

Cours de Simulation des Systèmes  
Industriels avec ARENA

Filière: SCM2

École Nationale des sciences appliquées de  
Tétouan

Abdellah El Fallahi  
Première version date de: Septembre 2018

La dernière modification date de:

18 juin 2020

## Table des matières

<b>1</b>	<b>Objectif</b>	<b>3</b>
<b>2</b>	<b>Introduction</b>	<b>3</b>
<b>3</b>	<b>Basic Panel</b>	<b>5</b>
<b>4</b>	<b>Advanced Panel</b>	<b>16</b>
<b>5</b>	<b>Transportation Panel</b>	<b>25</b>

## 1 Objectif

L'objectif principal de ce cours est de se familiariser avec le développement des modèles pour la modélisation des systèmes à états discrets, ensuite en utilisant le simulateur ARENA-SIMON on cherchera à simuler le fonctionnement de ces systèmes. Tant pour la modélisation que la simulation il va falloir commencer avec des modèles simples et réduits. Pour la simulation sous ARENA on commencera par les modules du Panel "basic Process". Aussi, il est demandé de se familiariser avec la manipulation des différentes modules de données comme les statistiques, les files d'attente, leurs attributs et fonctions, les expressions, les variables globales, ...

**Note Importante** : Vu le grand nombre de modules que ce soient graphiques ou de données disponibles sous ARENA, il est impossible d'aborder la totalité de ces modules. Par la suite, l'étudiant est donc invité à manipuler et s'entraîner en utilisant le maximum possible de ces modules dans des travaux hors classe. Juste à titre indicatif le travail hors classe doit être au moins 2 fois celui en classe.

## 2 Introduction

La simulation est un outil très utilisé pour étudier les résultats d'une action sur un élément sans réaliser l'expérience sur l'élément réel. Lorsque l'outil de simulation utilise un ordinateur on parle de simulation numérique. Généralement, le processus de simulation consiste en :

- Concevoir un modèle du système réel,
- Mener des expériences sur ce modèle,
- Étudier les différents scénarios possibles,
- Faire une analyse et interprétation des résultats obtenus par la simulation pour prendre des décisions relatives au système. Un processus de simulation peut être représenté par le schéma de la figure 1

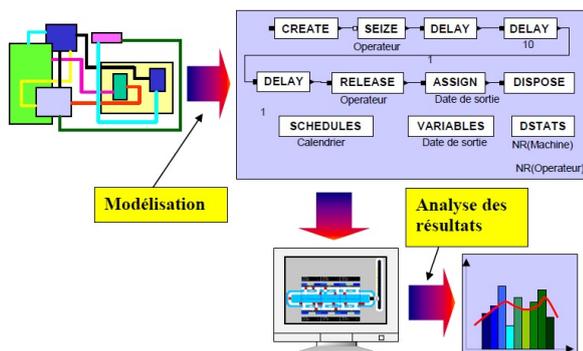


FIGURE 1 – Processus de simulation

Le but ultime de la simulation est la compréhension et la maîtrise du comportement d'un système dynamique, de comparer la performance des différentes configurations possibles pour le système en question, évaluer plusieurs scénarios et stratégies, dont le but est de booster les performances du système en question. Vu l'importance de la simulation plusieurs outils de simulation de flux ont vu le jour. La liste suivante donnera les 10 tops logiciels de simulation utilisés dans le domaine industriel.

- Extendsim
- Anylogic
- Simio
- FlexSim
- Repast
- Arena
- SIMUL8
- Promodel
- witness Horizon
- Net Logo

#### **Quel est le meilleur logiciel de simulation de flux ?**

C'est très difficile de donner une réponse bien justifiée à cette question. Comme on peut constater que la plupart des logiciels de simulation partagent tous un socle en commun. En revanche chaque logiciel à ses propres caractéristiques, avantages et inconvénients. Le choix du logiciel dépend de plusieurs paramètres à savoir :

- Nos connaissances en informatique
- Type du problème à modéliser
- Le coût d'acquisition du logiciel
- Frais de maintenance annuelle
- ...

ARENA est un système qui proportionne un environnement de travail intégré qui permet la modélisation et la simulation d'une ample classe de problème de différent champs disciplinaire. Il intègre dans un environnement très compréhensible, toutes les fonctions nécessaires pour la réussite d'une simulation d'un systèmes quelque soit sa complexité. Arena est l'interférence graphique d'un logiciel qui utilise comme langage de simulation l'outil SIMAN.

#### **Avantage du ARENA**

- C'est un outil très utilisé dans le domaine industriel,
- Son environnement de travail très facile à manipuler,
- Les fonctions qu'il contient sont faciles à utiliser,
- Une très grande capacité graphique,
- Offrir une grande polyvalence,
- Il intègre un outil d'analyse des données : input Analyser,
- Il est doté d'un système d'optimisation très puissant,
- Il est compatible avec tous les produits Microsoft Office.

#### **Inconvénients de ARENA**

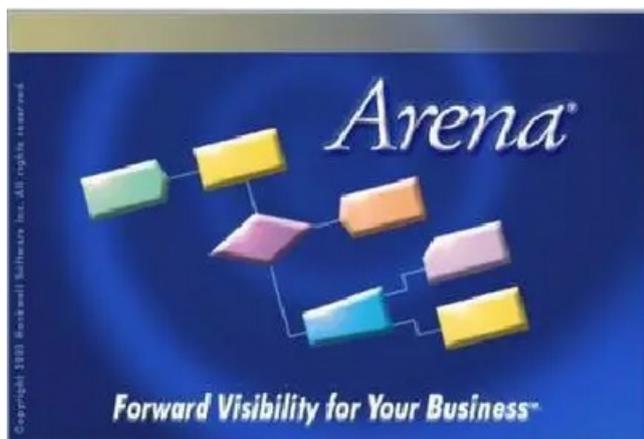


FIGURE 2 – Modèle de simulation ARENA

- Les modèles développés sous ARENA sont difficiles à traiter en utilisant d'autre outil de simulation
- Les éditions pour étudiants sont très limitées : obligation d'avoir des licences au moins éducatives
- La documentation n'est pas disponible facilement
- Problème de prise en charge des anciennes versions

**leurs types sont :**

- Time-persistent : Average, maximum and minimum selon le temps.
- Tolly : Average, maximum and minimum d'une liste de chiffres
- conteurs : la somme cumulée d'occurrence d'un événement
- Fréquences : Expression ou l'état d'une ressource.

**Sorties : Valeur final d'un élément.**

- Un événement c'est quelque chose qui se passe à un instant donné de temps.
- Ils s'organisent selon un calendrier de temps
- Les variables qui décrivent un système ne change pas entre événements.
- L'horloge de simulation c'est une variable qui enregistre le temps actuel de la simulation.

ARENA ne fixe pas le début et la fin de la simulation c'est l'utilisateur qui doit s'en chargé en fixant :

- Les conditions initiales souhaitées
- Conditions d'arrêt de la simulation

Les différents modules d'ARENA sont organiser en plusieurs Panel. Chaque Panel regroupe des fonctions, modules graphique et de data.

### 3 Basic Panel

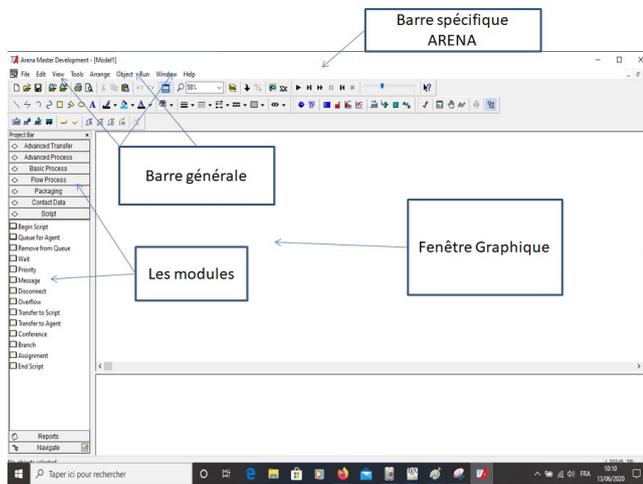


FIGURE 3 – Fenêtre du logiciel ARENA

Un modèle simple peut être représenté par trois modules ARENA à savoir : le premier module chargé de la création des entités en arrivée c'est le module create, le module process pour un traitement des entités et le module dispose pour expulser les entités du système.

