

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

الجمهورية الجزائرية الديمقراطية الشعبية

MINISTRY OF HIGHER EDUCATION
AND SCIENTIFIC RESEARCH

HIGHER SCHOOL IN APPLIED SCIENCES
--T L E M C E N--



المدرسة العليا في العلوم التطبيقية
École Supérieure en
Sciences Appliquées

وزارة التعليم العالي والبحث العلمي

المدرسة العليا في العلوم التطبيقية
-تلمسان-

Département de la Formation du Second Cycle

Polycopié pédagogique

Filière : Génie industriel

Spécialité : Management industriel et logistique

Présenté par : OUHOUD Amina

Grade : MCB

Titre

Simulation des systèmes de production

(TPs)

Année universitaire : 2021/2022

Table des matières

INTRODUCTION	1
TP°1 INITIATION À LA SIMULATION AVEC LE LOGICIEL ARENA	3
TP°2 TRAITEMENT EN SERIE, TRAITEMENT EN PARALLELE	9
TP°3 ETUDE DES PANNES DES MACHINES ET DES CALENDRIERS.....	16
TP°4 POSTE D'ASSEMBLAGE	20
TP°5 GESTION DES FILES D'ATTENTE	22
TP°6 LIRE ET ECRIRE D'UN FICHER / INPUT ANALYSE.....	25
TP°7 LES MOYENS DE MANUTENTION / ANIMATION 2D.....	27
TP°8 COÛT DE LA PRODUCTION /OPTQUEST	30
SOLUTION des TPs	33
Solution TP°1 INITIATION A LA SIMULATION AVEC LE LOGICIEL ARENA.....	34
Solution TP°2 TRAITEMENT EN SERIE, TRAITEMENT EN PARALLELE	45
Solution TP°3 ETUDE DES PANNES DES MACHINES et DES CALENDRIERS	52
Solution TP°4 POSTE D'ASSEMBLAGE.....	55
Solution TP°5 GESTION DES FILES D'ATTENTE	58
Solution TP°6 LIRE / ECRIRE A PARTIR D'UN FICHER	61
Solution TP°7 MOYEN DE MANUTENTION /ANIMATION 2D.....	67
Solution TP°8 COÛT DE LA PRODUCTION /OPTQUEST.....	75
CONCLUSION	81
Références Bibliographiques	82

Liste des figures (TP)

Figure 1. 1 L'écran d'accueil de Arena	4
Figure 1. 2 La barre d'outils standard	6
Figure 1. 3 La barre d'outils Draw et VIEW.....	6
Figure 1. 4 Les barres d'outils Animate et Animate Transfer	6
Figure 1. 5 Les barres d'outils Run Interaction	7
Figure 1. 6 Les barres d'outils Integration	7
Figure 1. 7 Schéma de la manipulation	7
Figure 2. 1 Le process de fabrication	9
Figure 3. 1 Process avec panne et pause de l'opérateur	16

Figure 6. 1 Fiche de commande.....	25
Figure 6. 2 Temps de traitement et temps de panne	26
Figure 8. 1 les paramètres du projet	30
Figure 8. 2 données des entités	31
Figure 8. 3 donnée de la ressource.....	31

Liste des figures (SOLUTIONS)

Figure1. 1 Modélisation.....	34
Figure1. 2 Boîte de dialogue pour le module Create.....	35
Figure1. 3 Boîte de dialogue pour le module Process	36
Figure1. 4 boîte de dialogue pour ressource	37
Figure1. 5 paramètres de simulation.....	38
Figure1. 6 Résultat de la réplication.....	39
Figure1. 7 Rapport de la ressourcee	39
Figure1. 8 Rapport de la file d'attente.....	40
Figure1. 9 Rapport de l'entité.....	40
Figure1. 10 donnée de l'entité	42
Figure1. 11 animation graphique 2D.....	42
Figure2. 1 Modélisation scénario 4.....	47
Figure2. 2 modèle ressource parallèle avec option Random	48
Figure2. 3 Bloc set.....	48
Figure2. 4 Attribute dans ressources	50
Figure3. 1 modélisation TP3 partie 1	52
Figure3. 2 les données de bloc schedule.....	52
Figure3. 3 Module Process.....	54
figure5. 1 Modèle partie 1	58
figure5. 2 Modèle partie 2	58
figure5. 3 Modèle partie 3	59
figure7. 1 modèle Arena TP 7	67
figure7. 2 Le temps d'attente des engrenages.....	72
figure7. 3 Le taux d'utilisation des ressources.....	72
figure7. 4 Le temps de cycle de chaque pièce	73
figure7. 5 Rapport Arena.....	73

