Pool standard

Nom du pool : le nom de ce pool. Chaque pool doit avoir un nom unique.

Nombre initial : le nombre d'éléments initialement disponibles dans le pool.

Allouer le pool à... la première entité : cette option alloue l'élément à l'entité qui l'attend depuis le plus longtemps.

Allouer le pool à... l'entité de plus haute priorité : cette option alloue l'élément à l'entité qui la plus haute priorité (plus faible valeur numérique de priorité).

Allouer uniquement à l'entité la mieux classée : lorsqu'il essaie d'allouer un élément, le pool ne considérera que l'entité de meilleur rang. Si l'option n'est pas cochée, le pool passe en revue toutes les entités en attente jusqu'à pouvoir allouer un élément. Ne pas cocher cette option si des entités attendent plus d'un élément de ressource à la fois.

Stopper la simulation si le nombre d'éléments libérés dépasse le nombre disponible : la simulation s'arrêtera si le nombre d'éléments dans la ressource devient supérieur au nombre d'éléments disponibles. Cette "génération d'éléments" peut arriver si les éléments sont libérés plus souvent qu'ils ne sont alloués, ou si le nombre libéré ne correspond pas au nombre alloué.

Utiliser un horaire : sélectionnez l'horaire s'appliquant à ce bloc (de type ON/OFF ou indiquant un nombre maximum d'éléments)

Onglet Cost

Define resource pool costs (Définir des coûts aux éléments de pool) : active les options de coût du bloc.

Waiting cost per time unit (Coût d'attente par unité de tps) : Le coût par unité de temps pour l'utilisation de cette ressource. L'unité de temps définissant ce coût doit être cohérente avec les unités de temps utilisées dans le modèle. Tant que cet élément de ressource est associé à une entité, l'entité accumule le coût indiqué. Le coût est ajouté à l'attribut "_cost" de l'entité. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, cette unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Cost per use (Coût à l'utilisation) : coût ajouté à l'attribut "_cost" de chaque entité associée à un élément de ce pool.

Total cost (Coût total) : le coût total des ressources utilisées durant la simulation.

Onglet Results

Ressources :

Total ressources : le nombre courant d'éléments du pool. Comprend à la fois le nombre disponible et le nombre en utilisation.

En utilisation : le nombre d'éléments du pool actuellement en usage dans le modèle

Disponibles : le nombre d'éléments du pool actuellement disponibles.

Utilisation : le pourcentage de temps d'utilisation des éléments du pool (hors du bloc) sur le temps total simulé.

Entités en attente :

Actuellement : le nombre actuel d'entités attendant pour ce pool.

Nombre moyen : le nombre moyen d'entités attendant pour ce pool.

Attente moyenne : la quantité moyenne de temps attendu par des entités pour ce pool.

Connecteurs

c : modifie le nombre d'éléments dans le pool en fonction de la valeur reçue, qui prévaut sur l'ancienne valeur.

Connecteurs de sortie continus : (dans l'ordre d'apparition.)

U (Utilisation) : taux d'utilisation du bloc.

L (Longueur) : nombre d'éléments disponibles.

use : le nombre d'éléments actuellement en usage.

waiting : le nombre d'entités qui attendent pour un élément de ce pool.

Animation

L'icône montre le nom du pool ainsi que le nombre d'éléments disponibles.

Resource Pool Release



Libère des éléments d'un pool de ressource au passage d'une entité.

Il y a deux manières de choisir le pool :

- Par l'option "Libérer par nom", en entrant explicitement le nom du pool et de nombre d'éléments à libérer.
- Par l'option "Libérer par attribut", en spécifiant l'attribut qui a été défini dans le bloc Queue en mode Ressource.

Les éléments sont immédiatement libérés vers leur Pool, qui vérifie aussitôt sa liste d'entités en attente pour essayer de leur allouer les éléments libérés.

Note : L'ordre dans lequel une entité passe par des blocs *Resource Pool Release* détermine l'ordre dans lequel les blocs Pool sont activés pour fournir des ressources à des entités en attente. Dans les cas où l'allocation des ressources est complexe (plusieurs pools alloués simultanément et différentes entités demandant différentes ressources) nous conseillons de libérer en premier le pool qui est le plus demandé. Cela permettra que cette ressource soit allouée avant d'autres pools.

Note 2 : l'attribut qui spécifie le pool est composé. L'attribut lui-même indique le nombre d'éléments du pool à libérer, tandis que l'attribut préfixé d'un souligné (__) spécifie le numéro du bloc pool.

Onglet Release

Select Release behavior (Choisit le mode de fonctionnement) : le comportement de base du bloc est de libérer des ressources simples issues de Pools (**Release Pool resources**). Si un bloc Resource Manager est présent, il peut aussi libérer des ressources « complexes » - voir le bloc Resource Manager.

Choisit les éléments libérés et leur quantité

Dans ce mode, le passage d'une entité libère automatiquement les ressources qui retournent dans leur pool d'origine.

Release by (Libérer par) : le choix **name (nom)** spécifie que les éléments du pool de ressource dont on donne le nom seront libérés lorsqu'une entité passera par le bloc. Le choix **Attribute** spécifie que l'attribut dont on donne le nom spécifie quel pool de ressource sera libéré au passage d'une entité.

If resource pool names are duplicated, find only closest (Si noms de Pools en double, prendre le plus proche) : cochez cette case pour éviter les doublons. Si deux Pools (ou plus) ont le même nom, seul le plus proche dans la hiérarchie sera proposé. Si les deux Pools sont au même niveau hiérarchique, ils seront ignorés. Cette option sert à limiter le nombre de blocs considérés, notamment dans des blocs hiérarchiques dupliqués ou issus d'une bibliothèque.

Release quantity (Quantité libérée) : par **Attribut** : l'attribut indiqué servira à déterminer le nombre d'éléments du pool libérer. Par **Fixed numbers** (Nombres fixes) spécifie le nombre d'éléments de pool qui seront libérés lorsqu'une entité passera par le bloc.

Erase attribute after release (Effacer cet attribut après libération) : si le nom du pool ou la quantité est identifié par un attribut, cocher cette option efface cet attribut.

Show Resource Pool block names on icon (Voir les noms des blocs Pool sur l'icône) : les noms des blocs Resource Pool pour lesquels une entité attend des éléments.

Show Resource Pool block numbers in dialog (Voir les numéros des blocs Pool dans le dialogue) : si cochée, le numéro du bloc Resource Pool sera affiché avec son nom dans le menu déroulant.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Connecteurs

L'entité qui entre est celle qui déclenche la libération d'un élément de pool.

L'entité qui sort est la même, sans altération.

Connecteur d'entrée continu (NumberToReleaseIn) : prévaut sur l'élément de dialogue **Release quantity** : pour spécifier le nombre d'éléments à libérer. Non utilisé si l'option **Release by attribute** est cochée.

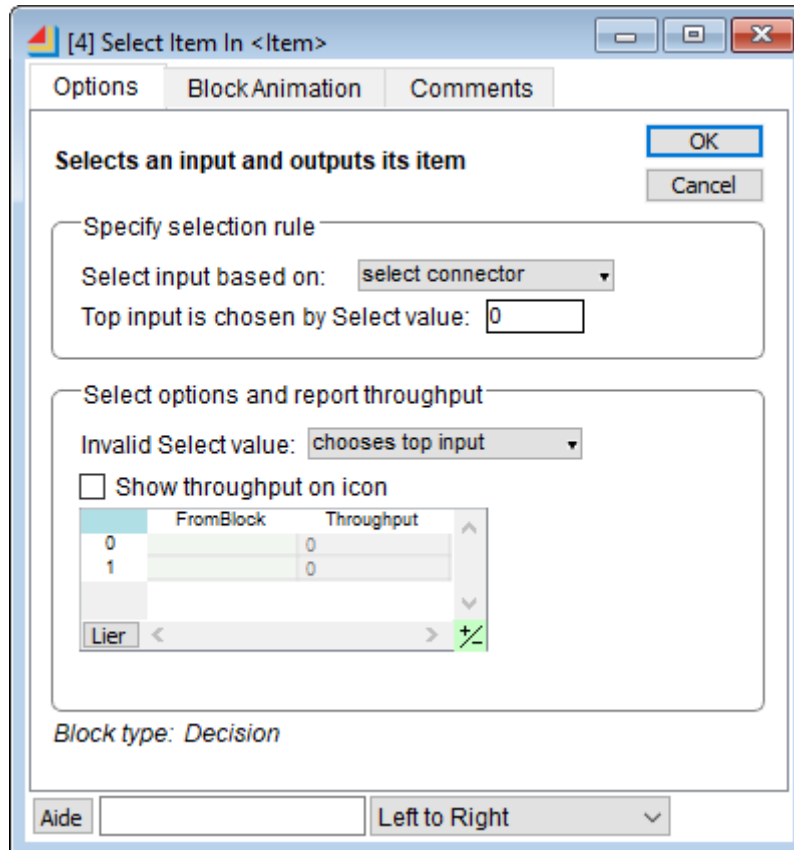
Animation

Le nom du Pool où retournent les éléments est affiché sur l'icône.

Select Item In



Sélectionne une entrée en fonction d'une décision.



L'entité présente sur l'entrée sélectionnée est transmise à la sortie. Le dialogue permet de choisir selon la priorité de l'entité, aléatoirement, séquentiellement, par simple convergence ou d'après la valeur du connecteur *select*. Vous pouvez utiliser ce bloc pour représenter les feux tricolores à une intersection, pour sélectionner des candidats, représenter l'accès d'interruption à une CPU, etc.

* Le bloc peut sélectionner la branche porteuse de l'entité de plus haute priorité.

* Le bloc peut s'appuyer sur une distribution aléatoire pour choisir la branche. Les probabilités de chaque branche devront être saisies dans la table. Si la somme des probabilités n'est pas égale à 1.0, le total sera écrit en rouge au bas de la table.

* Le bloc peut passer en séquence d'une entrée à la suivante à chaque entité, en commençant par celle du haut, et en bouclant.

* Avec l'option de convergence, le bloc acceptera les entités de toutes les branches. Les flux sont réunis en un seul flux discret, sans autre règle de décision.

* Le bloc peut lire la valeur sur le connecteur *select* pour choisir explicitement une branche par un numéro. Dans ce mode il faut aussi prévoir ce qui se passe si la valeur sur *select* ne correspond à aucune branche.

Si le connecteur *select* est lié à un ou plusieurs blocs *Get*, évitez de mettre les types de blocs suivants entre le Sélecteur et les blocs *Get*: Set, activités et files. Ces blocs peuvent modifier la valeur que vous utilisez pour votre choix.

Onglet Options

Select input based on (Choisir l'entrée) : sélectionnez une des logiques de choix : selon priorité de l'entité, aléatoirement, par connecteur select, séquentiellement, par convergence. Les éléments visibles dans le dialogue d'adaptent à ce choix de base.

Top input is chosen by Select value (Entrée du haut choisie si Select vaut) : valeur qui spécifie le connecteur du haut. Le connecteur suivant est désigné par la même quantité + 1, et ainsi de suite.

Use block seed (base aléatoire du bloc) : si cochée, entrez une valeur qui servira de base aléatoire locale pour la distribution utilisée par le bloc.

Select from (Choix à partir) : effectue le choix à partir de toutes les entrées ou seulement là où une entité attend. Dans le premier cas, une fois le choix fait, le bloc attend éventuellement qu'une entité se présente sur l'entrée choisie (même si des entités sont présentes sur les autres entrées). Dans le second cas, le bloc ne prend en compte que les entrées ayant une entité présente. Ainsi si la règle de décision est "aléatoirement" le bloc choisira au hasard parmi les entrées ayant une entité prête.

Show throughput on icon (Voir débit sur l'icône) : affiché le débit calculé sur l'icône du bloc

Invalid Select value (Valeur 'select' invalide) : le menu permet de choisir ce qui se passe si la valeur sur *select* ne correspond à aucune branche. Les choix sont : **chooses top input** (choisit entrée du haut), **chooses bottom input** (choisit entrée du bas) et **rejects input** (rejette l'entité). Dans le cas du rejet, le bloc ne laisse plus passer les entités, dans les autres cas, c'est un connecteur qui est préféré.

Colonne 1 From block (Bloc amont) : indique le nom du bloc auquel est relié le connecteur concerné.

Colonne 2 Probability : N'apparaît qu'avec le choix aléatoire. Entrez dans cette colonne les probabilités de chaque branche. Leur somme doit être égale à 1.0.

Colonne 2/3 Throughput (Débit) : ces cellules montrent le débit reçu sur chaque connecteur correspondant.

Equal Probabilities (Probabilités égales) : un clic sur ce bouton remplit la table de probabilités égales dont la somme vaut 1.0.

Connecteurs

Le connecteur d'entrée se déploie en autant de branches candidates que nécessaire.

En sortie l'entité est transmise telle quelle.

Le connecteur 'select' n'apparaît sur l'icône que lorsque l'option est choisie dans le dialogue.

Animation

L'icône montre quelle branche est sélectionnée.

Select Item Out



Sélectionne une sortie en fonction d'une décision.

L'entité présente en entrée est transmise à la sortie sélectionnée. Le dialogue permet de choisir selon une propriété de l'entité, la priorité des connecteurs, aléatoirement, séquentiellement, ou d'après la valeur du connecteur *select*. Vous pouvez utiliser ce bloc pour représenter les feux tricolores à une intersection, pour sélectionner des candidats, représenter l'accès d'interruption à une CPU, etc.

Vous devez d'abord choisir la logique pour orienter l'entité sur une branche. Les options sont :

* **Property** (Propriété de l'entité). La branche choisie est déterminée par un attribut ou la priorité de l'entité. Entrez les valeurs représentant les sorties dans la colonne *Sortie* ; par défaut 0 sélectionne la sortie du haut. Cette option retenant l'entité jusqu'à ce que la branche choisie soit acceptante, cela peut provoquer des blocages.

* **Connector priority** (Préférence des sorties). Le bloc essaie d'envoyer l'entité dans chaque sortie, dans l'ordre de préférence indiqué, jusqu'à ce qu'une sortie accepte l'entité. Un petit nombre indique une plus grande préférence pour cette branche (une valeur 1 est examinée avant une valeur 3).