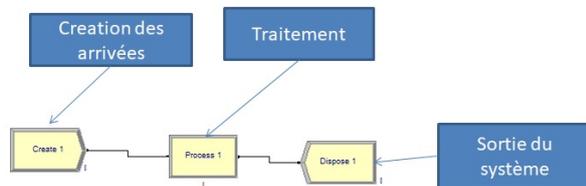


FIGURE 4 – Détail d'un modèle simple ARENA

Le module Create c'est le premier module dans n'importe quel système, les éléments principaux de ce module sont :

- Name : Fixe le nom du module
- Type : précise le type de l'entité en arrivée
- time Between arrivals : précise le temps entre deux arrivées consécutives
- Value : la valeur ou bien le paramètre de la distribution donnant le temps entre deux arrivées
- Units : l'unité de temps utilisée ( minute, heures, seconde, ...)
- Entities per arrival : donne le nombre d'entités qui arrivent ensemble, arrivée par groupe ou lot
- Max Arrivals : le nombre maximum des arrivées

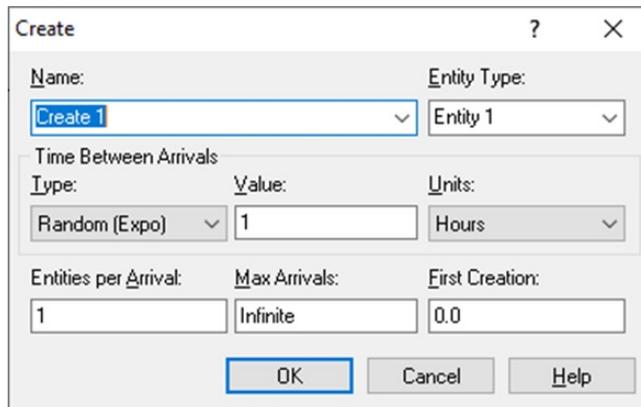


FIGURE 5 – Détail du module ARENA

— first creation : donne l’instant de la création de la première entité

## Process

le module process est sans doute l’un des plus important module dans ARENA étant donnée le module qui se charge des traitement des entités qui arrivent au système. Il dispose de plusieurs attribues et fonctions qui doivent être manipulés avec beaucoup de précaution. les principaux attributs et fonctions de se module sont :

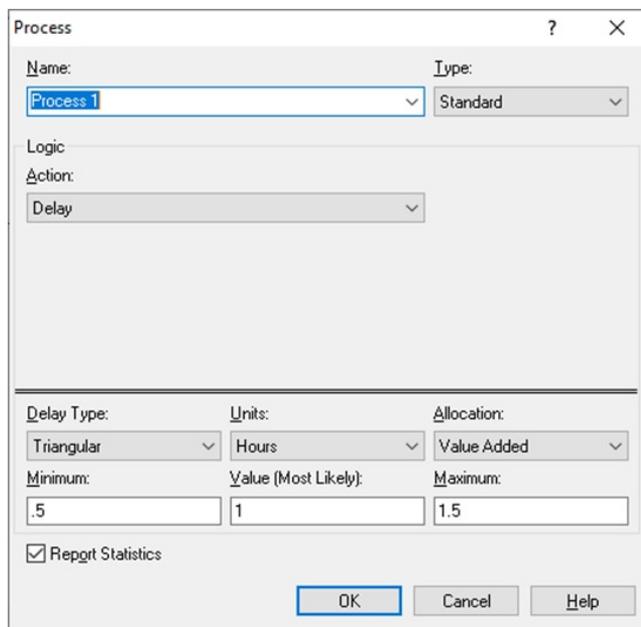


FIGURE 6 – Détail du module PROCESS ARENA

- Name : Fixe le nom du module
- Type : il peut être soit :
  - Standard : un processus simple
  - submodel : un sous modèle qui peut contenir plusieurs modules simples
- Action :
  - Delay : retard uniquement le passage de l'entité pour un certain temps donné par le delay time
  - Seize delay : réserve une ressource pour faire la tâche et retard l'entité
  - seize delay release : réserve une ressource, retard l'entité un certain temps et lorsque le temps de traitement termine en libère la ressource
  - Delay Release : retard l'entité pendant un certain temps et à la fin libère une ressource déjà réservée
- Delay type : précise le temps de service nécessaire pour le traitement d'une entité dans un processus donné.

## Decide

le module decide c'est le module chargé de l'orientation des flux dans un système. Il est souvent connecté à deux ou plusieurs voies de circulation des entités. la figure 8 donne un exemple de l'utilisation du module decide. Le flux généré par le module create sera orienté vers les processus 1 et 2 selon les conditions du module decide.

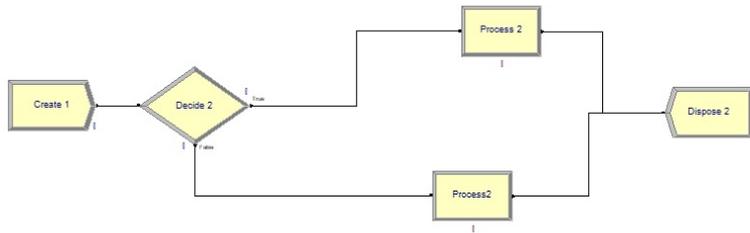


FIGURE 7 – Exemple de l'utilisation du module decide ARENA

L'orientation des flux peut être faites selon quatre conditions :

- 2 way by chance : voie 1 avec une probabilité  $\alpha$  et la deuxième voie avec une probabilité  $1 - \alpha$
- 2-way by condition : première voie avec un condition C1 et la deuxième avec la condition non C1
- N-way by chance : plusieurs choix basés sur la probabilité
- N-way by condition : plusieurs conditions C1 pour la voie 1, C2 pour la voie 2 et le reste pour la voie 3, 8

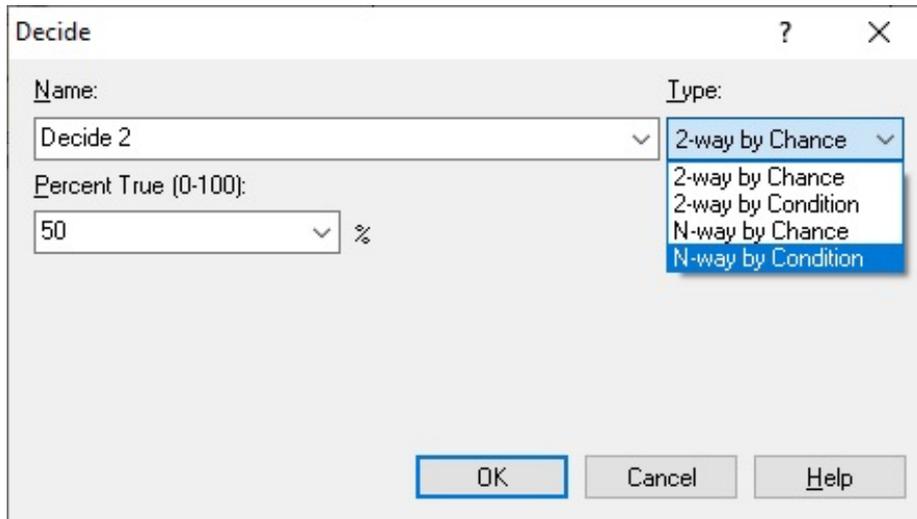


FIGURE 8 – Conditions des module decide ARENA

### Le module Assign

le module Assign permet d'affecter ou de modifier les valeurs initiales des attributs d'une entité ou bien les variables globales des modèles. La figure 9 montre un exemple simple d'utilisation de ce module, qui joue un rôle important en simulation.

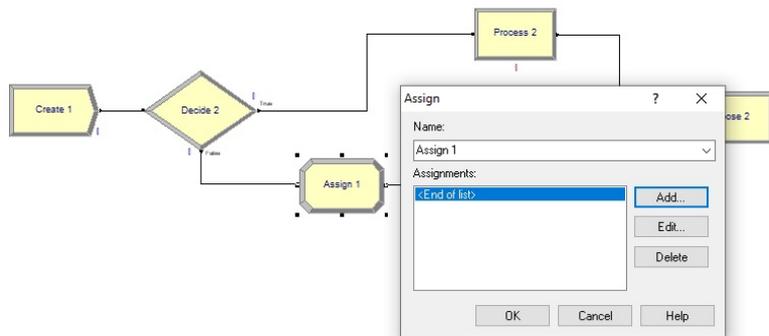


FIGURE 9 – Conditions des module decide ARENA

### Record Module

Lorsque on cherche à enregistrer des statistiques non élémentaires d'une entité ou bien d'un module on utilisera pour cela le module record comme le montre la figure 10.