INTRODUCTION

La simulation est un outil puissant pour la modélisation et l'analyse de systèmes complexes. La grande majorité des systèmes réels sont difficiles voire impossibles à étudier via des modèles analytiques en raison du manque des solutions analytiques. En revanche, un modèle de simulation peut presque toujours être construit pour fournir des informations utiles sur le fonctionnement du système et ses mesures de performance. La simulation aide l'analyste à comprendre comment un système fonctionne sous un régime donné. Il peut également être utilisé fréquemment dans les problèmes d'optimisation pour trouver des meilleures solutions ou des solutions satisfaisantes meilleures ou suffisamment bonnes. En effet, la portée de la simulation est désormais extraordinairement large, même dans les systèmes basés sur la technologie.

La modélisation par simulation est largement utilisée dans l'industrie comme outil d'aide à la décision dans de nombreux problèmes industriels, notamment l'estimation des capacités des installations, les tests de méthodes de fonctionnement alternatives, les décisions de mix de produits et les architectures système alternatives. Presque tous les grands projets d'ingénierie au cours des 30 dernières années ont bénéficié d'un certain type de modélisation et d'analyse par simulation. Parmi les exemples notables, citons le projet de gazoduc Trans-Alaska, le projet de tunnel sous la Manche britannique (Chunnel) et la planification des opérations pour le canal de Suez. La simulation continuera sans aucun doute à jouer un rôle majeur dans les études d'analyse des performances de systèmes complexes ;

Jusqu'à les années 1980, la simulation était assez coûteuse et prenait du temps en termes de temps de l'analyse et de temps d'exécution de la machine. L'avènement d'ordinateurs personnels dotés de processeurs et de graphiques puissants, a inauguré de nouvelles capacités qui ont fait de la simulation une approche particulièrement attrayante et rentable pour l'analyse des performances d'une grande variété de systèmes. Les utilisateurs de la simulation actuels peuvent créer et tester des modèles de simulation de manière interactive et tirer parti de fonctionnalités de visualisation et d'animation étendues. La programmation des modèles de simulation a été simplifiée dans un paradigme qui combine la programmation visuelle (graphiques) et la programmation textuelle (instructions).

La modélisation par simulation est un paradigme courant pour l'analyse de systèmes complexes. En effet, ce paradigme crée une représentation simplifiée d'un système à l'étude. Ensuite, Il procède à l'expérimentation, guidé par un ensemble d'objectifs prescrits, tels qu'une conception améliorée

du système, une analyse coût/bénéfice, une sensibilité aux paramètres de conception, amélioration des performances d'un système etc.

Ce document s'intéresse à la modélisation par simulation de systèmes industriels. Ceux-ci comprennent les systèmes de fabrication (par exemple, les lignes de production, les systèmes d'inventaire, les ateliers de travail, etc.) et les systèmes de transport (par exemple, les ports maritimes, les aéroports, etc.). Ce document aborde des sujets pratiques liés à la modélisation par simulation. Tout au long du polycopié, Arena / SIMAN [Kelton et al. (2000)] outil de simulation sera étudié et utilisé dans des exemples pratiques de la simulation.

Arena est un environnement de simulation visuelle à usage général qui a évolué au fil de nombreuses années et de nombreuses versions. Il est apparu pour la première fois en tant que langage de simulation SIMAN orienté blocs, et a ensuite été amélioré par l'ajout de nombreux modules fonctionnels, une visualisation complète de la structure et des paramètres du modèle, des outils d'analyse d'entrée et de sortie améliorés, des fonctions de contrôle et d'animation d'exécution et des rapports de sortie. Arena a été largement utilisé à la fois dans l'industrie et dans le milieu universitaire, raison pour laquelle ce polycopié sur la simulation des systèmes de production utilise Arena comme environnement ou outil de simulation.