Notez que c'est bien différent des priorités données aux entités, car c'est la branche qui a une priorité, pas l'entité.

* **Random** (Aléatoirement) s'appuie sur une distribution aléatoire pour choisir la branche. Les probabilités de chaque branche devront être saisies dans la table. Si la somme des probabilités n'est pas

égale à 1.0, le total sera écrit en rouge au bas de la table. Lorsque l'option **If output is blocked: item will try unblocked outputs** (Si sortie bloquée l'entité essaie une autre sortie) est choisie, le bloc essaie les autres connecteurs en séquence. Lorsque l'option **If output is blocked: item will wait for blocked output** (Si sortie bloquée l'entité attend un déblocage) est choisie, le bloc sélectionnera une sortie et attendra qu'elle puisse accepter l'entité ; ceci pourrait causer un blocage.

* **Select connector** (par connecteur Select). La valeur sur le connecteur *select* pour choisir explicitement une branche par un numéro. Dans ce mode il faut aussi prévoir ce qui se passe si la valeur sur *select* ne correspond à aucune branche. Dans le dialogue vous indiquez la valeur qui choisit la sortie du haut (0 par défaut). Les connecteurs suivants déterminent leur valeur en fonction de celle-ci, en ajoutant 1. Si la sortie du haut est choisie par 1, la seconde sortie sera choisie par 2, et ainsi de suite.

* **Sequential** (Séquentiellement). Passe en séquence d'une sortie à la suivante à chaque entité, en commençant par celle du haut, et en bouclant. Lorsque l'option **If output is blocked: item will try unblocked outputs** (Si sortie bloquée l'entité essaie une autre sortie) est choisie, le bloc essaie les autres connecteurs en séquence. Lorsque l'option **If output is blocked: item will wait for blocked output** (Si sortie bloquée l'entité attend un déblocage) est choisie, le bloc sélectionnera une sortie et attendra qu'elle puisse accepter l'entité ; ceci pourrait causer un blocage.

Si le connecteur *select* est lié à un ou plusieurs blocs *Get*, évitez de mettre les types de blocs suivants entre le Sélecteur et les blocs *Get*: Set, activités et files. Ces blocs peuvent modifier la valeur que vous utilisez pour votre choix.

Onglet Options

Select output based on (Choisir la sortie) : sélectionnez une option : selon propriété de l'entité, selon préférence des sorties, aléatoirement, par connecteur select, séquentiellement. Les éléments visibles dans le dialogue d'adaptent à ce choix de base.

Top output is chosen by Select value (Sortie du haut choisie par select =) : valeur qui spécifie le connecteur du haut. Le connecteur suivant est désigné par la même quantité + 1, et ainsi de suite.

Display string attributes in table (Attributs chaîne dans la table) : la table affichera les valeurs associées aux attributs chaîne.

Dynamically change priorities (Préférences dynamiques) : spécifie que les valeurs indiquant les préférences de connecteurs varient dans le temps. En cochant cette case, une matrice de connecteurs apparaît sur l'icône pour contrôler les valeurs de préférences.

If output is blocked (Si sortie bloquée) : permet de choisir une action en cas de blocage sur le connecteur de sortie. **Item will wait for blocked output** (L'entité attend un déblocage) signifie que le bloc bloquera l'entité jusqu'à qu'il y ait une disponibilité en aval. **Item will try unblocked outputs** (L'entité essaie une autre sortie) signifie que l'entité pourra passer par une sortie non bloquée si la sortie choisie initialement l'est. Notez que cela peut considérablement modifier le comportement du bloc et la logique des décisions. Par exemple, avec une logique aléatoire, le choix **Item will try unblocked outputs** va modifier les probabilités requises si une ou plusieurs sorties sont bloquées.

Invalid Select value (Valeur 'select' invalide) : le menu permet de choisir ce qui se passe si la valeur sur *select* ne correspond à aucune branche. Les choix sont : **chooses top output** (choisit sortie du haut), **chooses bottom output** (choisit sortie du bas), **rejects input** (rejette l'entité) et **generate error** (génère une erreur). Dans le cas du rejet, le bloc ne laisse plus passer les entités, dans les autres cas, c'est un connecteur qui est préféré.

In Select Output column N chooses top output (Dans la colonne Select output N choisit la sortie du haut) : entrez la valeur pour la sortie du haut. Le connecteur suivant est désigné par la même quantité + 1, et ainsi de suite.

Predict the part of the item before it enters this block (Prévoir la sortie d'avant d'accepter l'entité - bloc non stockeur) : propose une alternative au cas de blocage d'une sortie, en faisant que l'entité attend dans la file en amont, plutôt que dans le bloc Sélecteur. Cochez **Predict the part of the item**

before it enters this block, et le bloc demandera au bloc amont les propriétés de l'entité suivante, pour vérifier que la sortie correspondante est disponible. Alors seulement il accepte l'entité. Cela garantit qu'aucune entité ne se trouve "coincée" dans le bloc Sélecteur.

Show throughput on icon (Voir débit sur l'icône) : montre les valeurs de débit sur l'icône du bloc

Table

Colonne 1 To block (Bloc aval) : indique le nom du bloc auquel est relié le connecteur concerné.

Colonne 2 Priority : N'apparaît qu'avec le choix Priority. Entrez dans cette colonne les priorités données à chaque branche (ces valeurs seront écrasées par celles du connecteur si la case **Dynamically change priorities** est cochée.)

Colonne 2 Probability : N'apparaît qu'avec le choix aléatoire. Entrez dans cette colonne les probabilités de chaque branche. Leur somme doit être égale à 1.0.

Colonne 2 Property Value (Valeur propriété) : n'apparaît qu'en cas d'utilisation d'une propriété. Vous indiquez la valeur choisissant le connecteur correspondant.

Colonne 3 Select Output (Sortie) : spécifie quel connecteur de sortie sera choisi par la valeur d'attribut ou la probabilité spécifiée dans la ligne.

Colonne 2/3/4 Throughput (Débit) : ces cellules montrent le débit envoyé sur chaque connecteur correspondant.

Number of outputs = number of rows in table (Nombre de sorties = nombre de lignes dans la table) : modifie le nombre de connecteur de sortie sur l'icône, pour qu'il corresponde au nombre de lignes dans la table.

Fill down (Remplir) : un clic sur ce bouton utilise la valeur de **In Select Output column N chooses top output** comme point de départ pour remplir les valeurs de propriété et les numéros de sorties avec des valeurs par défaut.

Equal Probabilities (Probabilités égales) : un clic sur ce bouton remplit la table de probabilités égales dont la somme vaut 1.0.

Use block seed (base aléatoire du bloc) : si cochée, entrez une valeur qui servira de base aléatoire locale pour la distribution utilisée par le bloc.

Onglet Item Animation

Cet onglet ne présente pas de choix global. Une table permet de spécifier les choix pour chaque file interne :

Colonne 1 (Animation Option) : spécifie le choix d'animation pour chaque file interne. **No Change** "Tel quel" pour garder l'objet initial et **Change to** "changer en" pour pouvoir spécifier de nouveaux choix.

Colonne 2 (2D picture) : spécifie le symbole 2D associé à cette valeur.

Connecteurs

Le connecteur de sortie se déploie en autant de branches candidates que nécessaire.

En entrée, l'entité à envoyer sur la sortie sélectionnée.

Le connecteur 'select' n'apparaît sur l'icône que lorsque l'option est choisie dans le dialogue.

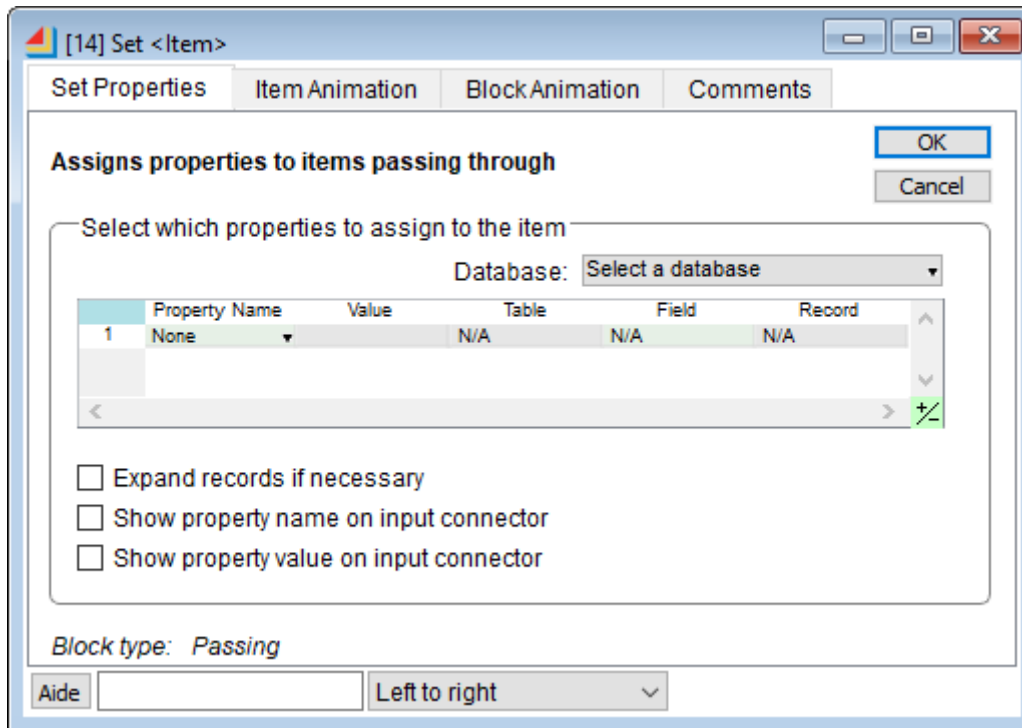
Animation

L'icône montre quelle branche est sélectionnée.

Set



Donne des propriétés aux entités avant de les faire passer. La valeur des propriétés est entrée dans le dialogue et par les connecteurs d'entrée.



Plusieurs propriétés peuvent être données, ce qui augmentera le nombre de connecteurs d'entrée reliés au modèle.

Les propriétés à affecter peuvent être des attributs, la priorité, et la quantité (valeur) de l'entité. Les attributs peuvent être des attributs chaîne ou numériques, et des attributs liés à la base de données Extendsim (DB Address).

Le connecteur DB permet de changer la base de données à laquelle le bloc fait référence. S'emploie pour des BDD de structure identique mais de contenu différent.

Onglet Propriétés

Table

Colonne 1 Property Name (Propriété) : spécifie la propriété à définir ou modifier sur l'entité. Le menu permet de choisir parmi des propriétés existantes ou d'en créer de nouvelles.

Colonne 2 Value (Valeur) : affiche la valeur à affecter à la propriété.

Colonne 2 Table : la colonne 3 s'emploie uniquement si la propriété sur la ligne associée est "DB address attribute", c'est à dire fait référence à la base de données. La colonne sert à spécifier la table, via un menu qui propose les options suivantes : **Select from list** (Dans une liste), **Type a name** (Entrer un nom), **Type an index** (Entrer un index), **Use a connector** (Utiliser un connecteur), **Get from attribute** (Utiliser un attribut). **Select from list** affiche une liste de tables pour y faire un choix. **Type an index** permet de saisir une valeur d'index. **Use a connector** affecte un connecteur d'entrée

continu à cette ligne afin que les valeurs de table soient lues sur un connecteur. **Get from attribute** permet de spécifier un attribut dans lequel lire la valeur.

Colonne 4 Field (Champ) : la colonne 4 s'emploie uniquement si la propriété sur la ligne associée est "DB address attribute". Un menu permet de choisir l'index de champ pour l'attribut de base de données. Les options sont identiques à celles pour la colonne Table.

Colonne 5 Record (Enregistrement) : la colonne 5 s'emploie uniquement si la propriété sur la ligne associée est "DB address attribute". Un menu permet de spécifier l'enregistrement pour l'attribut de base de données, par les options suivantes : **Type an index** (Entrer un index), **Use a connector** (Par un connecteur), **Get from attribute** (Par un attribut), **Last record** (Dernier enregistrement) et **Last record + 1** (Dernier enregistrement + 1). Les trois premières options sont comme celles décrites pour la colonne Table. **Last record** utilise le dernier enregistrement comme index pour la portion enregistrement de l'attribut de base de données. **Last record + 1** ajoute un enregistrement à la fin de la table sélectionnée et utilise cet enregistrement comme index pour la portion enregistrement de l'attribut de base de données. Notez : **Last record** et **Last record + 1** servent notamment lorsqu'on utilise des fonctions de type "DB Address" dans des blocs équation. La valeur d'un attribut de base de données sur l'entité qui arrive servira dans une équation qui s'appuie sur l'API DB Address.

Show property name on input connector (Nom sur le connecteur d'entrée) : affiche le nom de la propriété près du connecteur.

Show property value on input connector (Valeur sur le connecteur d'entrée) : affiche la valeur de la propriété près du connecteur.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Animate H-block object x to y (animer les objets de bloc H x jusque y) : anime la longueur de la file sur plusieurs objets d'animation présents sur le bloc hiérarchique englobant ce bloc. La plage est celle des numéros d'objets d'animation sur l'icône du bloc hiérarchique. Chaque objet dans cette plage animera une entité différente.

Connecteurs

L'entité entrante est renvoyée en sortie avec ses propriétés changées.

Matrice des connecteurs d'entrée : cette matrice contient les connecteurs des valeurs affectées aux propriétés.

DB : si relié, spécifie l'adresse de BDD à laquelle le bloc fait référence.

Animation

L'arrivée des entités anime l'icône du bloc.

Shift



Génère un planning dans le temps pouvant être utilisé pour modifier la capacité d'autres blocs dans le modèle.