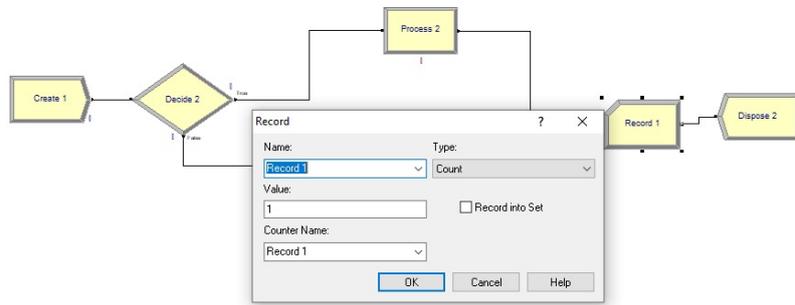


FIGURE 10 – Record module ARENA

Le module record peut enregistrer plusieurs statistiques comme :

- Count : Compter le nombre d'entités transitant par un processus
- Statistiques générales d'une entité ou bien processus
- Time interval : le temps qui passe une entité entre deux passages pas des points différents du système
- Time Between : le temps entre deux arrivées consécutives
- expression : toute expression qu'on peut établir sur le système étudié.

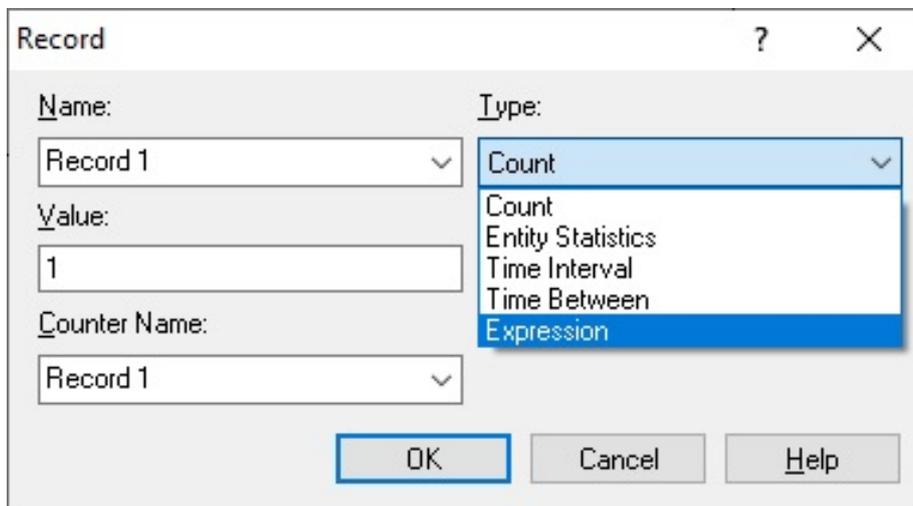


FIGURE 11 – Record Type ARENA

### Batch Module

Dans plusieurs systèmes industriels le traitement des entités doit se faire par groupe ou lot. Pour modéliser ce type de traitement ARENA dispose d'un module Batch qui permet le regroupement ou bien la consolidation traitements

entités par groupe ou bine par lot, comme le cas du transport ou bine emballage. le module batch à plusieurs attributs qui permettront la manipulation des lots d'une manière réaliste. la figure 12 présente la mode de fonctionnement du module batch.

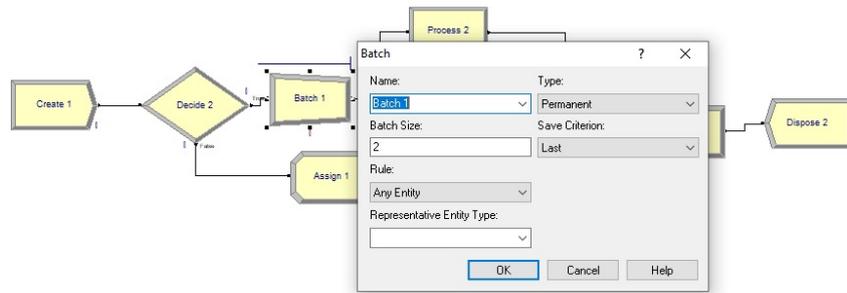


FIGURE 12 – Batch Module ARENA

## Separate Module

Pour traiter une matière première arrivée en lot ou bien on grande quantité il va falloir la diviser en plusieurs unités de produits simple. cette opération est assurée dans ARENA par le module separate. La figure 13 illustre quelques éléments de ce module, en revanche l'utilisation est invité tester l'ensemble des attributs pour une maîtrise du fonctionnement de ce module.

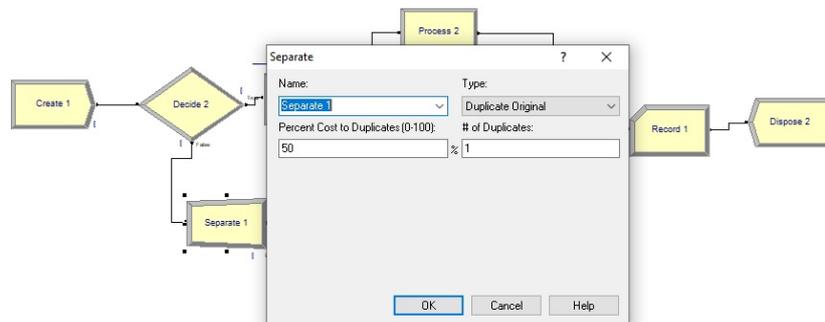


FIGURE 13 – separate Module ARENA

le reste des modules du Basic panel sont donnés par la figure 14.

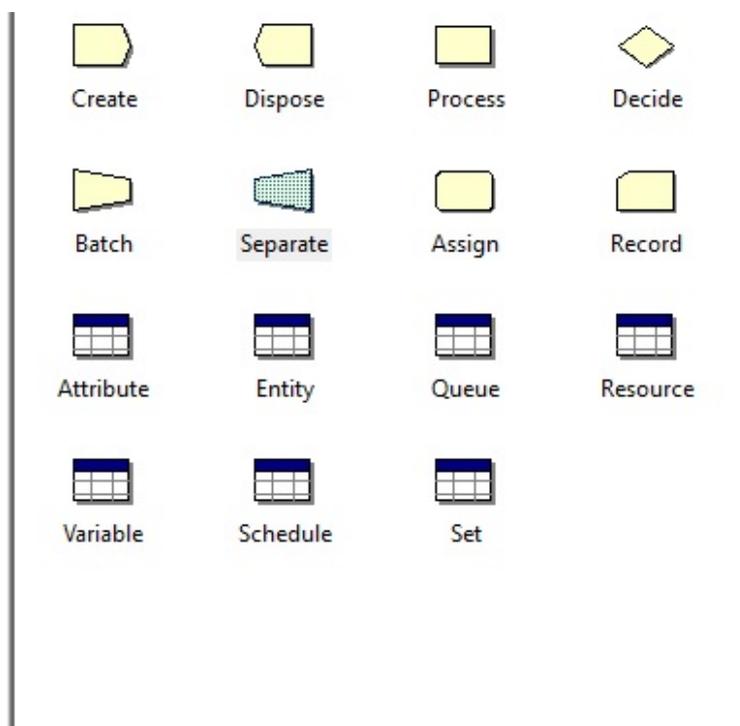


FIGURE 14 – Basic modules ARENA

### Exercice 1 : Guichet Automatique

**Information :** les clients arrivent à un guichet automatique selon les information suivantes :

- La taux d'arrivée est  $\text{expo}(5)$  minutes
- les clients arrivent ne groupe de 2,3 et 4 personnes avec les probabilités de 40%, 30% et 30% à l'agence bancaire.
- 15% des clients (type 1) ont la priorité maximale d'attention, le 38% des clients (type 2) ont une priorité intermédiaire et le reste des clients (type 3) ont une priorité basse d'attention.
- Les clients de type 1 et 2 vont être reçu par les guichets 1,2 et 3 et les clients de type 3 sont reçu dans les guichet 2 et 4.
- les temps de services requiet sont :  $\text{expo}(8)$ ,  $\text{expo}(5)$  et  $\text{expo}(3)$ , pour les clients de type 1, 2 et 3 respectivement.
- la guichet 3 travaille initialement 3 heures, ensuite il prend une pause d'une heure ensuite il reprend le travail jusqu'à la fin
- le guichet 4 travaille 4 heures et se retire.

**travail demandé :**

1. Donner un modèle pour la simulation de 200 clients
2. donner les statistiques de bases pour ce système

## Résolution

le premier module à créer est create qui permet de générer les arrivées des clients selon la figure 15.

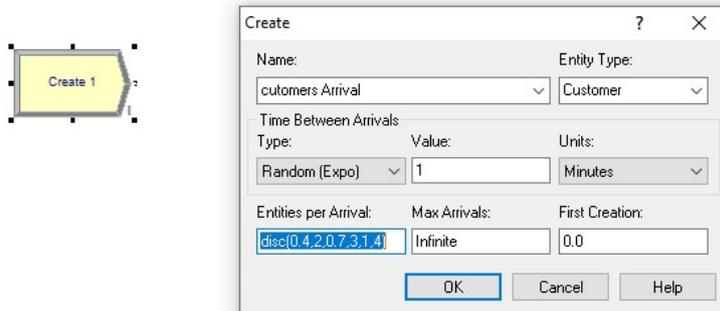


FIGURE 15 – Create guichet automatique

Pour affecter le type des clients qui arrivent alors on peut utiliser le module assigne comme le montre la figure 16.