Ce polycopié est conçu comme un manuel des travaux pratique de simulation des systèmes de production destiné aux étudiants de deuxième cycle spécialité génie industriel à un niveau d'introduction. Il vise les aspects pratiques de la simulation, ainsi que le fonctionnement de l'environnement de simulation Arena. Cependant, le polycopié n'est pas structuré comme un manuel d'utilisation pour Arena, et nous recommandons fortement aux lecteurs de consulter les fonctions d'aide d'Arena pour plus de détails sur les constructions des modèles avec Arena. En conséquence, ce polycopié a pour objectif :

1. Apprendre à modéliser un processus pour le simuler
2. Conduire un projet de simulation avec toutes ses étapes ;
3. Analyser les résultats trouvés par simulation ;
4. Comparer entre différentes alternatives possibles
5. Appliquer ces techniques à des cas pratiques.

TP°1 | INITIATION À LA SIMULATION AVEC LE LOGICIEL ARENA

Introduction

Les composants de modélisation fondamentaux d'Arena, appelés modules, sont sélectionnés à partir des bibliothèques, tels que *Basic Process*, *Advanced Process* and *Advanced Transfer*. Un module est une construction de haut niveau, composée de SIMAN blocs et / ou d'éléments. Par exemple, un module *Process* modélise le traitement d'une entité et se compose en interne de blocs.

Arena met en œuvre un paradigme de programmation qui combine la programmation visuelle et textuelle. Une session Arena typique comprend les activités suivantes :

1. Sélectionner les icônes de modules/blocs à partir des bibliothèques et les placer sur un canevas de modèle graphique (par glisser-déposer).
2. Connecter les modules graphiquement pour indiquer le flux physique des transactions et/ou le flux logique du contrôle.
3. Paramétrage des modules ou des éléments à l'aide d'un éditeur de texte.
4. Ecriture de fragments de code dans les modules à l'aide d'un éditeur de texte.

Arena dispose d'une interface utilisateur graphique (GUI) construite autour du langage SIMAN. En fait, il est possible de construire des modèles de simulation en utilisant les constructions SIMAN à partir des bibliothèques Blocks et Elements, puisque les modules Arena ne sont que des sous-programmes écrits en SIMAN.

Néanmoins, Arena est bien plus pratique que SIMAN, car il offre de nombreuses fonctionnalités pratiques, telles que des modules de haut niveau pour la construction de modèles, la définition et la collecte de statistiques, l'animation de simulations et la génération de rapports de sortie. La construction de modèles tend à être particulièrement intuitive, puisque de nombreux modules représentent des sous-systèmes réels dans le modèle conceptuel ou le système réel étudié. Les modèles complexes nécessitent généralement à la fois des modules Arena et des blocs SIMAN.

Globalement, Arena fournit un environnement de simulation modulaire permettant de modéliser pratiquement tout scénario impliquant un flux de transactions (discret ou continu) parmi un ensemble de processus. De plus, comme le modélisateur construit un modèle de manière interactive en mode graphique et textuel, Arena est occupé en arrière-plan à transcrire l'ensemble dans SIMAN. Comme Arena génère un code SIMAN correct et vérifie que le modèle ne contient pas d'erreurs syntaxiques (graphiques et textuelles), une grande partie du débogage initial a lieu automatiquement.

1.1 L'écran d'accueil d'Arena

Un écran d'accueil Arena est présenté à la *figure 1.1*. Le lecteur est invité à parcourir l'écran d'accueil Arena et à examiner les objets qui seront mentionnés dans la suite.