L'horaire peut être de type ON/OFF ou représenté par un nombre. Dans le cas ON/OFF, les blocs utilisant cet horaire seront en inactivité (suspens) lorsque l'horaire est inactif, et en fonctionnement normal lorsque l'horaire est actif. Lorsque l'horaire est numérique, les blocs utilisant cet horaire modifieront leur capacité en fonction de la valeur courante de l'horaire (qui figure alors une équipe de travail). Seuls les blocs de capacité multiple (Activity, blocs de ressource) peuvent utiliser l'horaire numérique. Un bloc Shift peut contrôler les horaires de travail de n'importe quel nombre de blocs dans un modèle. Si l'horaire est modifié dans le bloc Shift, tous les blocs utilisant cet horaire reflèteront la modification.

Les horaires se répètent en cycle si la case "Repeat" (Répéter) est cochée et qu'une durée de cycle est spécifiée. Cela permet de planifier des changements d'équipe répétés sur 24 heures ou une semaine.

Le connecteur de sortie indique l'état courant de l'horaire (pour les horaires ON/OFF, ON = 1 et OFF = 0). Le connecteur d'entrée se combine avec l'horaire pour créer des grilles complexes. Si le connecteur d'entrée est inférieur à 0,5, l'horaire est considéré comme inactif. Si le connecteur d'entrée est supérieur à 0,5, l'horaire qui s'applique alors est utilisé. Par exemple un horaire de 40 heures par semaine se superposant à un horaire de 7 jours par semaine peut se construire facilement au moyen de

deux blocs Horaire. Le premier serait "ON" 5 jours sur 7, le second (en aval) serait "ON" seulement 8 heures chaque jour. Si la sortie de l'horaire hebdomadaire est connectée à l'entrée de l'horaire quotidien, l'horaire résultant du second bloc serait bien de 8 heures par jour 5 jours sur 7.

Les horaires peuvent être modifiés de manière interactive durant la simulation via le menu déroulant "Shift name".

Le bloc peut aussi s'employer comme un "verrou" pour plusieurs blocs. Avec un horaire toujours "ON", le connecteur d'entrée pourra piloter l'activité de plusieurs blocs simultanément.

NOTE IMPORTANTE :

Il est assez facile de définir un horaire complexe incluant les pauses, week-ends, vacances, etc. Mais avant d'implémenter une telle complexité dans un modèle, il faut être sûr que ces détails sont pertinents. Par exemple, si rien ne se passe durant les week-ends, il est préférable de créer une semaine de 5 jours, plutôt que d'ajouter un bloc Shift et de définir une inactivité durant les week-ends.

Onglet Shift

Shift name (Nom de l'horaire) : un nom unique qui identifie l'horaire. Jusqu'à 31 caractères.

Shift type (Type d'horaire) : menu déroulant spécifiant le type d'horaire. Soit ON/OFF soit un horaire numérique.

Time unit (Unité de tps) : si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, une unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Table

Colonne 1 Time (Heure) : spécifie les heures de changement d'état.

Colonne 2 Number or on/off (Nombre ou on/off) : spécifie le nouvel état. On/Off pour les horaires de type On/Off, ou un nombre pour les horaires numériques.

Colonne 3 Description : en option, peut servir à commenter cet état.

Sort (Trier) : trie le contenu de la table.

Shift times in numeric format / Shift times in calendar format (Horaire au format numérique/calendaire) : si le modèle utilise des dates calendaires, ce menu permet de choisir le format des dates dans le bloc.

Repeat every (Répéter toutes les) : l'horaire repart à zéro au moment indiqué. Par exemple si l'horaire occupe les heures de 1 à 5 et que vous entrez 5 dans "Repeat...", l'heure 6 aura le même effet que l'heure 1. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, une unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

If shift names are duplicated, find only closest (Utiliser le plus proche bloc Shift) : si cochée, les noms d'horaires en double ne seront pas montrés dans le menu des horaires situé dans les blocs qui peuvent utiliser les horaires. Les blocs conservés sont ceux hiérarchiquement proches du bloc utilisateur de l'horaire.

Onglet Results

Shift status (Horaire courant) : affiche l'état courant de l'horaire (ON ou OFF). Peut s'employer pour modifier interactivement l'état durant la simulation.

Last shift change time (Heure du dernier changement) : l'heure à laquelle l'horaire a changé d'état.

Time into current shif (Durée dans cet état) : affiche le temps passé depuis le dernier changement d'état.

Onglet Blocks

Blocks using this shift (Blocs utilisant cet horaire) : liste des blocs qui s'appuient sur cet horaire.

Refresh (Rafraîchir) : rafraîchir l'affichage des blocs utilisant cet horaire.

Check for duplicate shift names (Interdire les noms d'horaire en double) : si cochée, le bloc signale une erreur s'il rencontre un nom d'horaire déjà utilisé.

Show blocks using this shift on block icon (Voir sur l'icône les blocs utilisant cet horaire) : si cochée, le bloc affiche un symbole indiquant des blocs qui utilisent son horaire sur son icône et sur elle des blocs.

Connecteurs

Le connecteur d'entrée s'emploie pour modifier l'état de l'horaire. Si le connecteur d'entrée est inférieur à 0.5, l'horaire est considéré comme inactif (OFF). Si le connecteur d'entrée est supérieur à 0.5, l'horaire qui s'applique alors est utilisé.

Le connecteur de sortie indique l'état courant de l'horaire (ON = 1, OFF = 0, ou bien le nombre en activité pour un horaire numérique).

Animation

L'état courant est affiché au-dessus de l'icône.

Shutdown



Génère des signaux de "marche" et "arrêt", généralement utilisés pour piloter les blocs Activity et Valve.

[118] Shutdown <Item>

Shutdowns Options Results Item Animation Comments

Used to control shut downs OK Cancel

Set failure and repair behavior

Model failures of: a single component

Failures are: independent of repair TBF restarts after failure

Failures caused by: progression of time

Set time between failures (TBF)

Exponential Plot Sample

Mean: 1000

Location: 0

Set time to repair (TTR)

Exponential Plot Sample

Mean: 10

Location: 0

Select shutdown signal type

Send a value to signal shutdown

Down value: 1 Up value: 0

Block type: Source

Aide Left to right

Les arrêts peuvent être plutôt des pannes, c'est à dire provoqués aléatoirement, ou bien plutôt de type maintenance, c'est à dire issus d'un planning, ou encore mélanger ces deux logiques.

Arrêts aléatoires

Les arrêts aléatoires présentent deux types de comportements

1 - Vous spécifiez une distribution **Time between failures** ou Intervalle entre arrêts (TBF). Cette distribution correspond au temps entre le début d'arrêts consécutifs. Chaque fois que survient un arrêt et qu'une entité est générée, une durée aléatoire d'arrêt est calculée selon la distribution **Time to repair** ou Durée de réparation (TTR). Dans ce cas, deviennent visibles l'option permettant de spécifier la durée de l'arrêt (TTR) et une valeur d'attribut spécifiant quelles entités dans le bloc Activity ciblé seront affectées. (Voir Quelles entités et Quelle durée, ci-dessous).

Si les arrêts sont indépendants des réparations, cela signifie que le nouveau TBF est calculé dès que survient la panne, et cela quelle que soit la durée de l'arrêt.

2 - Au contraire s'ils sont dépendants des réparations, il faut attendre la fin de la réparation (TTR) pour commencer le calcul du nouveau TBF. Pour le dire autrement, dans le premier cas les périodes de TBF et TTR se superposent. La période de TTR est incluse dans celle de TBF. Dans le second les périodes ne se superposent pas. Le TTR suit le TBF. Le TBF suivant sera calculé lorsque le TTR sera terminé. Par exemple, dans le cas de la courroie de distribution d'une voiture, le temps de bon fonctionnement ne débute qu'une fois la voiture réparée et prête à rouler et non pas au moment où la précédente s'est abîmée.

Une ou plusieurs sources ?

Dans le cas le plus simple d'un équipement unique affecté par une seule source, deux menus au bas de l'onglet Arrêts déterminent le détail de la relation entre TBF et TTR.

Si "multiple components" (plusieurs sources) est choisi, les valeurs de TBF et TTR sont alors définies dans une table de BDD et non dans le dialogue du bloc. Cela vous permet de définir plusieurs schémas de pannes, et chaque ligne dans la table représente un schéma de pannes. Par exemple, supposons que l'on modélise les pannes d'un vélo, il faut prendre en compte à la fois l'usure de la chaîne et celle des pneus. Dans la mesure où la chaîne comme les pneus sont usés chaque fois que le vélo est utilisé, ET que la chaîne n'a pas le même taux d'usure que le pneu, les pannes simultanées seraient définies avec une table de deux lignes, une pour chaque schéma.

Dans ce mode vous devrez utiliser le mode Editeur pour modifier la BDD ExtendSim. Il faut définir deux champs dans la table, un pour les enregistrements de TBF et un pour les enregistrements de TTR. Chaque cellule sera généralement définie comme une distribution dans la BDD. Chaque ligne contiendra ainsi un enregistrement avec une distribution de TBF et un enregistrement avec une distribution de TTR, permettant un nombre indéfini d'enregistrements de schémas dans le bloc. Dans notre exemple, la table contiendrait deux lignes : une avec les informations de TBF/TTR pour la chaîne, l'autre avec les informations de TBF/TTR pour les pneus.

Option pour le mode 'multiple components' sources multiples :

1 - Sources dépendantes :

Les sources peuvent être définies comme étant "simultanées accumulatives", c'est à dire que le temps utilisé pour progresser vers une panne est le même pour toutes les sources de pannes. Elles sont dites dépendantes. C'est le cas du vélo : si la chaîne est cassée, les pneus ne s'usent pas !

2 - Sources indépendantes :

Si l'une est en panne, cela n'empêche pas une autre de continuer sa progression vers son arrêt prochain. Dans ce cas un autre menu est proposé, offrant donc un état synthétique.

a - indiquer l'état de chaque composant :

Dans le premier cas deux choses peuvent survenir :

- si le bloc génère des entités, une entité sera générée à chaque fois qu'un composant est en panne. NOTE : vous pouvez savoir quel composant est concerné en définissant l'attribut ID Source.
- si le bloc envoie un signal continu, le nombre de connecteurs de sortie continus sera égal au nombre d'enregistrements dans la table sélectionnée. Chaque connecteur indiquera l'état courant du composant qu'il représente.

b - indiquer l'état synthétique :

Si au moins un composant est en arrêt, le bloc émet un état d'arrêt. Vous pouvez utiliser la sortie IDA pour déterminer quel composant est responsable de l'actuel état d'arrêt. NOTE 1 : si un champ "ID" a été sélectionné, les valeurs émises par le connecteur IDA seront issues du champ ID. NOTE 2 : un signal d'arrêt par entité ne peut être sélectionné si les options "Multiple components" (sources indépendantes) et "Aggregate state" (Etat synthétique) sont sélectionnées. Pourquoi ? parce qu'un connecteur discret ne saurait transmettre l'état du bloc pannes dans une telle configuration.

Progression du temps ou comptage ?

Par défaut le bloc progresse avec le temps de la simulation. En mode 'Multiple components', il est généralement préférable de choisir 'Failures caused by progression of simulation time' (arrêts causés par la progression du temps). Dans cette configuration, il est possible d'utiliser un seul bloc Shutdown pour affecter plusieurs activités indépendantes.

Cependant, il est possible d'utiliser le connecteur "#" pour compter le passage d'entité afin de déclencher l'arrêt. Par exemple, le connecteur "#" d'un bloc Activity qui compte le nombre d'entité terminées peut être servir pour le connecteur "#" du bloc Shutdown.

Arrêts planifiés

Tous les arrêts ne sont pas aléatoires : vous pouvez programmer des arrêts de deux façons :

1. Vous utilisez le connecteur d'entrée "Sched". Par exemple, avec un bloc Create en mode Programme, une maintenance programmée peut être modélisée en reliant la sortie du Create au connecteur *Sched*. Dans cet exemple, lorsque le bloc Create envoie un signal TRUE (valeur ≥ 0.5), le bloc Shutdown arrête sa progression vers l'arrêt suivant et envoie un signal d'arrêt par son connecteur de sortie. Une fois l'arrêt programmé terminé, le bloc Create envoie un signal FALSE (valeur < 0.5), le bloc Shutdown envoie un signal de marche par son connecteur de sortie, et sa progression vers l'arrêt aléatoire suivant reprend. Comme nous parlons de valeurs booléennes (1 ou 0), le bloc Shutdown doit par ailleurs être utilisé en mode valeurs, et non en mode entités.

2. Des arrêts issus d'un bloc Shift présent dans le modèle peuvent être représentés en utilisant le menu Shift de l'onglet Options du bloc Shutdown. Par exemple pour modéliser des arrêts lorsque le personnel termine son poste. Dans un tel cas, lorsque le personnel travaille, le bloc Shutdown calcule sa progression vers l'arrêt aléatoire suivant. Mais lorsque le personnel est inactif, le bloc Shutdown envoie un signal d'arrêt par son connecteur de sortie, et sa progression vers l'arrêt aléatoire suivant s'interrompt.

Note : l'onglet Options propose plusieurs manières de prendre en compte les pannes/réparations en cours lorsqu'est reçu un signal d'arrêt programmé. Ces options sont examinées plus loin.

Avec un bloc Activity

Le signal d'arrêt ou de marche peut être envoyé par une entité ou une valeur continue, reliée au connecteur SD d'un bloc Activity.

Si "**Send a value to signal a shutdown**" (Envoyer une valeur pour signaler un arrêt) est choisi, le bloc Activity est entièrement arrêté si un signal d'arrêt est reçu : toutes les entités sont arrêtées, et aucune nouvelle entité ne peut entrer dans le bloc.

Si "**Send an item to signal a shutdown**" (Envoyer une entité pour signaler un arrêt) est choisi, le bloc Shutdown crée une entité à chaque événement d'arrêt (déterminé par la distribution TBF), et cette entité stocke le TTR dans un attribut. Un second attribut peut être ajouté pour cibler des entités spécifiques à arrêter. Notez que, contrairement au cas précédent, l'envoi d'une entité d'arrêt n'arrête pas le bloc tout entier, mais seulement des entités spécifiques dans le bloc Activity. Et si le bloc a encore de la capacité disponible, les entités qui se présentent en entrée peuvent être acceptées, même si certaines de celles présentes dans le bloc sont en arrêt.