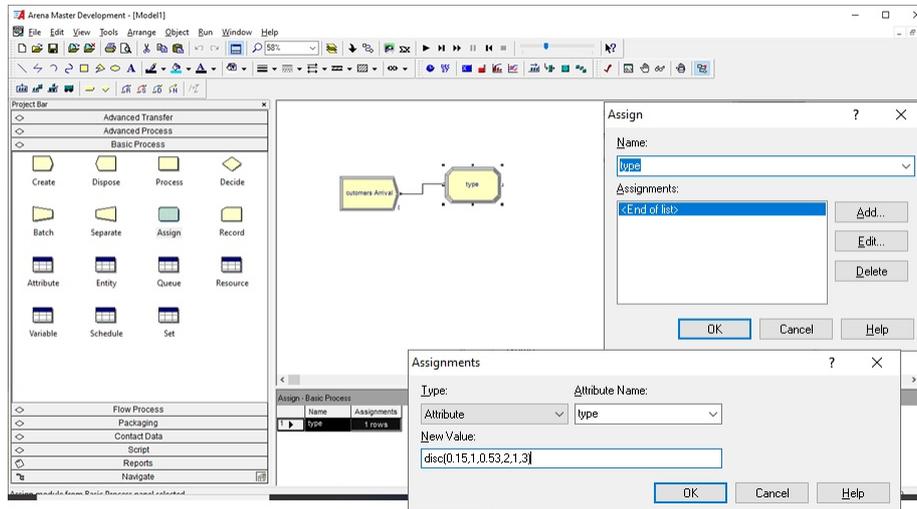


FIGURE 16 – assigne guichet automatique

de la même manière le même processus assign peut être utilisé pour affecter les guichets aux clients. En suite il faut utiliser le module decide pour orienter

les clients vers les guichets. selon le modèle de la figure 17. le modèle global pour

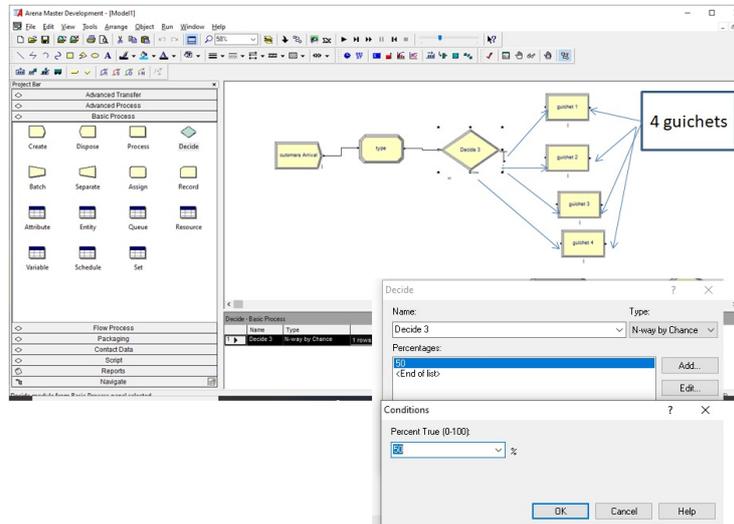


FIGURE 17 – Decide guichet automatique

ce petit système peut être donné par 18.

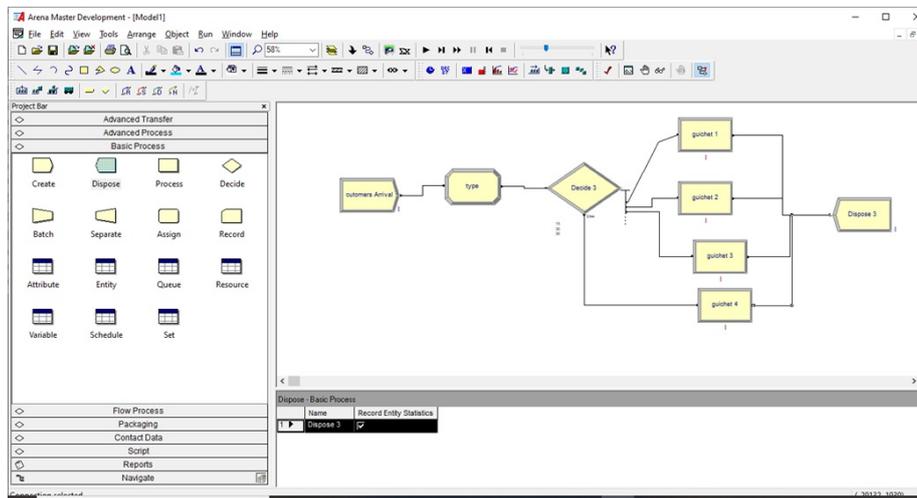


FIGURE 18 – modèle global pour le guichet automatique

## 4 Advanced Panel

Le Panel Advanced process améliore et enrichi les modules présentés par le Basic Panel 19. En effet, plusieurs modules comme le process ont été divisés en plusieurs sous-modules simples. Cette division permet d'optimiser et de booster la simulation. Par exemple le module process qui peut avoir une utilisation polyvalente comme la réservation des ressources, le retardement des entités et la libération des ressources, a été remplacé par trois modules chacun est chargé d'une seule tâche. donc on peut dire que process dans basic process peut être remplacé par seize, delay, release dans advanced process, comme le montre la figure 20. la figure 21 montre les différentes étapes pour la réservation d'une

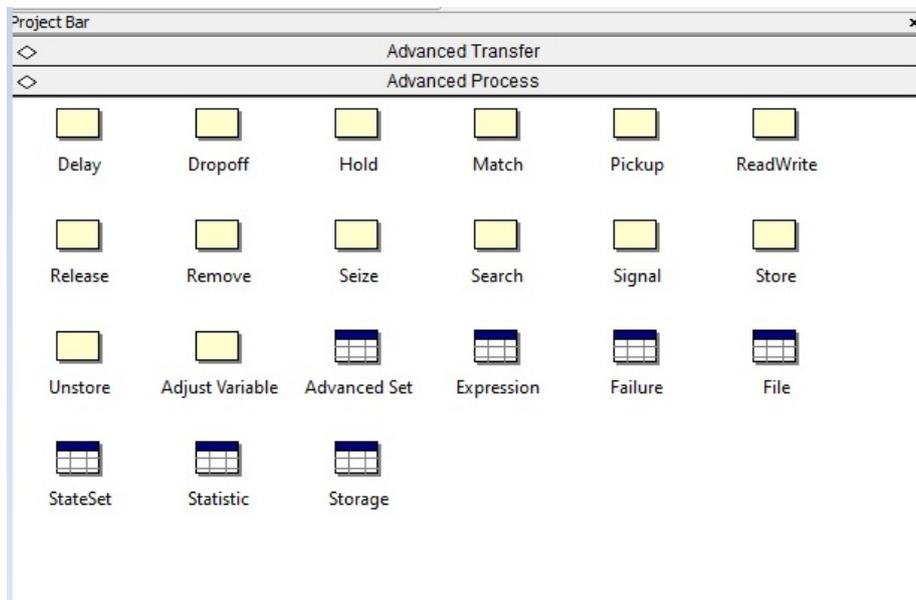


FIGURE 19 – Modules du Panel advanced process

ressources, ou bien ensemble de ressources, le retardement de l'entité et ensuite la libération d'une ou plusieurs ressources précédemment réservées.