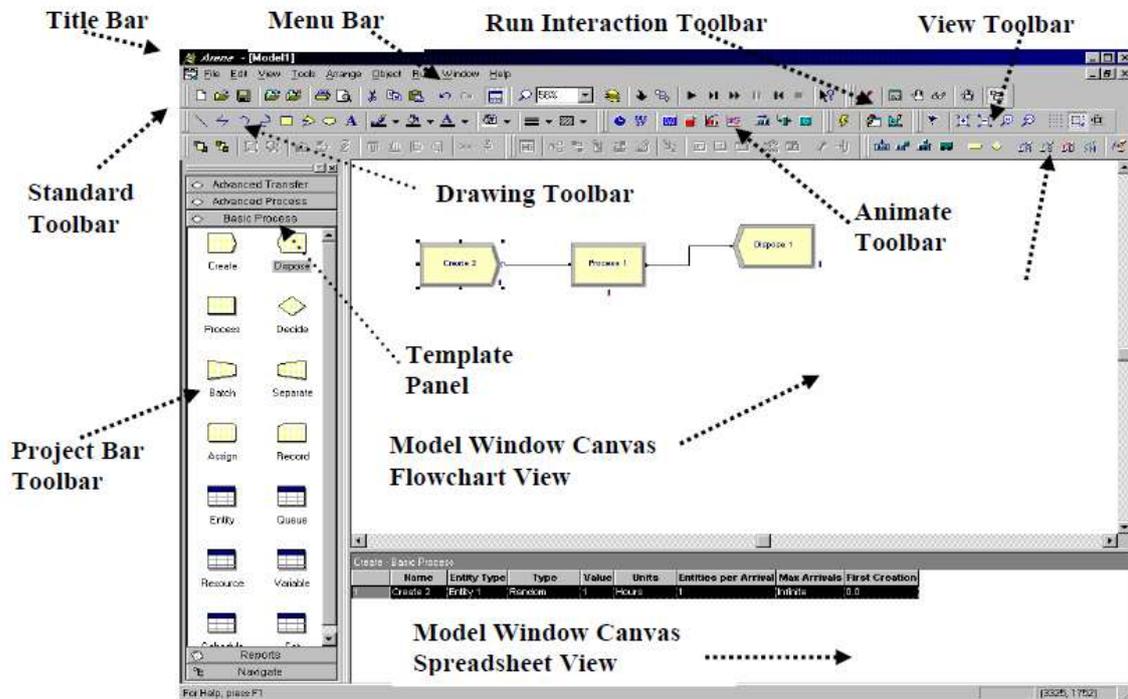


Figure 1.1 L'écran d'accueil de Arena

L'écran d'accueil d'Arena comporte une *barre de titre (Title Bar)* avec le nom du modèle en haut. En dessous se trouve la barre de menu, qui comprend un ensemble de menus généraux et de menus spécifiques à Arena. Sous la barre de menu se trouve un ensemble de barres d'outils (*tool bar*), qui peuvent être affichées et masquées en cliquant avec le bouton droit de la souris sur leur zone grise de fond. Ces barres d'outils sont constituées de boutons qui permettent de construire des modèles ; il est à noter que certains boutons de la barre d'outils reproduisent de manière pratique la fonctionnalité de certaines options de la barre de menu.

La barre de menus et l'ensemble des barres d'outils Arena sont décrits ci-après.

1.1.1 La barre de menu de Arena (Menu Bar)

La barre de menu Arena se compose d'un certain nombre de menus généraux *File, Edit, View, Window et Help*, qui prennent en charge des fonctionnalités assez génériques. Il comprend également l'ensemble de menus spécifiques à ARENA :

- Le menu *Tools* donne accès aux outils liés à la simulation et aux paramètres d'ARENA.
- Le menu *Arrange* prend en charge les opérations de dessin d'organigramme.
- Le menu *Object* prend en charge les connexions de module et la création de sous-modèles.

1.1.2 La barre de projet (*PROJECT BAR*)

La barre d'outils *PROJECT BAR* permet à l'utilisateur d'accéder aux différentes bibliothèques, où se trouve les modules Arena, les blocs SIMAN et divers autres objets. Les bibliothèques peuvent être attachées à la barre d'outils de la barre de projet en cliquant sur le bouton *Attache (ATTACH)* de la barre d'outils *Standard*.

Plus précisément, lorsque le bouton *ATTACH* est cliqué, une boîte de dialogue apparaît à l'écran et affiche les fichiers *.tpo* correspondant à chaque bibliothèque de modèle. Le choix d'un fichier *.tpo* attachera son dans la *barre de projet (project bar)*. Les bibliothèques disponibles dans ARENA sont les suivants :

- La bibliothèque *BASIC Process* se compose d'un ensemble de modules de base, tels que *Create, Dispose, Process, Decide, Batch, Separate, Assign and Record*.
- La bibliothèque *Advanced Process* fournit des modules de base supplémentaires ainsi que des modules plus avancés, tels que *Pickup, Dropoff et Match*.
- La bibliothèque *Advanced transfer* se compose de modules qui prennent en charge les transferts d'entités dans le modèle. Il peut s'agir de transferts ordinaires ou de transferts utilisant des équipements de manutention.
- La bibliothèque *Blocs* contient l'ensemble complet des blocs SIMAN.
- La bibliothèque *Elements* contient les éléments nécessaires pour déclarer les ressources du modèle, les files d'attente, les variables, les attributs et certaines collections de statistiques.