Avec un bloc Valve

Lorsque vous le reliez à un bloc Valve (bib. Rate), les pannes contrôlent le débit de contrainte du bloc, donc la logique de 1 (true) pour les arrêts, et 0 (false) pour le fonctionnement est inversée. Zéro représentera l'arrêt (débit de zéro), et des valeurs autres que zéro représenteront un débit non nul. Notez aussi que la Vanne attend une valeur pour le connecteur R, donc l'option **Send an item to signal a shutdown** n'est pas applicable. La vanne a un bouton **Add Shutdown** (Ajouter Shutdown) qui ajoute automatiquement un bloc Shutdown au modèle.

Le bloc pousse les entités. Notez que l'unité de temps pour le TTR doit correspondre à celle du bloc qui subit l'arrêt (bloc Activity).

Onglet Arrêts

Paramétrage des arrêts et périodes de réparations

1. **Failures modeled using a single fail stream** (Arrêts modélisés par une seule source, voir introduction) : lorsque tous les objets affectés par la panne ont un schéma similaire.

TBF : Intervalle entre arrêts : sélectionnez la distribution pour le calcul de l'intervalle entre arrêts dans ce menu. Voyez en fin d'aide pour le choix des distributions.

TTR : Durée de l'arrêt : sélectionnez la distribution pour le calcul de la durée de l'arrêt dans ce menu. Voyez en fin d'aide pour le choix des distributions.

2. **Failures modeled using multiple fail streams** (Arrêts modélisés par plusieurs sources) : si choisie, le bloc devra faire appel à une base de données. Voir plus haut sur l'utilisation de ce mode.

DB : spécifie la base de données contenant la table utilisée pour les arrêts simultanés.

Table : spécifie la table utilisée pour les arrêts simultanés.

ID field (Champ ID) : en option, fonctionne en conjonction avec le connecteur "IDA" de sorte à identifier quel enregistrement dans la table est le schéma des arrêts utilisé pour générer les événements d'arrêt et de marche.

TBF field : spécifie le champ utilisé pour la valeur de TBF.

TTR field : spécifie le champ utilisé pour la valeur de durée de l'arrêt.

Down value field (Champ Arrêt) : en option, L'intersection entre ce champ et l'enregistrement courant du schéma des arrêts prévaut sur la valeur d'arrêt saisie dans le dialogue.

Up field (Champ Marche) : en option, L'intersection entre ce champ et l'enregistrement courant du schéma des arrêts prévaut sur la valeur de marche saisie dans le dialogue.

State field (Champ Etat) : en option, ce champ sert à stocker l'état courant de chaque composant.

Match field (Champ Correspondance) : en option, en conjonction avec le paramètre "Match value" ce champ sert à sélectionner un sous-ensemble d'enregistrements dans la table. S'applique notamment lorsqu'il existe une table principale des arrêts et qu'un bloc Shutdown est utilisé pour modéliser la fiabilité de sous-ensembles de composants.

3 – **Failures are independent / dependent of repair** (Les arrêts sont indépendants / dépendants de la réparation) : voir introduction.

4 - **Failures caused by Time / Count** (Arrêts causés par temps/comptage) : voir introduction.

5 – **Components are independent / dependent** (Les composants sont indépendants / dépendants) : voir introduction.

Choix du type de signal d'arrêt

1 - **Send a value to signal a shutdown** (Envoyer une valeur pour signaler un arrêt) :

Selon ce choix fondamental, c'est une valeur numérique qui signale un arrêt. Si *Send a value to signal a shutdown* est sélectionnée, le connecteur de sortie continu enverra la valeur spécifiée dans Down value (Valeur d'arrêt) pendant toute la durée de Time to repair. Elle enverra la valeur de Valeur de marche le reste du temps. Dans ce cas **Time between failures** sert à déterminer quand démarrer les intervalles de TTR.

Down value (Valeur d'arrêt) : la valeur émise par le bloc durant les arrêts.

Up value (Valeur de marche) : la valeur émise par le bloc entre les arrêts.

2 - **Send an item to signal a shutdown** (Envoyer une entité pour signaler un arrêt) :

Selon ce choix fondamental, c'est une entité qui détermine un arrêt.

a1 – Which items (Quelle entités - arrêts modélisés par une seule source) : cette option spécifie un filtre, via un attribut sur l'entité d'arrêt et qui recevra une valeur particulière. Si cette entité va dans le connecteur SD d'un bloc Activity qui est paramétré pour regarder cet attribut, l'arrêt ne concernera que les entités correspondant à cette valeur d'attribut.

a2 - Component ID (ID composant - arrêts modélisés par plusieurs sources) : cette option indique, via un attribut sur l'entité d'arrêt, le composant à l'origine de l'arrêt.

b – How long (Quelle durée) : cette option spécifie un attribut sur l'entité d'arrêt, et dont la valeur indiquera la valeur du temps de réparation aléatoire TTR.

Onglet Options

1 - Options Scheduled downs (arrêts programmés) : ce menu spécifie des comportements en cas d'arrêts planifiés parvenant par le connecteur *Sched*, et d'arrêts aléatoires calculés par le bloc. Ces options affectent l'interaction des arrêts qui proviennent du connecteur *Sched*, et les arrêts aléatoires générés par le TBF et le TTR. Dans les trois cas, les calculs du TBF/TTR sont interrompus par l'arrivée d'un arrêt planifié. Les options spécifient ce qui se passe lorsque le bloc reprend ses calculs :

TBF/TTR preserved (TBF/TTR sont conservés) : le bloc reprend là où il en est resté, qu'il ait été actif ou en arrêt au moment de l'interruption programmée.

TBF/TTR ignore schedule (TBF/TTR ignorent le programme) : le bloc continue son calcul de valeurs aléatoires durant l'arrêt programmé, et lorsque cet arrêt se terminera, il démarrera la phase qui devrait être en cours si aucune interruption programmée n'avait eu lieu.

TBF/TTR reset (TBF/TTR sont réinitialisés) : le bloc redémarre un nouveau TBF calculé à l'instant de la fin de l'arrêt planifié.

2 - Options d'horaire :

S'il existe des blocs Shift dans le modèle, vous pouvez sélectionner un horaire par le menu. Dans ce cas l'horaire se substitue aux valeurs que vous enverriez par le connecteur *Sched*.

Off shift (En horaire off) : Ce menu spécifie des comportements du bloc en cas d'événements liés aux horaires. Cf. ci-dessus les comportements pour les arrêts planifiés.

TBF/TTR sont conservés : le bloc reprend où en était resté, qu'il ait été actif ou en arrêt au moment où il est passé en horaire off.

TBF/TTR ignorent l'horaire : le bloc continue son calcul

TBF/TTR sont réinitialisés : le bloc redémarre un nouveau TBF calculé à l'instant du retour en horaire actif.

Use shift value as output signal (Valeur Horaire prévaut sur autres signaux) : cocher cette case signifie que la valeur émise par l'horaire prévaut sur la valeur saisie dans les zones *Down value* et *Up value*.

3 - Use block seed (Base aléatoire du bloc) : détermine un nombre pour définir la base aléatoire employée par ce bloc. La valeur finale est la base du bloc + le numéro du bloc. Si la case n'est pas cochée, la base est le numéro du bloc (un identifiant unique) +1. Dans la plupart des cas, chaque bloc devant générer des valeurs aléatoires devrait avoir sa propre base aléatoire. L'onglet Nombres aléatoires du dialogue Paramètres inclut un contrôle des bases aléatoires, qui vérifie qu'il n'y ait pas deux fois la même base aléatoire.

Onglet Results

1 - statistiques des arrêts :

Num unsched downs (Arrêts aléatoires) : nombre d'arrêts non planifiés pour cette simulation.

Num sched downs (Arrêts planifiés) : nombre d'arrêts planifiés pour cette simulation.

Num off-shift downs (Arrêts horaires off) : nombre d'arrêts en horaire off pour cette simulation.

Total num downs (Total des arrêts) : nombre total

Unsched down time (En arrêts aléatoires) : durée des arrêts non planifiés pour cette simulation.

Sched down time (En arrêts planifiés) : durée des arrêts planifiés pour cette simulation.

Off-shift down time (En horaires off) : durée des arrêts en horaire off pour cette simulation.

Total down time (Durée totale en arrêts) : durée totale

TBF Start time (Heure début de TBF) : heure de début du TBF en cours

TTR Start time (Heure début de TTR) : heure de début du TTR en cours

Table de suivi des sources multiples : si vous modélisez les pannes provenant de plusieurs composants, une table apparaît avec une ligne par composant. Dans ce cas, les statistiques individuelles au-dessus sont des statistiques synthétiques combinant tous les composants.

2 - Historique des valeurs TBF et TTR : si la case « **Record log...** » est cochée, ce tableau mémorise l'ID d'arrêt (si pannes provenant de plusieurs composants), l'heure de début du TBF, le TBF, l'heure de début des TTR et les TTR survenus durant la simulation.

Onglet Item Animation

1 - Symbole 2D : spécifie le symbole 2D de l'entité quittant ce bloc.

Connecteurs

1 - Connecteur de sortie discret : envoie l'entité contenant les informations d'arrêt.

2 - Connecteur de sortie continu : émet la valeur contenant les informations d'arrêt.

Connecteurs d'entrée continus : (dans l'ordre d'apparition.)

1 - TBF (Time between failures ou Temps de bon fonctionnement) : prévaut sur le TBF indiqué dans le bloc. Si ce connecteur est relié, sa valeur indique l'intervalle entre pannes.

2 - TTR (Time to repair ou temps de réparation) : prévaut sur le TTR indiqué dans le bloc. Si ce connecteur est relié, sa valeur indique le temps de réparation (ou durée de la panne).

3 - Wear : Si ce connecteur est relié au connecteur L d'une Activity, il va noter quand l'activité est en cours et quand elle est inactive. Les calculs du TBF ne feront référence qu'au temps effectivement travaillé pour déterminer les arrêts. Notez que la valeur de L est prise en compte dans son aspect quantitatif également, c'est à dire que si L vaut 2, le prochain arrêt aura lieu dans moitié moins de temps que si L vaut 1. Si vous voulez que L soit considéré comme un booléen, indépendant du nombre d'entités dans le bloc, utilisez un bloc Math ou Equation pour modifier le signal.

4 - which : prévaut sur la valeur quelles entités qui contrôle les entités affectées par l'arrêt. (Voir Quelles entités plus haut.)

5 - sched. (Programmé) : à utiliser lorsque vous voulez ajouter des arrêts planifiés (par exemple maintenance) à des arrêts aléatoires calculés par le bloc. Les valeurs arrivant sur ce connecteur sont considérées comme des booléens, où TRUE provoque immédiatement un arrêt planifié.

6 - DV (Valeur d'arrêt) : prévaut sur la valeur d'arrêt. Si ce connecteur est relié, sa valeur sera utilisée comme valeur émise pendant la durée des arrêts.

7 - UV (Valeur de marche) : prévaut sur la valeur de marche. Si ce connecteur est relié, sa valeur sera utilisée comme valeur émise entre les arrêts.

8 - # (nombre) : lorsque l'option *Failures caused by the count at the “#” connector* est choisie, ce connecteur doit être relié. Souvent il sera relié au “#” du bloc Activity, qui lui indiquera combien d'entité ont quitté le bloc.

9 - MV (Valeur correspondance) : ce connecteur n'est utilisé que si un champ correspondance est défini dans les options des arrêts de sources multiples. Il permet de passer outre la valeur spécifiée dans la boîte de dialogue. De cette façon, dans le cas d'un bloc hiérarchique sauvegardé en bibliothèque, il est possible d'assigner via ce connecteur MV un sous-ensemble d'arrêt spécifique pour chaque instance du bloc hiérarchique. NOTE : pour l'instant, ce connecteur n'est pas apte à recevoir des messages lors de la phase de simulation.

Connecteurs de sortie continus : (dans l'ordre d'apparition.)

1 - TD (nb total d'arrêts) : nombre total d'arrêts survenus.

2 - SD (nb d'arrêts par programme) : nombre d'arrêts par programme survenus.

3 - UD (nb d'arrêts aléatoires) : nombre total d'arrêts aléatoires survenus.

4 - TDT (temps total en arrêt) : total du temps d'arrêt accumulé.

5 - SDT (temps total en arrêt par programme) : total du temps d'arrêt par programme accumulé.

6 - UDT (temps total en arrêt aléatoire) : total du temps d'arrêt aléatoire accumulé.

7 - TBF (heure début TBF) : heure de début du TBF courant.

8 - TTR (heure début TTR) : heure de début du TTR courant.

9 - FID (ID arrêt) : valeur ID de la panne en cours. Ce connecteur n'a de signification que si 1.) le bloc est en mode 'plusieurs sources', et 2.) qu'un champ ID a été choisi dans la table.

10 - OSD (nombre total d'arrêt par horaire off) : Nombre total d'arrêt du à un horaire off pendant la durée de la simulation

11 - OST (temps total en horaire off) : total du temps en horaire inactif accumulé.

Distributions :

1 - Beta : distribution continue ayant des limites haute et basse finies. Les deux paramètres 'forme' peuvent servir à générer une grande variété de types de densité bornée. La limite basse est le paramètre Position ; la limite haute est le paramètre Maximum. Cette distribution est souvent utilisée pour une estimation grossière en l'absence de données, pour une distribution d'une proportion aléatoire, ou pour une durée de traitement. La distribution Uniforme est un cas particulier de la distribution Beta (les deux paramètres 'forme' valent un).

2 - Binomiale : sort une valeur correspondant au nombre de succès (argument (1) - Probabilité) parmi un nombre fixe d'essais indépendants (argument (2) - N). La probabilité est un nombre réel et N est un entier. Par exemple, cette distribution peut montrer le nombre de pièces défectueuses dans un lot de taille N, la probabilité d'erreur dans la transmission d'un message consistant en un nombre spécifique de bits, ou la probabilité qu'un certain nombre de personnes guérissent d'une maladie rare.

3 - Cauchy : distribution continue sans limites ayant un pic central accentué mais des traînes particulièrement longues. Celles-ci sont beaucoup plus lourdes que celles d'une distribution Normale. Lambda est une valeur d'échelle.