### Exemple d'utilisation des modules du Advanced Panel

on considéré un système à file d'attente simple M/M/1. Les véhicule arrive à une station de lavage selon loi de poisson de paramètre 1 voiture tout les 50 minutes, le temps de lavage est estimé à 40 minutes. L'opération de lavage est effectuée par un opérateur, la station dispose de deux opérateurs. On cherche la simulation de ce système pendant une durée de 8 heures. Contrairement à ce qu'on a fait jusqu'à maintenant en travaillant avec une seule ressource, dans ce cas nous avons deux opérateurs et par la suite on parle de notion de set ou ensemble. Donc le premier travail à faire est de modéliser le problème comme le montre la figure 22 En cas d'un ensemble de ressources la réservation d'une ou plusieurs ressource se fait selon le modèle de la figure 23. Donc il faut définir le

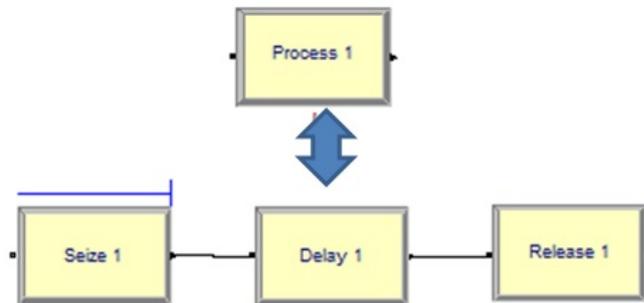


FIGURE 20 – décomposition du module process en advanced panel

set avant son réservation, les étapes à suivre pour définir un set sont montrées dans la figure refsetdefinition.

un ensemble de ressources est défini par :

- Set Name : le nom de l'ensemble
- Type : le type i peut être :
  - ressource
  - Tally
  - Counter
  - Entity Type
  - Entity Picture
- Members : donne les noms des membres constituant le set.

le modèle global peut être donnée par la figure 26.

Dans ce qui suit on donnera une description détaillée de l'ensemble des modules de ce Panel.

#### **Delay :**

Le module Delay retarde une entité pour une quantité de temps spécifique. Quand une entité arrive à un module Delay, on évalue le temps de retard et l'entité reste dans ce module durant le temps spécifié. Trois types de retard on peut avoir, no added value, added value, transfert et wait. On peut aussi calculer les coûts associés à ce retard et on les affecte à l'entité.

#### **Applications Typiques :**

- Traitement d'un chèque dans une banque
- Réaliser un temps de Setup à une machine
- Transférer un document à un autre département

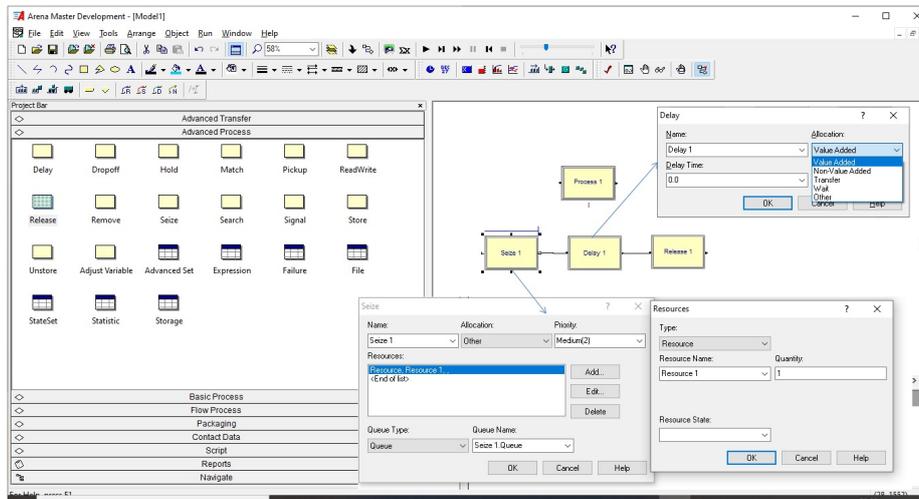


FIGURE 21 – détail des trois modules seize, Delay et Release

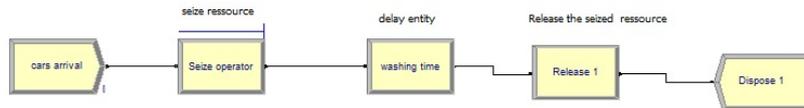


FIGURE 22 – Waching Cars model

**Dropoff :**

Le module DropOff prend un numéro spécifique d'entité d'un module et l'envoi à un autre, selon ce qui est spécifié par une connexion graphique.

**Application Typiques :**

- chargés des étagères avec des produits
- Séparer une forme pour utilisation dans différents départements

**HOLD :**

Le module HOLD retient une entité dans une file d'attente en attendant un signal, une condition. Si l'entité est retenue pour un signal, le module qui donne le signal s'utilise dans une partie du modèle pour autoriser le mouvement de l'entité au module suivant. si l'entité est retenue en attendant une condition donnée, l'entité reste dans le file d'attente du module HOLD jusqu'à la vérification de la condition.

**Application Typiques :**

- Attendre le passage d'un feu de circulation jusqu'à son passage en vert
- tester l'état d'une machine ou d'un opérateur pour continuer le processus
- Retenir l'entrée des clients jusqu'à ce que une place se libère

**Match :**

Le module Match réunit un numéro spécifique d'entités qui attendent dans différentes files d'attente. Le module Match peut se réaliser quand on a au moins une entité dans chaque file d'attente de celles spécifiées. Quand une entité arrive à un module match elle entre dans une file d'attente selon le point d'entrée avec lequel elle est connectée. Les entités restent dans les files d'attente jusqu'à la vérification de la condition du Match. Une fois la condition du match vérifiée une entité de chaque file d'attente sort pour être matché avec les autres. Les entités sortant du match se synchronisent pour sortir du module.

**Application Typiques :**

- prélèvement de plusieurs produits pour une demande
- Synchroniser la sortie d'un patient avec le paiement d'une facture

**Pickup :**

Le module Pickup enlève un nombre d'entités consécutives d'une file d'attente. Les entités enlevées s'ajoutent à la fin du groupe des entités entrantes.

**Application Typiques :**

- Prélever une demande de plusieurs files localisées.
- Prélever des formulaires remplis pour un service

**Readwrite :**

le module Readwrite s'utilise pour lire les données à partir d'un fichier d'entrée ou bien du clavier et affecte les valeurs des données à une liste de variables ou attributs ( ou toute autre expression). Ce module aussi s'utilise pour écrire les données dans un dispositif de sortie, tel que l'écran ou bien fichier. Quand une entité arrive à un module Readwrite, le fichier spécifié sera examiné pour savoir s'il est ouvert ( active). sinon, l'archive s'ouvre automatiquement.

**Application Typiques :**

- lecture en ligne d'une information( arrivées ou sorties)
- Écrire des informations dans un fichier de données, tel que l'arrivée d'une commande, temps d'arrivée, ou bien temps de finalisation d'une tâche

**Release :**

Le module Release s'utilise pour libérer des ressources réservées précédemment. Pour chaque ressource à libérer il faut préciser la quantité ainsi que le nom.

**Application Typiques :**

- Terminer une demande client et libérer l'opérateur
- Sortir d'un hôpital et libère un lit.

**Remove :**

le Module Remove permet d'enlever une entité d'une file d'attente et l'envoie vers un module spécifique. El ranking de l'entité signifie la localisation de l'entité dans la file d'attente

**Application Typiques :**

- Appelle un patient d'un chambre qui attend pour un examen
- faire entrée une personne âgée attendant dans une file d'attente

**Seize :**

Le module Seize s'utilise pour réserver des unités d'une ressource donnée, ou des membres d'un ensemble de ressources ou bien une ressources selon l'ordre choisi pour la méthode alternative, tel que la quantité ou bien une expression.

**Application Typiques :**

- Commencer la préparation de la commande d'un client
- Commencer une tâche sur une machine
- commencer le traitement d'un patient

**Signal :**

Quand une entité arrive à un module signal, on évalue le signal et on envoie le code du signal. Dans ce temps, les entités dans les modules d'attente qui attendent ce signal s'enlève des files d'attentes. L'entité qui envoie le signale continue le processus jusqu'à ce qu'elle rencontre un Delay ou bien sort des système :

**Application Typiques :**

- Signale quand le feu passe en vert
- signale quand le traitement d'une tâche est terminé

**Store :**

Le module STORE ajoute une entité à l'entrepôt. Les stockages sont très utiles pour l'animation de l'entité tant que celle-ci se trouve dans ces modules.

**Application Typiques :**

Animation d'une entité avec un numéro de retard dans les opérations ( chargement, déchargement, traitement)

**Unstore :**

Le module Unstore enlève une entité du stock. quand une entité arrive au module Unstore, le stock précisé se perd une entité.

**Application Typiques :**  
enlevé un patient du total des patient hospitalisés dans un hôpital.

## Modules des données

**Advanced set :**  
Le module Advanced set spécifié les systèmes des files d'attente, les systèmes des stockages, et autre systèmes et ses membres respectives. Un système définit un groupe d'élément similaires qu'ont peut spécifié par un nom en commun ou bien un indice du système. Les éléments qui constitue le set se dénomment membres.

**Application Typiques :**  
Plusieurs files d'attente pour acheter dans un supermarché ( files d'attente)

**Expression :**  
Le module EXpression permet la définition de toutes les expressions qu'on veut utiliser ainsi que leurs valeurs. Les expression peuvent être simple ou bien de deux dimensions. \_ Une valeur de l'expression peut être une combinaison de chiffres, noms des symboles, des distributions statistiques (Norm(10,3), expo(5), ect )

**Application Typiques :**

- temps de traitement par type de produit
- temps de traitement par opérateur

**Failure :** \_ Le module Failure s'utilise avec les ressources. quand une défaillance arrive,le dispositif tombe en panne. l

**Application Typiques :**  
arrêt d'une machine.

**File :**  
Le module File doit être inclut une fois que les archives externes sont demandés pour être utiliser par le module readwrite. Ce module identifie le nom de l'archive et définit le mode d'accès, le format, et les caractéristiques opérationnelles des archives.

**Application Typiques :**

- L'archive prédéfinie des données d'un vol d'une compagnie aérienne
- Archive qui spécifie les temps et information importante

**Stateset :**  
Le module Staeset s'utilise pour définir les états d'une ressource ou un set de ressource. les états peuvent être associés à un auto-état ou bien de nouveaux états de la ressource.

**Application Typiques :**

- les états d'un opérateur, tel que occupé avec un client
- les états d'une machine, tel que disponible, en réparation, en arrêt

**STATISTIC set :** Le module Statistic Set s'utilise pour définir la statistique additionnelle qu'on cherche à avoir durant la simulation et aussi il précise les fichiers de sortie.

**Application Typiques :**

- collecter des données pour comparer deux configuration différentes
- suivre le nombre de clients dans un entrepôt en utilisant la fonction NSTO et les modules se STORE/Unstore

**Storage :**

Le module Storage définit le nom d'un entrepôt. Les entrepôts sont créés automatiquement pour chaque module qui fait référence à l'entrepôt.

**Application Typiques :**

définir un entrepôt animé pour un système d'entreposage.

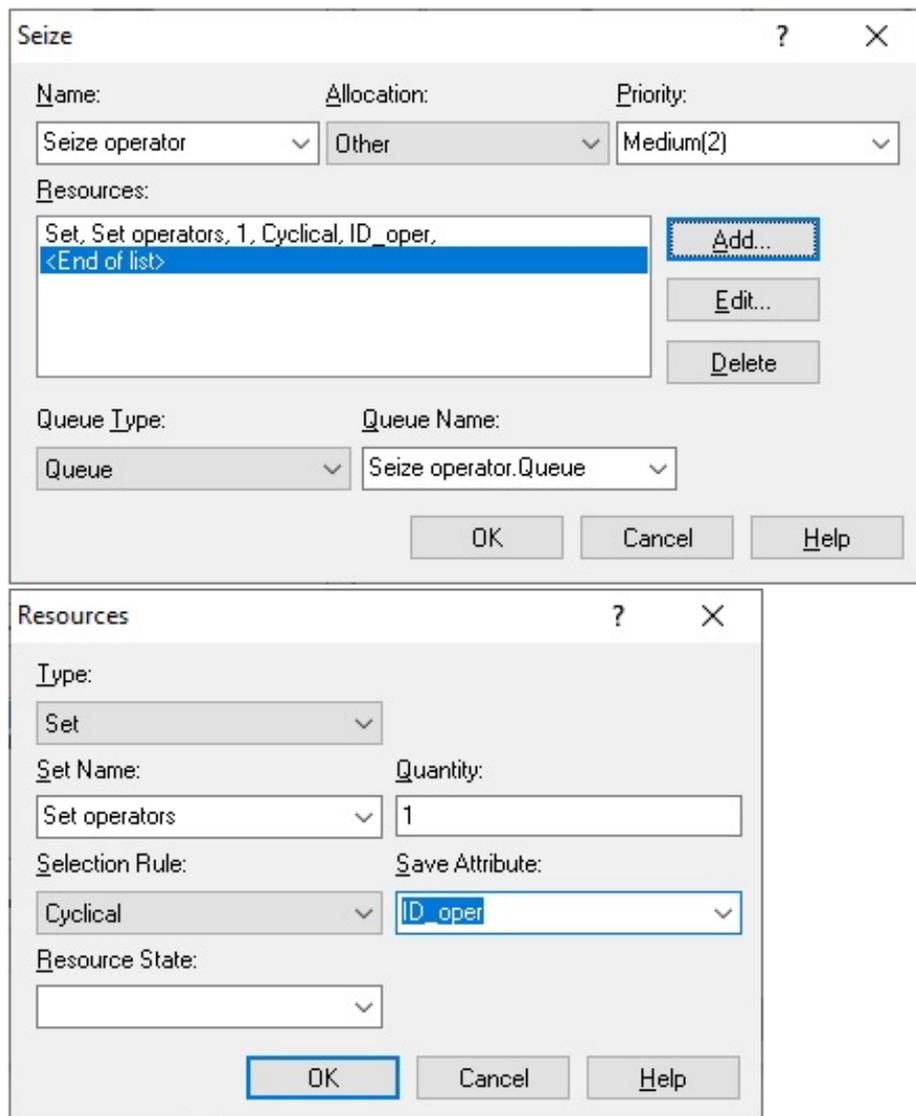


FIGURE 23 – Seize set



FIGURE 24 – définition d'un ensemble de ressources

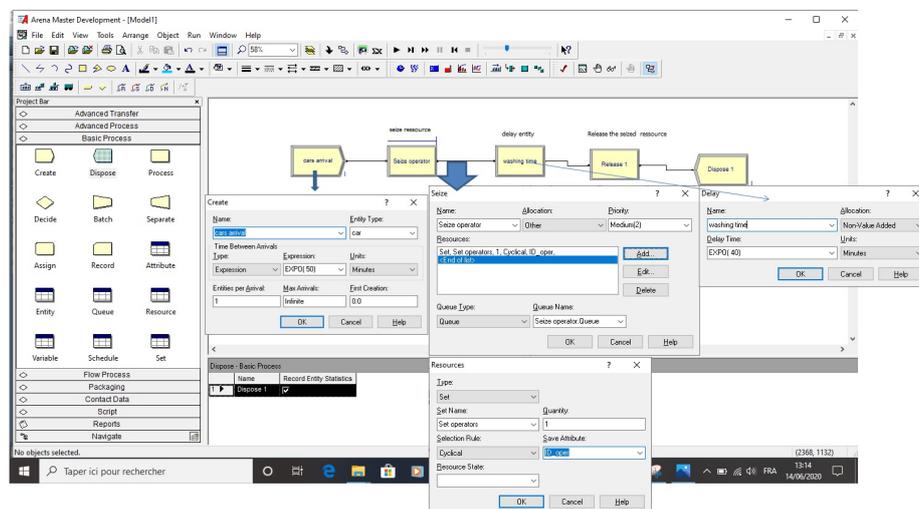


FIGURE 25 – Détail du odèle global

## 5 Transportation Panel

Le Panel Advanced Transfer contient l'ensemble de modules nécessaires pour la simulation des systèmes de transfert comme le montre la figure ??.

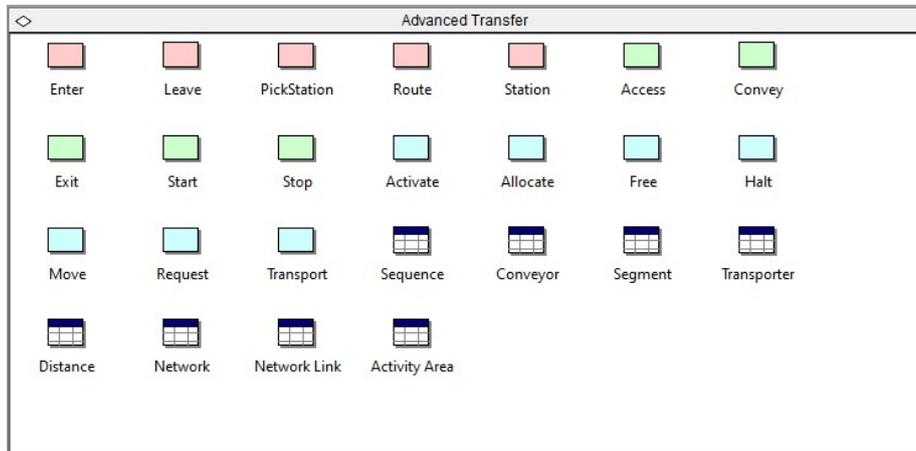


FIGURE 26 – l'ensemble de modules du Panel Advanced Transfert

**Enter :** Le module ENTER définit une station ( ou un système de stations) qui correspond à une localisation physique ou logique où se passe le processus. Quand une entité arrive à une station ENTER, il provoque un retard dans le déchargement et dans quel n'importe quel dispositif de transfert utilisé pour transférer l'entité à la station du module ENTER.

**Applications typiques :** dans un système de production, où une chaîne transporteuse arrive avec les produits des différentes stations avant ENTER.