4 - Khi 2 : distribution continue ayant une limite gauche. Notez que la distribution Khi 2 est un sous-ensemble de la distribution Gamma avec $\beta = 2$ et $\alpha = \nu/2$. Nu est un paramètre de forme.

5 - Constante : émet une valeur constante.

6 - Table empirique : entre des valeurs dans la première colonne et entre la probabilité sur 100 d'obtenir cette valeur dans la deuxième colonne. La colonne Valeur contient les différentes valeurs à émettre ; la colonne Probabilité donne la chance d'obtenir cette valeur. Les probabilités ne peuvent être que relatives car ExtendSim les met automatiquement à l'échelle. Vous pouvez entrer les valeurs directement ou les importer dans la table de données par le Presse-Papier en utilisant les commandes du menu Edition. Discrète : ne concerne que la distribution empirique. Les données de la table de données sont utilisées comme probabilités discrètes des valeurs données dans la colonne "Valeur".

C'est-à-dire que les valeurs listées dans la colonne "Valeur" correspondent aux nombres exacts émis par le bloc.

7 - Table empirique Échelonnée : avec la distribution Empirique uniquement. Les données de la table de données seront considérées comme des probabilités d'une plage de données. La plus petite valeur dans la colonne Valeur définit la limite basse de la classe, la valeur suivante définit la valeur haute de cette même classe. Les distributions échelonnées imposent que le dernier ensemble de points définisse la limite haute de la distribution. C'est pourquoi le niveau de probabilité des deux derniers points doit être le même. Se cela n'est pas le cas dans la table de données, un point supplémentaire sera ajouté aux données.

8 -Table empirique Interpolée : avec la distribution Empirique uniquement. La distribution de probabilité sera interpolée entre les points de données. Les valeurs émises seront les valeurs de la table et les valeurs situées entre celles-ci. La probabilité qu'une valeur soit émise est ajustée.

9 - Erlang : émet une valeur qui varie autour de la moyenne donnée (1), avec de nombreux résultats possibles en fonction de la valeur du deuxième argument, "K". Cette distribution est utilisée dans le trafic téléphonique et la théorie des files d'attente lorsqu'un temps d'activité ou de service se décompose en plusieurs phases, chacune étant distribuée de façon exponentielle. La distribution Erlang est souvent utilisée comme temps de service lorsque vous voulez simplifier un modèle en combinant plusieurs étapes similaires en une seule étape représentative. La valeur de "K" doit être un entier. Comme pour la distribution Weibull, la courbe ressemble à d'autres distribution selon la valeur de sa moyenne, et surtout selon la valeur de "K". Si "K" vaut 1, la courbe ressemble à une distribution exponentielle alors que des valeurs plus grandes font que la courbe se rapproche d'une distribution normale.

10 - Exponentielle : distribution ayant la forme d'une exponentielle déclinante. Émet une valeur qui varie autour de la Moyenne (1), celle-ci étant un nombre réel non négatif. Cependant, la distribution est biaisée (longue traîne sur la droite), il est donc plus probable que les valeurs soient comprises entre 0 et la Moyenne, qu'entre la Moyenne et deux fois la Moyenne. Cette distribution est la plus utilisée en sciences, dans les processus d'entreprise et dans la théorie des files d'attente. S'emploie pour représenter la longueur d'une conversation téléphonique, la durée de vie de composants électroniques, le temps entre deux pannes, ou toute autre situation dans laquelle les événements sont totalement indépendants les uns des autres. Généralement, cette distribution n'est pas adaptée à la modélisation de délais ou de temps de traitement.

11 - Valeur Extrême Type 1A : distribution continue sans limites. Elle est également appelée Loi de Gumbel. Elle est utilisée pour trouver les extrema d'un nombre d'échantillon de plusieurs distributions. Beta est un paramètre d'échelle.

12 - Valeur Extrême Type 1B : distribution continue sans limites. La distribution Extreme Value IB distribution modélise les maxima de longues séquences (finies) de variables aléatoires. Beta est un paramètre d'échelle.

13 - Gamma : distribution continue limitée par zéro à gauche, et sans limite à droite. Les lois Exponentielle et d'Erlang sont des cas spéciaux de la distribution Gamma. Très souple, la distribution Gamma peut s'employer pour des intervalles entre arrivées, des durées de tâches ou des durées de vie.

14 - Géométrique : distribution discrète limitée par zéro à gauche, et sans limite à droite. Peut être définie comme le nombre d'échecs avant un succès dans une série d'essais. Sa forme est similaire à une distribution exponentielle. Les emplois traditionnels sont : demande dans un stock, nombre d'entités inspectées avant de trouver un défaut.

15 - HyperExponentielle : une distribution utilisée dans le trafic téléphonique et la théorie des files d'attentes, spécifiée par sa Moyenne. Elle perturbe la distribution Exponentielle à l'opposé de Erlang. Le deuxième argument, s, s'échelonne entre 0 et 0.5 avec 0.5 qui donne une distribution Exponentielle.

16 - Hypergéométrique : distribution discrète limitée par [0,s]. Elle décrit le nombre de défauts, x, dans un échantillon de taille s, parmi une population de taille N qui a m défauts au total.

17 - Normale inverse : distribution continue avec une limite basse. Vaut toujours zéro pour x minimum et comporte toujours un biais positif. La distribution Normale Inverse est également connue sous le nom de distribution de Wald. Alpha est un paramètre de forme. Beta est un mélange de forme et d'échelle.

18 - Weibull Inverse : distribution continue avec une limite basse. Vaut toujours zéro pour x minimum et comporte toujours un biais positif. En général, la distribution Weibull inverse se caractérise par des pics très accentués mais limités, et une longue traîne positive. Alpha est un paramètre de forme. Beta est un mélange de forme et d'échelle.

19 - Johnson SB : distribution continue avec des limites haute et basse finies. Les distributions Johnson SB, Lognormale et Johnson SU sont des transformations de la distribution Normale et permettent de décrire un ensemble d'événements ayant des causes naturelles. Lambda est la plage de valeurs de X au-dessus du minimum. Gamma est un paramètre d'asymétrie. Delta est un paramètre de forme.

20 - Johnson SU : distribution continue sans limites. Les distributions Johnson SU, Lognormale et Johnson SB sont des transformations de la distribution Normale et peuvent être utilisées pour décrire un ensemble d'événements ayant des causes naturelles. Lambda est la plage de valeurs de X au-dessus du minimum. Gamma est un paramètre d'asymétrie. Delta est un paramètre de forme.

21 - Laplace : parfois appelée distribution double exponentielle, la distribution Laplace est une distribution continue sans limites avec un pic central très accentué au niveau de theta. La distribution est à l'échelle de phi.

22 - Logarithmique : distribution discrète limitée par [1,...]. En général, si les données sont limitées par [0, ...], il faut alors traduire les données avant l'adéquation. Theta est lié à la taille de l'échantillon et à la moyenne.

23 - Logistique : distribution continue sans limites, symétrique autour de sa moyenne. Sa forme est semblable à celle de la distribution Normale, sauf que la distribution Logistique a une traîne plus longue. Alpha est un paramètre de décalage. Beta est un paramètre d'échelle.

24 - Log-Logistique : Pour Forme = 1, ressemble à une distribution exponentielle. Pour Forme < 1, tend vers l'infini à Position et décroît quand X augmente. Pour Forme > 1, vaut zéro à Position, puis forme un pic et décroît.

25 - Lognormale : logarithme naturel de la variable qui suit une courbe gaussienne par (1) Moyenne et (2) Écart-type. Cette distribution émet une valeur > 0 et est biaisée de façon à ce que la plupart des valeurs soient proches de la valeur minimum (biais positif). Cette distribution s'emploie souvent pour des traitements multiplicatifs, alors que la distribution Normale convient mieux pour les traitements additifs. Cette distribution est très utilisée pour examiner la valeur vénale de gage, comme un taux de rendement ou un retour sur investissement.

26 - Négative Binomiale : Nombre d'échecs avant le Sème succès. P spécifie la probabilité de succès.

27 - Normale : courbe gaussienne spécifiée par (1) Moyenne et (2) Écart-type. Sort une valeur à peu près égale à la moyenne à chaque unité de temps, la valeur pouvant être comprise entre 0 et la moyenne ou entre la moyenne et deux fois la moyenne. La moyenne est un nombre réel et l'écart-type est un nombre réel non négatif. Plus l'écart-type est grand, plus on trouve de valeurs écartées de la moyenne. Par exemple, si la moyenne est de 6 et que la probabilité que les valeurs se situent à ± 4 est de 68 % (c'est-à-dire que 68 % des valeurs tombent entre 2 et 10), vous entrez un écart-type de 4. Le calcul effectué est 4/1, où 1 représente 1 écart-type, largeur de valeurs (68 %). Cependant, si la probabilité que les valeurs tombent dans cette même fourchette est de 96 %, ou 2 écarts-type, entrez un écart-type de 2. Le calcul effectué est 4/2.

28 - Pareto : distribution continue avec une limite basse. Elle a une valeur finie pour x minimum et décroît régulièrement quand x augmente. Une variable aléatoire Pareto est l'exponentielle d'une variable aléatoire Exponentielle, et possède des caractéristiques semblables. Minimum est la valeur minimum de X. Alpha est un paramètre d'échelle.

29 - Pearson type V : une distribution typiquement utilisée pour représenter le temps mis pour effectuer une tâche. C'est une distribution continue limitée par zéro à gauche, et sans limite à droite. La densité à une forme similaire à lognormale mais peut avoir un pic plus grand près de $x = 0$.

30 - Pearson type VI : une distribution typiquement utilisée pour représenter le temps mis pour effectuer une tâche. C'est une distribution continue limitée par zéro à gauche, et sans limite à droite.

31 - Poisson : décrit le nombre d'événements qui se produisent dans un intervalle donné en fonction de (1) Moyenne. La variance est égale à la moyenne, donc plus la moyenne est grande, plus on trouvera de valeurs éloignées de la moyenne. Cette distribution est utilisée pour représenter le nombre d'appels téléphoniques par minute, le nombre d'erreurs par page ou le nombre d'arrivées dans un système.

32 - Puissance : distribution continue avec des limites haute et basse finies. C'est un cas particulier de la distribution Beta avec $q = 1$. La distribution Uniforme est un cas particulier de la distribution Fonction Puissance avec $p = 1$. Minimum est la valeur minimum, Maximum est la valeur maximum. Alpha est un paramètre de forme.

33 - Rayleigh : distribution continue avec une limite basse. Elle est souvent employée pour représenter des durées de vie car son taux d'incident augmente de façon linéaire avec le temps, par exemple la durée de vie de valves. Cette distribution est également utilisée pour les problèmes de bruit en communication. Sigma est un paramètre d'échelle.

34 - Triangulaire : sort une valeur N , N étant un nombre réel (décimal) supérieur ou égal au nombre réel sélectionné pour l'argument 1 (le minimum) et inférieur ou égal au réel sélectionné pour l'argument 2 (le maximum), avec la consigne supplémentaire que N tend vers une valeur plus probable. Vous pouvez utiliser cette distribution pour spécifier une distribution dans laquelle vous connaissez la plus petite valeur possible, la plus grande valeur possible et une tendance centrale. Cette distribution se comportera comme une distribution Normale sauf qu'elle pourra être biaisée (si la valeur 'plus probable' est spécifiée à gauche ou à droite de la moyenne) et qu'aucune valeur ne dépassera les bornes. Notez que la valeur 'plus probable' est le mode, pas la moyenne. Pour déterminer la moyenne d'une distribution triangulaire, faites la somme des valeurs minimum, maximum et plus probable, puis divisez le total par 3.

35 - Uniforme, entier : sort un nombre entier supérieur ou égal à l'entier sélectionné pour l'argument 1 et inférieur ou égal à l'entier sélectionné pour l'argument 2. Dans cette distribution, toutes les valeurs entières situées entre le minimum et le maximum ont la même probabilité. Par exemple, vous pouvez utiliser cette distribution pour indiquer que le prix d'un nouveau produit doit se situer entre 200 et 400 € ou pour montrer des valeurs représentant ce qui pourrait se passer "dans le meilleur des cas" et "dans le pire des cas".

36 - Uniforme, réel : (par défaut) sort un nombre réel (décimal) supérieur ou égal à la valeur sélectionnée pour l'argument 1 et inférieur ou égal à la valeur sélectionnée pour l'argument 2. Dans cette distribution, toutes les valeurs situées entre le minimum et le maximum ont la même probabilité. Par exemple, vous pouvez utiliser cette distribution pour représenter quel serait le "meilleur scénario" ou le "pire scénario", ou bien pour indiquer qu'un équipement coûterait au minimum 347,50 € et au maximum 452,95 €.

37 - Weibull : Cette distribution peut avoir les propriétés d'autres distributions (comme Exponentielle ou Rayleigh) en fonction des arguments (1) Echelle et (2) Forme, ceux-ci étant des nombres réels non négatifs. Elle est souvent utilisée pour décrire des taux d'échec, des durées de vie ou le temps mis pour effectuer une tâche. Sa courbe change considérablement selon la valeur d'Echelle et surtout la valeur de Forme. La variable Forme doit être supérieur à 0. Par exemple, si l'Echelle est 1 et la Forme est 1, la distribution Weibull est essentiellement une distribution exponentielle. Cependant, si l'Echelle est 1 et la Forme est 2, la courbe ressemble à une distribution normale biaisée. Notez que d'autres termes peuvent être utilisés pour l'échelle et la forme de la distribution Weibull. Reportez-vous à la documentation et lisez-là attentivement. Dans Stat:Fit, alpha est la forme et beta est l'échelle.

References

Pritsker, Introduction to Simulation and SLAM II, Halstead Press, pg. 715, 1986.

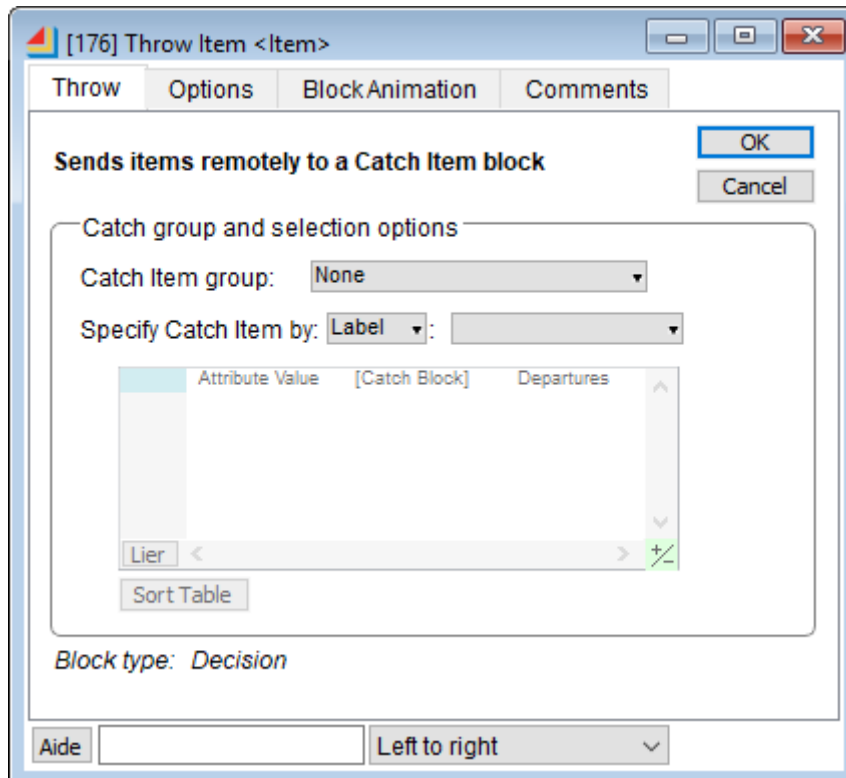
Gordon, System Simulation, Prentice Hall, pp. 125 - 150, 1978.

Law and Kelton, Simulation Modeling & Analysis, McGraw Hill, pp 330-353, 1982

Throw Item



Ce bloc "envoi" des entités à un bloc Catch Item. N'importe quel nombre de blocs Throw peuvent envoyer leurs entités à un unique bloc Catch. Les entités sont transmises sans connecteur, même via des blocs hiérarchiques. Le lien entre blocs peut se faire en spécifiant le label et le numéro du bloc du bloc Catch correspondant, ou à partir d'une valeur d'attribut.



Onglet Throw

Catch Item group (Groupes de réception) : spécifie le groupe de blocs Catch parmi lequel choisir par le menu déroulant. Cela permet d'organiser des familles de blocs Throw et Catch.

Specify Catch block by attribute value (Spécifier Catch par valeur d'attribut) : si cette option est sélectionnée, l'attribut et le tableau de correspondance déterminent vers quel bloc Catch ira l'entité. Choisissez d'abord dans le menu déroulant l'attribut qui définit l'orientation de l'entité. Entrez les valeurs possibles pour cet attribut dans la colonne "Attrib value" (Valeur d'attribut) de la table. Pour chaque valeur, sélectionnez le bloc Catch correspondant par le menu associé à chaque ligne de la colonne "Catch".

Specify Catch block by label (Spécifier Catch par label) : si cette option est sélectionnée, toutes les entités passant par ce bloc iront vers le bloc Catch spécifié dans le menu déroulant.

Sort table (Trier la table) : trie le tableau par valeur d'attribut.

Onglet Options

Use attribute value as Catch Item block number (Utiliser la valeur d'attribut pour n° de bloc Catch) : les numéros d'attribut sont directement des numéros de blocs Catch, sans table de correspondance.

Also show Catch block numbers (Voir aussi n° des blocs Catch) : ajoute le numéro du bloc Catch à son label dans les menus déroulants. Peut servir à distinguer des blocs ayant le même label.

If block labels are duplicated, find only closest (Si labels en double, envoyer au plus proche) : si deux blocs Catch (ou plus) ont le même label, seul le plus proche dans la hiérarchie sera proposé dans le menu déroulant pour être utilisé par ce bloc Throw. Si les deux blocs Catch sont au même niveau hiérarchique, le label de bloc Catch sera ignoré. Cette option sert à limiter le nombre de blocs considérés, notamment dans des blocs hiérarchiques dupliqués ou issus d'une bibliothèque.

Connecteurs

La sortie est l'entité passée au bloc Catch.

Animation

L'icône montre quand une entité arrive.

Le nom du bloc Catch destinataire apparaît sur l'icône, si le bloc envoie vers un bloc Catch spécifique.

Transport



Transporte des entités d'un point à un autre en fonction d'informations de vitesse et distance.

[129] Transport <Item>

Behavior Cost Results Transport Animation Item Animation

Block Animation Comments

Moves items from one block to another OK Cancel

Define transport capacity

Capacity:

Use shift:

Define how fast and how far the items move

Travel time: in a straight line

Move time: time units Calculate

Distance:

Item speed: feet / time unit

Items can not pass

Select From and To locations for calculated distance

2D	2D
From X location: <input type="text" value="976"/>	To X location: <input type="text" value="976"/>
From Y location: <input type="text" value="159"/>	To Y location: <input type="text" value="159"/>

From location is:

To location is:

Calculate distance:

Distance ratio:

Block type: Residence

Aide

Si vous spécifiez que le temps de trajet est la **vitesse et distance calculée** (speed and calculated distance), les positions de départ de d'arrivée des entités transportées par le bloc devront être définies dans l'onglet Transport. Cette option sert notamment pour l'animation 2D et 3D du bloc. Le temps de trajet peut aussi être le **temps du transport** (Move time), ou une valeur dépendant des informations de **Distance** et de **Vitesse** (Item speed) définies dans l'onglet.

Tout comme le bloc Convoyeur, ce bloc représente des entités transportées simultanément sur une certaine distance.

L'animation des entités peut se faire en 2D. Si vous sélectionnez l'option **2D animation shows simultaneous item movement** (Animation 2D montre les mouvements simultanés) dans l'onglet

Transport Animation, vous verrez bouger plusieurs entités le long des liens dans le modèle. Notez que pour activer cette option, vous devez désactiver la commande *Animer les liens* dans le menu Simulation, car elle interfère avec cette animation.

Onglet Transport

Block capacity (Capacité) : nombre maximum qu'entités que peut contenir le bloc.

Use shift (Utiliser l'horaire) : sélectionnez l'horaire s'appliquant à ce bloc (de type ON/OFF ou indiquant un nombre maximum d'entité dans le bloc)

Trael time (Temps trajet) : permet de spécifier comment calculer le temps pris par une entité pour aller du point de départ au point d'arrivée. Si vous spécifiez **Move time** (Temps du transport), le paramètre numérique **Move time** déterminera le temps de trajet. Si vous spécifiez **distance and speed** (distance et vitesse), le temps de trajet sera calculé à partir de ces deux informations. Si vous spécifiez **calculated distance and speed** (distance et vitesse calculées), le temps de trajet sera calculé à partir de la distance et la vitesse une fois la distance calculée à partir des options au bas de l'onglet.

Move time : spécifie le temps total pris par l'entité pour aller du point de départ au point d'arrivée. Le menu à sa droite permet de choisir l'unité de temps du **Move time**.

Calculate (Calculer) : le bouton calcule le temps de déplacement du bloc en fonction des options au bas de l'onglet.

Distance : spécifie la distance requise pour transporter les entités.

Speed (Vitesse) : spécifie la vitesse des entités transportées

Items cannot pass (Dépassement interdit) : si cochée, les entités ne peuvent se doubler si certaines avancent plus vite que d'autres. L'occupation de chaque entité sur le convoyeur doit alors être indiquée (sous forme d'une constante, d'une valeur lue sur un attribut ou un connecteur).

From X location (position De X) : Position X de départ pour calculer la distance.

From Y location (position De Y) : Position Y de départ pour calculer la distance.

To X location (position Vers X) : Position X d'arrivée pour calculer la distance.

To Y location (position Vers Y) : Position Y d'arrivée pour calculer la distance.

From location is (position De est) : ce menu permet d'indiquer comment sera donnée la position de départ (De) pour le transport de l'entité. Les options sont : valeur donnée pour X et Y; position du bloc; bloc hiérarchique supérieur; bloc précédent et bloc stockeur précédent. La position est donnée par des coordonnées ou définie par les emplacements des blocs.

To location is (position Vers est) : ce menu permet d'indiquer comment sera donnée la position d'arrivée (Vers) pour le transport de l'entité. Les options sont : valeurs données pour X et Y, position du bloc, bloc H supérieur du bloc suivant, bloc suivant, et bloc stockeur suivant. La position est donnée par des coordonnées ou définie par les emplacements des blocs.

Calculate distance (calculer distance) : en suivant les liens (**along connections**) inclura les lignes à angle droit dans le calcul de la distance de transport. En ligne droite (**in a straight line**) ignorera les liens. Notez que cette option affecte aussi la manière dont les entités sont animées, car en suivant les liens fera que les entités suivront les liens en animation 2D.

Distance Ratio (ratio distance) : ce menu permet de spécifier le ratio entre les pixels sur la table de travail, et la longueur du transport. Servira à contrôler comment les longueurs définies dans le dialogue du bloc affecteront l'animation (2D). Les options sont : déduit de vitesse et distance (**Use speed and distance directly**) ; utiliser un ratio de (**Use distance ratio of**). Déduit de vitesse et distance utilisera un ratio de 1. (i.e. un pixel égal un mètre/pied.). Vous pouvez également spécifier votre ratio.

Onglet coût

Define costs (Définir des coûts) : active les options de coût du bloc.

Transportation cost (Coût de transport) : le coût ajouté à chaque entité par unité de temps passé dans le transport. L'unité de temps définissant ce coût doit être cohérente avec les unités de temps utilisées dans le modèle. Tant que cet élément de ressource est associé à une entité, l'entité accumule le coût indiqué. Le coût est ajouté à l'attribut "_cost" de l'entité. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, cette unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Cost per item (Coût par entité) : coût ajouté à l'attribut "_cost" de chaque entité passant par ce bloc. C'est une constante, quelle que soit la durée du transport.

Total cost (Coût total) : le coût total accumulé par ce bloc durant la simulation.

Onglet Results

Comptage en cours : le nombre d'entité dans le bloc à cet instant de la simulation. (ou en fin de simulation, si elle est terminée.)

Comptage moyen : nombre moyen d'entités dans le bloc.

Arrivées : nombre d'entités entrées dans le bloc.

Départs : nombre d'entités ayant quitté le bloc.

Utilisation : pourcentage du temps où le bloc a été utilisé sur l'ensemble de la simulation. Dépend de la capacité : par exemple, si la capacité est de 5, et que le bloc transporte 2 entités 100% du temps, l'utilisation est de .40 (40%).

Total cost (Coût total) : le coût total d'attente de toutes les entités passées par le bloc.

Onglet Animation Transport

From X location (position X De) : les coordonnées X de la position de départ pour l'entité 2D. Peut être défini directement lorsque "Position De est : valeur donnée pour X et Y", sinon est calculé par le bloc. Cette option est désactivée si "*Travel time is speed and calculated length*" est sélectionné. Dans ce cas, la position De est déterminée par le champ du même nom.

From Y location (position Y De) : les coordonnées Y de la position de départ pour l'entité 2D. Peut être défini directement lorsque "Position De est : valeur donnée pour X et Y", sinon est calculé par le bloc. Cette option est désactivée si "*Travel time is speed and calculated length*" est sélectionné. Dans ce cas, la position De est déterminée par le champ du même nom.

To X location (position X Vers) : les coordonnées X de la position d'arrivée pour l'entité 2D. Peut être défini directement lorsque "Position Vers est : valeur donnée pour X et Y", sinon est calculé par le bloc. Cette option est désactivée si "*Travel time is speed and calculated length*" est sélectionné. Dans ce cas, la position Vers est déterminée par le champ du même nom.

To Y location (position Y Vers) : les coordonnées Y de la position d'arrivée pour l'entité 2D. Peut être défini directement lorsque "Position Vers est : valeur donnée pour X et Y", sinon est calculé par le bloc. Cette option est désactivée si "*Travel time is speed and calculated length*" est sélectionné. Dans ce cas, la position Vers est déterminée par le champ du même nom.

From location is (position De est) : ce menu sélectionne la position de départ pour l'animation du transport. Les options sont :

- **Entered X and Y location** (valeur données pour X et Y) : entrez les positions X et Y dans les champs Position X De et Position Y De. Le mouvement des entités commence à ces coordonnées
- **Block location** (position du bloc) : la position du bloc est utilisée pour point de départ de l'animation

- **Enclosing hierarchical block** (bloc hiérarchique supérieur) : la position du bloc hiérarchique englobant ce bloc est utilisée pour point de départ de l'animation. Si le point d'arrivée est calculé (bloc suivant, bloc stockeur suivant), ce bloc doit être le dernier avant que l'entité ne quitte le bloc hiérarchique.

- **Previous block** (bloc précédent) : le point de départ de l'animation est le premier bloc en amont.

- **Previous non-passing block** (bloc stockeur précédent) : le point de départ de l'animation est le premier bloc stockeur ou de décision placé en amont. Cette option sautera tous les blocs passeurs entre ce bloc et le précédent bloc non passeur.

To location is (position Vers est) : ce menu sélectionne la position d'arrivée pour l'animation du transport. Les options sont :

- valeur données pour X et Y : entrez les positions X et Y dans les champs Position X Vers et Position Y Vers Le mouvement des entités se termine à ces coordonnées.

- position du bloc : la position du bloc est utilisée pour point d'arrivée de l'animation.

- bloc hiérarchique supérieur du bloc suivant : a position du bloc hiérarchique englobant ce bloc est utilisée pour point d'arrivée de l'animation. Si le point d'arrivée est calculé (bloc suivant, bloc stockeur suivant), ce bloc doit être le dernier avant que l'entité ne quitte le bloc hiérarchique.

- bloc suivant : le point d'arrivée de l'animation est le premier bloc en aval.

- bloc stockeur suivant : le point d'arrivée de l'animation est le premier bloc stockeur ou de décision placé en aval. Cette option sautera tous les blocs passeurs entre ce bloc et le bloc suivant non passeur.