**Leave :** le module LEAVE on utilise pour transférer une entité à une station ou un module. Quand une entité arrive à une station LEAVE, elle attend pour avoir un dispositif de transfert libre ( ressource, transporteur ou bien un convoyeur).

**Applications Typique :** dans un système de production, où une chaîne transporteuse part avec les produits aux différentes stations après le LEAVE.

**PickStation :**

Le module PickStation permet qu'une entité peut sélectionner une station en particulier parmi l'ensemble des stations possible.

**Applications typiques :**

Un client sélectionne entre plusieurs files d'attente devant les guichets celle qui a le minimum possible de clients en attente.

**Route :**

Le module ROUTE permet le transfert d'une entité à une station spécifique, ou bien la station suivante dans une séquence de stations. Le temps de voyage pour d'une station de départ vers la station d'arrivée doit être définie à ce niveau.

**Applications typiques :**

Envoie d'un produit à une station donnée dans un temps prédéfinie.

**Station :**

Le module Station définit une station ( ou bien un système de stations) correspond à une localisation physique ou logique où se passe un processus.

**Applications typiques :**

Définition d'une zone pour la préparation des commandes par exemple ou bien une station de carburant.

**Access :**

Le module Access Permet l'affectation d'une ou plusieurs cellules de transport à une entité ( produits) pour le mouvement d'une station à autre.

**Applications Typiques :**

- Produits avec accès à un transporteur qu'on enverra à une cabine de peinture
- Un cristal qu'à accès à un transporteur qu'on transfère vers une station de coupe

**CONVEY :**

Le module CONVEY transfère une entité dans un transporteur depuis son actuelle station à son prochaine station. Le temps de transfert pour le transporter l'entité à partir d'une station à la station suivante se base sur la vitesse du transporteur ( doit être spécifier dans le module CONVEY) aussi la distance entre les stations doit être précise dans le module Segment.

**Applications Typique :**

- Transporter des produits finis à la zone d'emballage.
- Transporter la matière première entre le stock initial et la zone de production

**Exit :**

Le module EXIT lance les entités dans un transporteur spécifique

**Applications typiques :**

Des voyageurs qui récupère l'équipage d'un transporteur.

**The Start :**

Le module Start change l'état d'un transporteur de inactif à actif.

**Applications Typiques :**

lance le transporteur après le programme de maintenance.

**Stop :**

Le module STOP fixe l'état d'un transporteur en inactif. quand une entité entre dans un module STOP, le transporteur s'arrête immédiatement et le transporteur et le numéro des entités actuellement dans le transporteur.

**Applications typiques :** arrêter un transporteur pour une révision programmée.

**Activate :**

Le module activate augmente la capacité d'un transporteur préalablement arrêté.

**Application typiques :** Activation d'une bande transporteuse ou bien une courroie de transmission ou bien un convoyeur.

**Allocate :**

Le module Allocate permet l'affectation d'un transporteur à une entité hors le mouvement de celle-ci.

**Applications Typique :**

affecter un taxi pour récupérer les voyageurs en attente.

**Free :**

le module FREE lance l'entité qui a été récemment dans le transporteur, en laissant libre le transporteur et de cette manière il peut être utiliser une deuxième fois.

**Application Typique :**

Un autobus termine sa tournée.

**Halt :**

la module HALT permet de changer l'état d'un entité de transport à inactif. Une fois une entité de transport est arrêtée, aucune entité ne peut avoir le contrôle du transporteur jusqu'à ce qu'il devient actif.

**Application typiques :**

l'arrêt d'un chariot pour des opérations de maintenance.

**Move :**

Le module Move avance un moyen de transport d'une station à autre sans entité. L'entité reste dans son station actuelle jusqu'à ce que le transporteur arrive à sa destination.

**Application Typiques :**

- Le mouvement d'un opérateur d'un poste à autre ( voyage sans produit)
- passage d'un serveur à la cuisine

**Request :**

Le module Request affecte une entité de transport à une entité entrante et fait bouger l'entité de transport à la localisation de l'entité à transporter. Quand une entité arrive à un module REQUEST on lui affecte un transporteur disponible sinon elle va attendre.

**Applications typiques :**

Des clients d'un restaurant qui sont prêts pour demander un serveur.

**Transport :**

Le module Transport permet le transfert de l'entité qui contrôle ainsi que l'unité de transport d'une station à autre.

le retard pour bouger l'entité et le transporteur à partir d'une station à l'autre est basé sur le vitesse du transporteur ( cette vitesse doit être préciser dans le module Transporter) et la distance entre les deux stations ( doit être spécifier dans le module Distance).

**Applications typiques :** // un camion de la poste transporte des paquets depuis la zone de ramassage des paquets à la station de traitement des paquets.

**Modules de données**

**Sequence :**

Le module Sequence on utilise pour définir une séquence de l'entité qui traverse le système. Une séquence consiste en une liste de stations que l'entité va visiter.

**Applications Typique :**

- Définir le trajet du chemin qui va traverser les entités
- Définir la séquence des étapes que les patients d'un hôpital doivent suivre.

**Conveyor :**

Le module CONVEYOR permet la définition d'un transporteur que accumule ou non pour le mouvement de l'entité entre les stations. Quand on utilise un transporteur pour transférer entre les modules, le transporteur doit être défini en utilisant ce module.

**Application Typiques :** transporteur de bouteilles dans un usine d'eau.

**Segment :**

Le module SEGMENT défini la distance entre deux stations dans le système du segment d'un transporteur de type conveyor.

**Applications Typique :**

On défini la distance que réalise une bande transporteuse entre une station et autre, exemple contrôle de qualité et emballage.

**Transporter :**

le module Transporter permet la définition d'un dispositif de transport pour le mouvement de l'entité depuis une station à autre. Dans ce module on définit la vitesse et quantité à transporté.

**Application Typiques :**

Vitesse et quantité des chaises roulante qu'on bouge entre les zones de diagnostic.

**Distance :**

ce Module s'utilise pour définir les parcours ente toutes les stations que les transporteurs de libre-trajet ( free path) peuvent visiter. Dans ce module on définit les distances entre deux stations.

**Exercice d'application**

Dans un atelier de production on traite deux type de produit PA et PB. L'atelier est constitué de trois machines :

- Milling Machine
- Turning Machine
- Painting Machine

Chaque produit suit sa propre séquence de passage par les trois machine, les temps de traitement de chaque produit par chaque machine est différent de l'autre produit. le tableau suivant donne les séquence de passage ainsi que le temps de traitement de chaque produit sur chaque machine, tous les temps sont en minutes. Les produit arrivent à l'atelier selon une loi de poisson de moyenne 30 arrivées par heure. Le transfert des produits entre les stations durera 2 minutes.

Produit	sequence	milling time	turning time	pinting time
Part A	milling - painting - turning	8	5	10
Part B	turning - milling - painting	8	12	6

**Travail à faire :**

On cherche à modéliser et simuler le travail de cet atelier pendant une durée de 8 heures.

**éléments de la solution**

- Il faut commencer par la création des séquences
- Il faut créer l'arrivée des produits
- Affecter à chaque produit entrant une étiquette qui porte les informations suivantes, milling time, turning time, painting time et la séquence de passage par les machine.

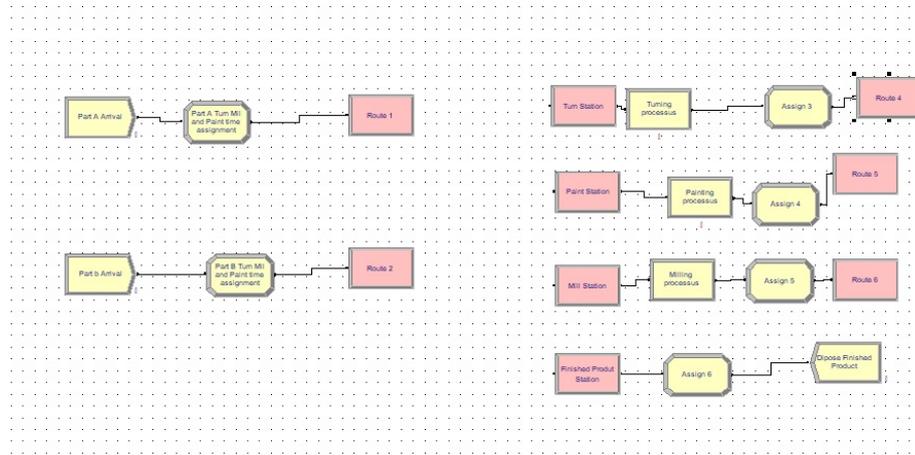


FIGURE 27 – model Global

Le modèle global de simulation peut être présenté comme dans la figure 27. le processus d'arrivée est peut être présenter par la figure 28. En arrivant les deux produits passeront par le module assign qui permet d'attribuer la séquence, le temps de milling, le temps de turning et le temps de painting par produit. Ce processus est modéliser comme dans la figure 29. les processus de traitement sont presque les mêmes sauf en ce qui concerne le temps de traitement de chaque produit. Le milling par exemple peut être modéliser comme le montre la figure 30. le transport entre les stations est assuré par l'utilisation des routes dont la modélisation est donnée par la figure ?? à la fin l'animation de ce modèle peut se faire comme le montre la figure 32.