2D animation shows simultaneous item movement (animation 2D montre les mouvements simultanés) : si cochée, plusieurs entités se déplaceront simultanément dans le modèle 2D. L'animation peut être soit en ligne droite, en partant de la "position De" et se terminant à la "position Vers". Soit en suivant les liens tracés entre ces deux positions. Désactivez la commande Animer les liens du menu Simulation si vous choisissez cette option, sinon l'animation bloc à bloc et l'animation simultanée tenteront de se dérouler en même temps. Si vous choisissez "*Travel time is speed and calculated length*" dans l'onglet Behavior, l'animation suivra le choix pour "Calculer distance".

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Connecteurs

En entrée l'entité à transporter.

En sortie les entités inchangées, à la fin de leur transport.

Connecteurs de sortie continus : (dans l'ordre d'apparition.)

U (Utilisation) : indique le taux d'utilisation du bloc.

L (Longueur) : indique le nombre d'entités présentes dans le bloc.

Connecteurs d'entrée continus : (dans l'ordre d'apparition.)

D (Délai) : si relié, prévaut sur le délai indiqué dans le dialogue.

Demand : si relié, le bloc n'acceptera d'entités que si la valeur sur ce connecteur est supérieure au nombre courant d'entités en cours de transport.

X start (x début) : Modifie la valeur X de départ pour la distance du transport.

Y start (y début) : Modifie la valeur Y de départ pour la distance du transport.

X end (x fin) : Modifie la valeur X d'arrivée pour la distance du transport.

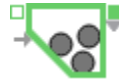
Y end (y fin) : Modifie la valeur Y d'arrivée pour la distance du transport.

Animation

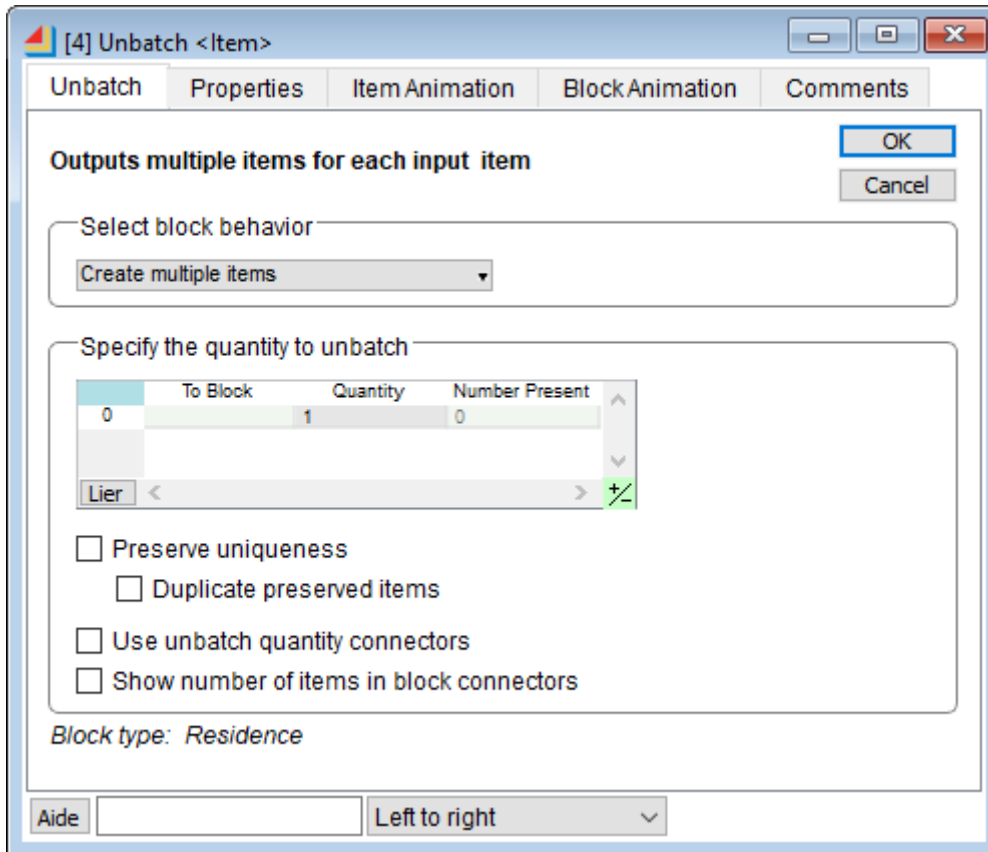
L'icône montre le nombre d'entités présentes dans le bloc.

Voir les onglets d'animation pour les spécificités d'animation du bloc.

Unbatch



Produit plusieurs entités à partir d'une entité unique en entrée.



Le nombre d'entités produites sur chaque sortie est spécifié dans le dialogue. Par défaut, le bloc retient ses sorties jusqu'à ce que celles-ci soient utilisées, ou jusqu'à ce qu'une autre demande parvienne au connecteur. Les attributs et priorités de l'entité en entrée sont transmis aux sorties. Si vous avez sélectionné *Preserve uniqueness* (*Individualiser*) dans le bloc Batch et dans ce bloc, les entités sont générées avec leurs propriétés d'origine.

Ce bloc permet par exemple de ventiler différentes copies d'une facture, de transmettre le même message à plusieurs destinataires, de lancer des process en parallèle.

Le bloc Unbatch n'a pas de capacité de stockage (au-delà de l'entité en cours) : toutes ses sorties doivent être absorbées avant que le bloc puisse accepter l'entrée suivante.

Toute valeur non-nulle de *Quantity* dans le tableau du dialogue doit avoir le connecteur correspondant relié.

Pour faire varier le nombre d'entités produites par un connecteur de sortie, vous devez cocher la case **Use unbatch quantity connectors** ('Utiliser les connecteurs pour démultiplier') pour afficher la valeur de connecteurs d'entrée sur l'icône.

Si arrive une entité qui représente un lot dont les éléments sont individualisés (entité résultante d'un Batch où "Preserve uniqueness" était coché), et que l'option "Preserve uniqueness" n'est pas coché dans le Unbatch, l'entité qui sort par le connecteur du haut contiendra l'information sur les entités mises en lot, les entités sortant sur les autres connecteurs ne contiendront aucune information sur les constituants du lot.

Onglet Unbatch

Select Block Behavior (Comportement du bloc) : cet élément ne s'applique que si le modèle calcule des coûts automatiquement. Pour cela, un paramètre de coût pour au moins un des blocs du modèle doit être supérieur à zéro. Si **create multiple items** (créer des doublons) est sélectionné, l'information de coût est copiée pour toutes les entités en sortie. Si **release cost resources** (libérer les ressources de coût) est sélectionné, le bloc va libérer les ressources par le même connecteur que celui utilisé pour constituer le lot. Les informations de coût de l'entité associée aux ressources seront mises à jour.

Table

Colonne 1 To Block (Vers bloc) : identifie le bloc auquel le connecteur associé est relié.

Colonne 2 Quantity : spécifie combien d'entités seront émises par le connecteur correspondant.

Colonne 3 Number Present (Nombre accumulé) : affiche le nombre d'entités actuellement en attente sur le connecteur de sortie correspondant.

Preserve uniqueness (Individualiser) : si l'entité a été marquée par un bloc Batch avec cette même option, le fait de délotter recrée toutes les propriétés initiales (attributs et priorités). Dans le cas contraire, le bloc crée des entités dont les propriétés sont fusionnées.

Use preserved items to determine unbatch quantities (Utiliser les quantités individualisées) : si cochée, les quantités indiquées pour délotter se sont pas utilisées, mais le bloc cherche cette information sur l'entité elle-même. Ne cochez cette case que si le bloc reçoit des entités mises en lot de manière individualisée.

Duplicate preserved items (Dupliquer les entités d'individualisation) : Si Preserve uniqueness n'est pas cochée dans le Unbatch mais avait été cochée dans le Batch correspondant, ceci copie les entités individualisées autant de fois que précisé. Le résultat est la création de nouvelles entités qui sont des doublons des entités contenues dans celle qui arrive au bloc Unbatch. Par exemple, si le Unbatch a 3 sorties avec une entité partant à chaque sortie et qu'arrive une entité résultant d'un lot de 10 (individualisées), le bloc Unbatch créera 20 entités supplémentaires, en dupliquant sur chaque sortie l'entité représentant 10 qu'il enverra sur la première sortie. Trois entités sortent, mais sont chacune doublon de l'entité contenant 10 individus.

Use unbatch quantity connectors (Utiliser les connecteurs pour démultiplier) : si cochée, active les connecteurs d'entrée pour déterminer les quantités à démultiplier pour chaque entrée.

Show number of items in block connectors (Nombre d'entités indiqué sur les connecteurs) : affiche un connecteur continu en sortie, qui reporte le nombre d'entités délottées.

Onglet Propriétés

Colonne 1 (Property) : les propriétés disponibles pour l'entité sont listées ici.

Colonne 2 (Action) : dans cette colonne vous choisissez l'action à effectuer pour chaque propriété de l'entité démultipliée. Les choix sont **Preserved value** (Preserver), **Batched value** (Valeur du lot), **Distribute** (Distribuer) et **Automatic**. **Preserved value** utilisera la valeur stockée via l'option Preserve uniqueness (ne fonctionne que si Preserve uniqueness a été utilisé lors de la mise en lot). **Batched value** donnera à chaque entité démultipliée la valeur de l'entité constituant le lot. **Distribute** divisera en parts égales la valeur entre les entités démultipliées. **Automatic** adoptera *Preserved* en mode Preserve uniqueness, et *Batched* autrement.

Onglet Item Animation

Cet onglet ne présente pas de choix global. Une table permet de spécifier les choix pour chaque file interne :

Colonne 1 (Animation Option) : spécifie le choix d'animation pour chaque file interne. **No Change** "Tel quel" pour garder l'objet initial et **Change to** "changer en" pour pouvoir spécifier de nouveaux choix.

Colonne 2 (2D picture) : spécifie le symbole 2D associé à cette valeur.

Onglet Block Animation

Connecteurs

L'entrée est l'entité à éclater.

Les connecteurs de sortie font passer les entités issues de l'entité d'origine.

Les connecteurs de sortie continus (uniquement visibles si l'option **Use unbatch quantity connectors** est sélectionnée) contrôle le nombre d'entités passant par chaque connecteur de sortie.

Animation

L'icône montre les sorties qui ont des entités. L'icône change de taille en fonction du nombre de connecteurs défini.

Workstation



Représente une station de travail pouvant faire attendre et traiter des entités, mêlant des fonctions d'un bloc Queue et d'un bloc Activity.

Dans l'onglet Behavior, vous définissez combien d'entités peuvent être traitées à la fois, et combien de temps dure le traitement.

La durée du traitement est spécifiée dans la portion **Define activity behavior**. Vous choisissez si le délai est une constante, issu du connecteur "D", la valeur d'un attribut, spécifié par une distribution ou issu d'une table de correspondance.

Si vous choisissez que le bloc doit être contrôlé par un bloc Shift, l'horaire précisera la capacité de traitement du bloc (horaire du type Nombre), ou bien des périodes de non-fonctionnement (horaire de type on/off).

Si le délai est choisi par une table de correspondance, la ligne de la table à considérer sera spécifiée par la valeur d'un attribut sur l'entité à traiter. Une fois la ligne déterminée, les valeurs sur la ligne spécifient une distribution aléatoire, choisie dans le menu déroulant **distribution**. Pour un délai particulier en fonction de la valeur de l'attribut, utilisez la distribution *Constant* du menu.

Les connecteurs et éléments de dialogue visibles s'ajustent en fonction de vos choix.

Onglet Behavior

Maximum number of items waiting (Nombre maximum d'entités en attente) : définit la capacité de la file d'attente intégrée à la station de travail.

Maximum number of items in process (Nombre maximum d'entités en traitement) : nombre maximum d'entités que peut contenir le bloc dans sa partie activité. Cette valeur détermine le seuil de blocage de l'activité.

Delay is (Le délai est) : **a constant** (une constante), **from the "D" connector** (issu du connecteur "D"), **an item's attribute value** (la valeur d'un attribut), **specified by a distribution** (spécifié par une distribution) ou **from a lookup table** (issu d'une table de correspondance). Ce choix fondamental du mode de calcul du temps de traitement détermine les options de la portion **Define activity behavior**.

Attribut : spécifie quelle valeur d'attribut doit être utilisée pour contrôler le temps de traitement. Le menu apparaît aussi lorsque le délai provient d'une table de correspondance. Dans le cas, la valeur sert à choisir une ligne dans la table.

Delay (Délai D) : affiche le délai courant. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, une unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Display string attributes (Afficher les attributs chaîne) : spécifie si les attributs chaîne dans la table de correspondance doivent être affichés sous forme de chaînes ou de valeurs d'indice.

Use block seed (Utiliser la base aléatoire du bloc) : en cochant cette case et en indiquant un chiffre pour la base, vous déterminez une base aléatoire locale pour les distributions calculées par ce bloc.

Distribution : permet de choisir une distribution dans une liste. En fonction de la distribution, les arguments appropriés s'affichent. Voir le bloc Nombre Aléatoire sur les distributions et leurs arguments.

Shift (Horaire) : cet élément n'apparaît que si le bloc est contrôlé par un horaire de type Nombre, qui prévaut sur **Maximum number of items in process** (cette option disparaît, car l'horaire définit cette contrainte).

Use shift (Utiliser l'horaire) : sélectionnez l'horaire applicable à ce bloc. L'horaire peut être soit ON/OFF, soit indiquer un nombre maximum d'entités dans le bloc.

Onglet Cost

Define processing costs (définir des coûts de traitement) : cochez cette case pour activer les informations de coût.

Processing cost (coût de traitement) : le coût ajouté à chaque entité pour chaque unité de temps passée dans l'activité. L'unité de temps utilisée pour définir ce taux doit être cohérente avec l'unité de temps utilisée dans le modèle. Le coût temporel pour chaque entité est ajouté à l'attribut "_cost" de chaque entité. Si une unité de temps non générique est sélectionnée dans le dialogue Paramètres de la simulation, une unité spécifique doit être sélectionnée pour ce paramètre. L'astérisque (*) après une unité de temps indique que c'est l'unité par défaut dans le modèle, et sera toujours la même que l'unité sélectionnée dans le dialogue Paramètres de la simulation.