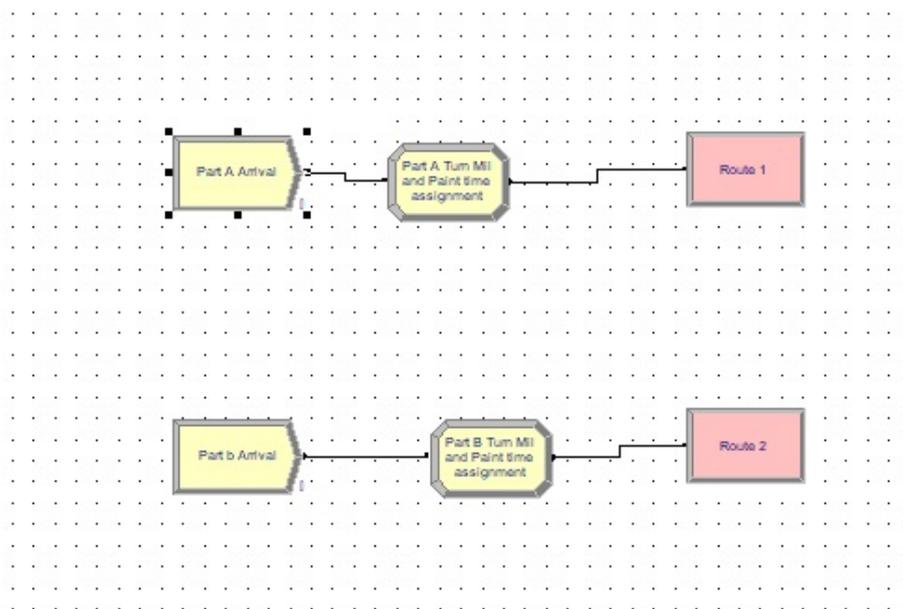


FIGURE 28 – Processus des arrivées

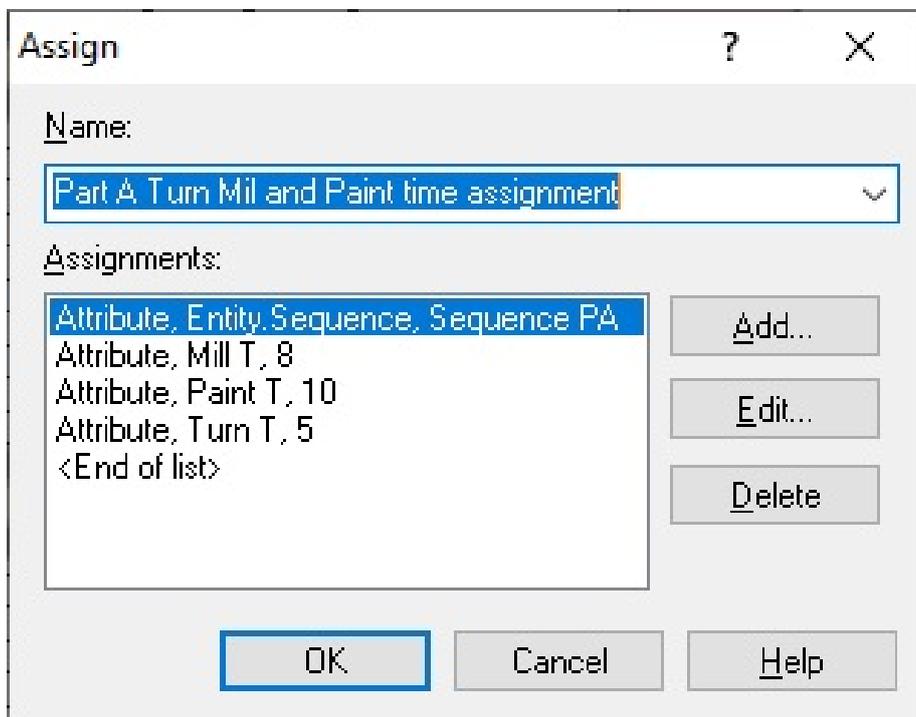


FIGURE 29 – Affectation des attributs aux produits en arrivée

The screenshot shows the 'Process' dialog box in Arena software. The dialog is titled 'Process' and has a question mark icon and a close button (X) in the top right corner. The 'Name' field is set to 'Turning processus' and the 'Type' is 'Standard'. The 'Logic' section shows 'Action' as 'Seize Delay Release' and 'Priority' as 'Medium(2)'. The 'Resources' list contains 'Resource, Turn Machine, 1' and '<End of list>', with 'Add...', 'Edit...', and 'Delete' buttons. The 'Delay Type' is 'Expression', 'Units' is 'Minutes', and 'Allocation' is 'Value Added'. The 'Expression' field contains 'expo(Turn T)'. The 'Report Statistics' checkbox is checked. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

FIGURE 30 – traitement sur la machine milling

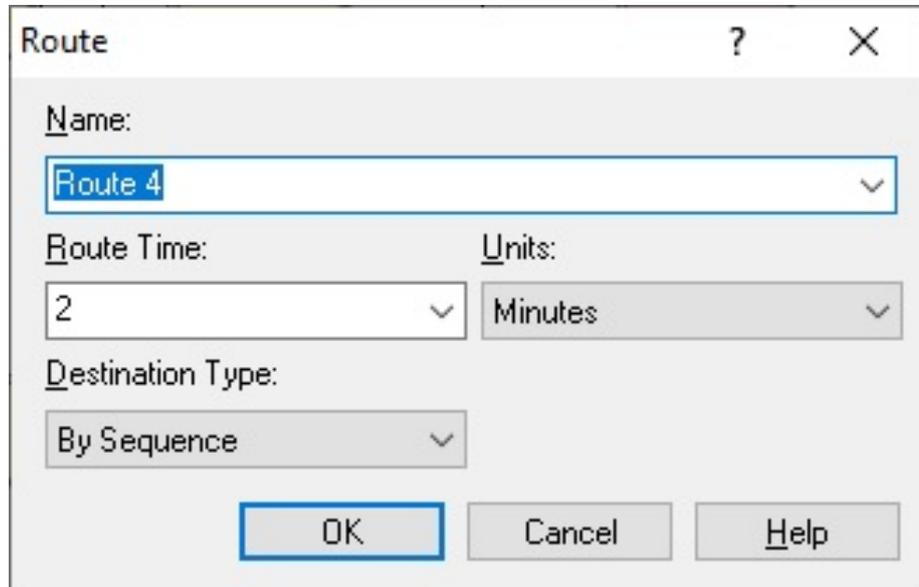


FIGURE 31 – Réservation des routes

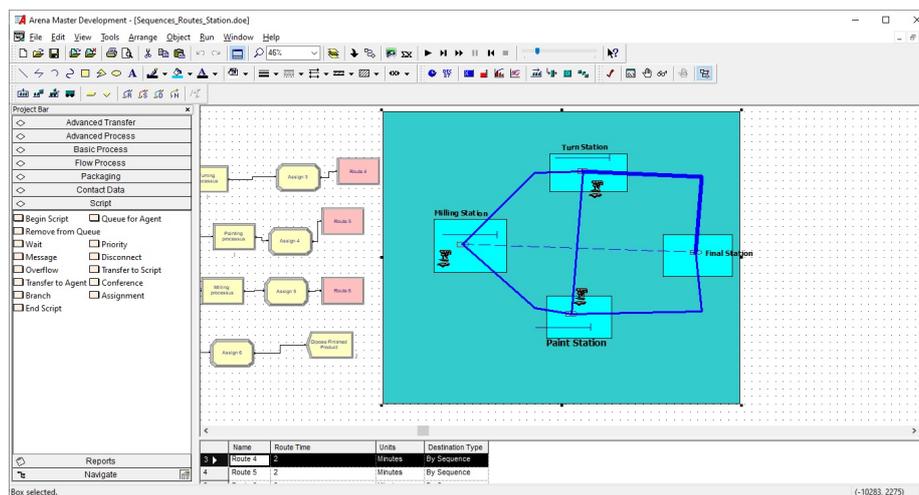


FIGURE 32 – animation des ressources