Cost per item (coût par entité) : le coût ajouté à l'attribut "_cost" de chaque entité passant par l'activité. Ce coût est constant, quel que soit le temps passé dans le bloc.

Total cost (coût total) : le coût total associé à cette activité pour toutes les entités ayant passé ou séjourné dans le bloc.

Activity cost (Coût de l'activité) : les coûts d'activité accumulés par ce bloc seront affichés ici.

Calculate waiting cost (Calculer les coûts d'attente) : si sélectionné, la file accumule un coût d'attente en fonction de l'attribut "_rate" de chaque entité.

Queue cost (Coût de la file) : les coûts de file d'attente accumulés par ce bloc seront affichés ici.

Total cost (Coût total) : le coût total associé à cette activité pour toutes les entités ayant passé ou séjourné dans le bloc.

Onglet Ressource

Table

Colonne 1 (Resource Pool) : définit quelles ressources doivent être utilisées par la station. Les éléments d'un pool de ressource appelés par la station sont effectivement alloués lorsque l'entité quitte la portion Queue du bloc, et libérés lorsque l'entité quitte le bloc lui-même.

Colonne 2 (Quantity) : nombre requis pour chaque type de ressource (et pour chaque entité).

Show Resource Pool block numbers in popup (Voir numéros des blocs Resource Pool dans le menu) : si cochée, le numéro du bloc Resource Pool sera affiché avec son nom dans le menu déroulant.

Onglet Results

Length (current, average, maximum) (longueur (courante, moyenne, maxi)) : nombre (en cours, moyen, maximum) d'entités dans la partie file du bloc.

Wait (current, average, maximum) (attente (courante, moyenne, maxi)) : temps de séjour (en cours, moyen, maximum) d'une entité dans la partie file du bloc.

Utilization : Pourcentage du temps durant lequel le bloc a été utilisé sur l'ensemble de la simulation, en relation avec la capacité du bloc. Le temps d'utilisation pris en compte est le temps où le bloc a contenu des entités, qu'il s'agisse d'entités en cours de délai, ou d'entités ayant déjà subi le délai voulu mais retenues parce qu'aucun bloc ne les avait tirées. Ainsi, avec une capacité de 5, si le bloc traite 2 entités 100% du temps, l'utilisation est de .40 (40%).

Arrivals : donne le nombre d'entités reçues dans le bloc.

Departures : donne le nombre d'entités qui ont quitté le bloc.

Total cost (Coût total) : le coût total associé à cette activité pour toutes les entités ayant passé ou séjourné dans le bloc.

L (length) connector reports (Le connecteur L (longueur) indique) : le connecteur peut indiquer soit la longueur de la file, soit la longueur de l'activité, soit les deux.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Connecteurs

En entrée l'entité à traiter.

En sortie l'entité qui a terminé son traitement.

C : Capacité. prévaut sur le paramètre **Maximum number of items in process**.

Connecteurs de sortie continus : (dans l'ordre d'apparition.)

F : émet 1 lorsque le bloc est plein, sinon 0.

U : Utilisation. Voir description plus haut.

L : (Longueur) Nombre d'entités dans le bloc.

Animation

L'animation fournit de nombreuses informations sur l'état du bloc. Le bloc affiche une, deux ou trois petites horloges, pour montrer les entités en cours. Si la capacité du bloc est de un, la couleur de l'horloge reflète l'état du bloc. Dans le cas d'une capacité de trois ou plus, les trois horloges ont la signification suivante : une seule horloge indique une seule entité en traitement. Deux horloges, plusieurs entités en traitement. Trois horloges, le nombre maximum d'entités en traitement.

Si une entité est bloquée, la couleur de l'horloge représentant l'entité en traitement devient rouge.

Si le bloc est en arrêt, une croix rouge barre la portion traitement de l'icône.

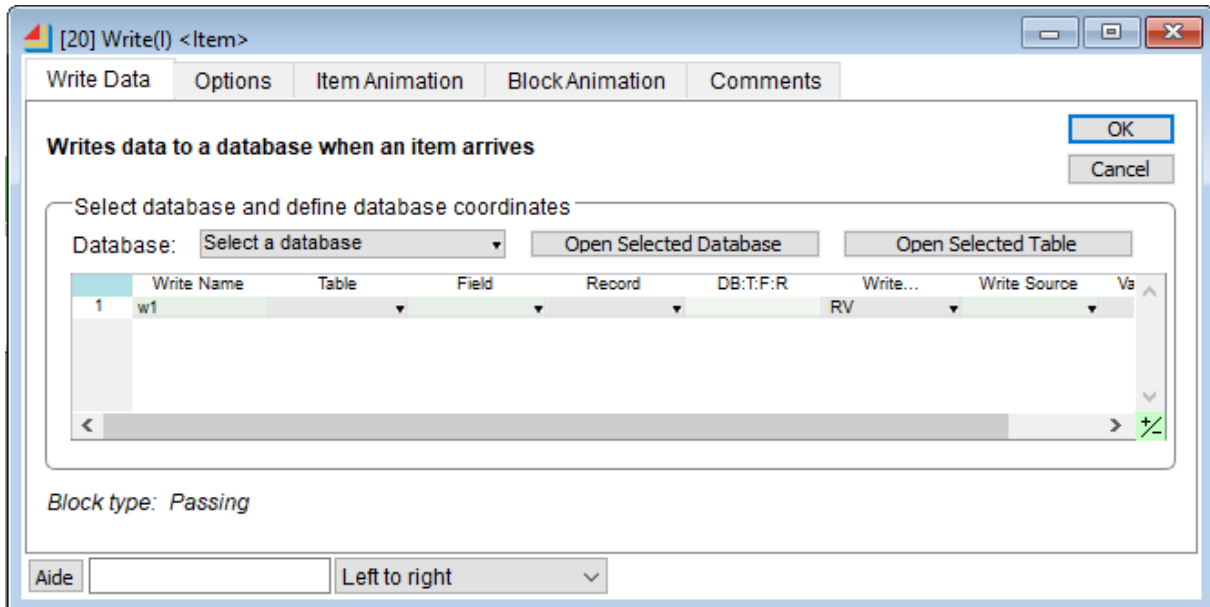
Si le bloc est en horaire inactif, une croix bleue barre la portion traitement de l'icône.

La partie file d'attente de l'animation affiche un nombre qui représente le nombre d'entités en train d'attendre.

Write(I)



Ecrit des données dans la BDD au passage d'une entité.



Vous définissez un certain nombre d'écritures à effectuer au passage de l'entité, chaque écriture remplissant une ligne de la table de données du dialogue. Dans la table vous définissez où écrire les données, en sélectionnant des options dans les menus (BDD puis table puis champ puis enregistrement).

Des connecteurs d'entrée peuvent spécifier le nom de chaque écriture, ou encore les valeurs à écrire.

Vous devez sélectionner une base de données dans le dialogue, ou par un connecteur qui change en cours de simulation.

Des connecteurs apparaîtront si l'option **Use a connector** (Utiliser un connecteur) est sélectionnée dans les colonnes Table, Field ou Record. Si les connecteurs existent, ils doivent obligatoirement être reliés.

Le nombre de lignes dans la table peut être modifié en cliquant dans la case +/- dans le coin inférieur droit de la table. Vous pouvez changer le nombre de lignes, copier ou supprimer les lignes sélectionnées.

Onglet Write Data

Database (BDD) : sélectionne la base de données à référencer.

Open selected database (Ouvrir cette BDD) : ouvre la BDD en mode Edition.

Open selected table (Ouvrir cette table) : ouvre le mode Edition pour la table sélectionnée.

Tableau

Colonne 1 Write name (Nom) : donne un nom aux données écrites par cette ligne. L'utilisateur pourra ainsi savoir de quoi il s'agit, et l'information pourra aussi être affichée sur les connecteurs du bloc.

Colonne 2 Table : sert à référencer la table. La cellule contient un menu déroulant avec les options suivantes : **Select from list** (Dans une liste), **Type an index** (Entrer un index), **Use a connector**

(Utiliser un connecteur) et **Get from attribute** (Utiliser un attribut). **Select from list** affiche une liste de tables pour y faire un choix. **Type an index** permet de saisir une valeur d'index. **Use a connector** affecte un connecteur d'entrée continu à cette ligne afin que l'index de table soit lu sur un connecteur. **Get from attribute** permet de spécifier un attribut dans lequel lire la valeur d'index de table.

Colonne 3 Field (Champ) : permet de choisir le champ cible dans un menu. Les options du menu sont similaires à celle de sélection de table.

Colonne 4 Record (Enregistrement) : ces options du menu sont similaires à celle de sélection de table, avec quatre options supplémentaires. **Use run number** (Numéro simu) indique que l'index d'enregistrement doit être le numéro de simulation + 1. S'emploie dans le cas de simulations multiples. **Last record** (**Dernier enregistrement**) utilise le dernier enregistrement de la table cible pour index d'enregistrement. **Last record + 1** trouve le nombre d'enregistrements dans la table sélectionnée, ajoute un enregistrement en fin de table, et utilise ce dernier enregistrement pour index d'enregistrement. **Item count** (Comptage entités) utilise le nombre d'entités qui ont passé par le bloc pour index d'enregistrement (l'index sera 5 si 5 entités sont passées). **Record search using attribute** (Utiliser un attribut) propose de rechercher un enregistrement dont le contenu corresponde à celui d'un attribut de l'entité de passage. Le premier enregistrement qui correspond sera retenu. **Record search using connector** (Utiliser un connecteur) fait de même, mais la correspondance est faite sur une valeur de connecteur plutôt qu'une valeur d'attribut.

Colonne 5 (DB:T:F:R) : affiche l'Adresse BDD référencée par cette ligne. Si un "x" est affiché à un emplacement de l'adresse, cela signifie que cet élément de l'adresse est variable.

Colonne 6 Write (Ecrire...) : change en fonction des choix des colonnes précédentes. Les choix sont les suivants : **record value RV** (Valeur d'enregistrement), **parent record value PRV** (Valeur d'enr. Parent) et **parent record index PRI** (Index d'enr. Parent). **record value** va simplement prendre la valeur contenue dans l'enregistrement, et implique que le champ ne soit pas enfant. Si le champ cible est un champ enfant, 2 choix sont possibles (qui reflètent la structure des champs enfants qui contiennent en fait deux informations : la valeur prise chez le parent, et l'index de l'enregistrement d'origine chez ce parent) : **parent record value** prendra la valeur telle que définie dans l'enregistrement parent (c'est l'équivalent du RV). **parent record index** va lire chez l'enfant l'index de l'enregistrement, qui doit correspondre à un enregistrement parent (≥ 1 et \leq au nb d'enregistrement de la table parent).

Colonne 7 Source : sert à sélectionner d'où vient la valeur à écrire. Les choix sont les suivants : **Type a number** (Entrer un nombre), **Use a connector** (Utiliser un connecteur) et **Get from attribute** (Utiliser un attribut). **Type a number** permet de saisir une constante. **Use a connector** définira un connecteur d'entrée. **Get from attribute** lira la valeur sur un attribut de l'entité.

Onglet Options

Expand number of records, if necessary (Augmenter le nombre d'enregistrement si nécessaire) : si la tentative d'écriture va au-delà du nombre d'enregistrements de la BDD et que cette option est cochée, la table sera agrandie automatiquement. Si l'option n'est pas cochée, cela provoque une erreur.

If writing to a Date Time field, write values received in (Données en entrée dans l'unité de temps) : une donnée de type date sera transformée dans l'unité de temps de la simulation, ou gardée dans un format de date calendaire.

Input date replaces/is added/is subtracted (Les données en entrée remplacent/s'ajoutent/se soustraient) : trois options pour l'écriture des données dans la BDD. Soit remplacer les données cible par les données en entrée, soit ajouter les données en entrée aux valeurs des données cible, soit soustraire les données en entrée des valeurs des données cible.

Treat blank write values as zeros (Considérer les valeurs vides comme égales à zéro) : si l'option choisie incrémente ou décrémente la valeur de la donnée cible, cette option permet d'assimiler une valeur vide (BLANK) comme égale à zéro.

Show input connector labels (labels connecteurs d'entrée) : si la case est cochée, tous les connecteurs d'entrée afficheront leur label.

Show input connector values (valeurs connecteurs d'entrée) : si la case est cochée, tous les connecteurs d'entrée afficheront leur valeur.

Onglet Item Animation

2D animation of item (animation 2D de l'entité) : Le menu principal de ce groupe propose : **Do not change item animation** (ne pas changer l'animation) ; **Change all items to** (changer toutes les entités en) ; **change item animation using property** (changer en fonction de l'attribut). Votre choix ici détermine l'animation 2D de chaque entité qui passe ainsi que les options qui s'affichent dans l'onglet.

Do not change item animation : les autres options sont désactivées et le symbole d'animation des entités n'est pas affecté.

Change all items to : apparaît un menu pour sélectionner le nouveau symbole d'animation pour toutes les entités quittant le bloc.

Change item animation using property : une table apparaît, permettant de spécifier des symboles d'animation en fonction de valeurs d'attribut.

2D picture (symbole 2D) : spécifie le symbole 2D de l'objet quittant ce bloc.

Tableau

Colonne 1 : **Property Value** (valeur attribut) : contient les valeurs de l'attribut utilisé pour spécifier l'animation (la table n'est visible que si l'option *change item animation using property* est sélectionnée.)

Colonne 2 : **2D Picture** (symbole 2D) : spécifie le symbole 2D associé à cette valeur.

Connecteurs

En entrée l'entité qui déclenche l'écriture dans la base de données.

En sortie la même entité.

BDD : spécifie la BDD à référencer. Prévaut sur les choix du dialogue.

Connecteur d'entrée continu : les lignes de la table qui font appel à un connecteur pour le choix de la table/champ devront être reliées. Également, les données à écrire dans la BDD si elles doivent être spécifiées par un connecteur.