1.1.3 La barre d'outils standard (*The Standard Toolbar*)

La barre d'outils *standard* (*The Standard Toolbar*) d'Arena contient des boutons qui prennent en charge la création de modèles . Un bouton important de cette barre d'outils est le bouton

Connect . Ce bouton permet de connecter des modules Arena ainsi que des blocs SIMAN, et le diagramme résultant décrit le flux de contrôle logique. La barre d'outils *Standard* fournit également des boutons de style VCR pour exécuter un modèle Arena en mode interruption afin de

suivre son évolution .

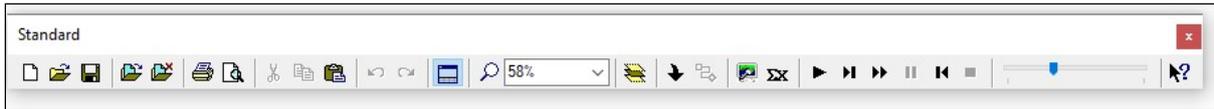


Figure 1. 2 La barre d'outils standard

1.1.4 Les *barres d'outils de dessin et d'affichage (The Draw and View Toolbars)*

La barre d'outils *Draw* prend en charge le dessin statique et la coloration des modèles Arena. Dans le même ordre d'idées, la barre d'outils *View* aide l'utilisateur à visualiser un modèle. Ses fonctions incluent le zoom avant, le zoom arrière. Ces fonctions permettent de visualiser facilement de grands modèles à différents niveaux de détail.

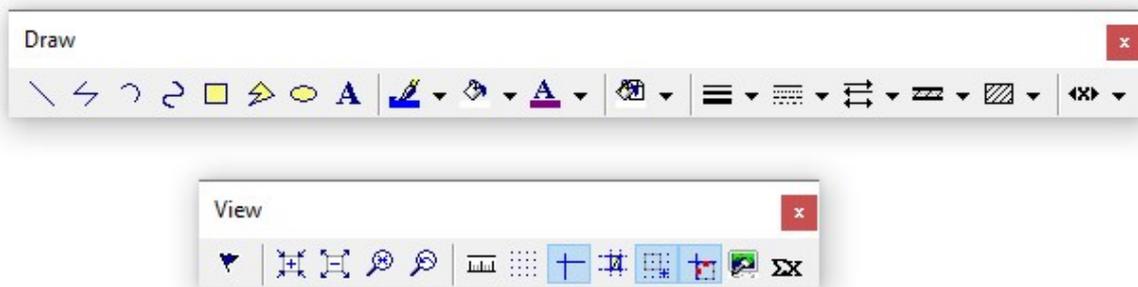


Figure 1. 3 La barre d'outils Draw et VIEW

1.1.5 Les *barres d'outils Animate et Animate Transfer*

La barre d'outils *Animate* est utilisée pour l'animation (visualisation dynamique) des objets du modèle Arena pendant les exécutions de simulation. Les objets animés incluent l'horloge et la date de simulation , les files d'attente (*queue*) , les fenêtres de surveillance des variables(*variable*) , les graphiques dynamiques et les fonctions d'histogramme . La barre d'outils *Animate Transfer* est utilisée pour animer les activités de transfert d'entités, y compris la maintenance des matériaux .



Figure 1. 4 Les barres d'outils Animate et Animate Transfer

1.1.6 La barre d'outils Exécuter (The Run Interaction Toolbar)

La barre d'outils *Run interaction* prend en charge les fonctions de contrôle d'exécution pour surveiller les exécutions de simulation, telles que l'accès au code SIMAN et aux fonctions de débogage de modèle. Il prend également en charge la visualisation de modèle, comme le bouton

Animer les connecteurs  qui active et désactive l'animation du trafic d'entité sur les connexions de module.

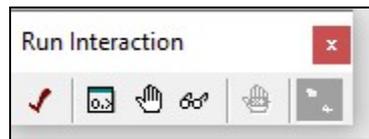


Figure 1. 5 Les barres d'outils Run Interaction

1.1.7 La barre d'outils d'intégration (The Integration Toolbar)

La barre d'outils d'intégration prend en charge le transfert de données (importation et exportation) vers d'autres applications. Il permet également la programmation et la conception Visual Basic.



Figure 1. 6 Les barres d'outils Integration

- Le menu *Run* permet de contrôler l'exécution de la simulation. Son option Configuration... ouvre un formulaire qui permet à l'utilisateur de saisir des informations, telles que les paramètres du projet (nom, analyste, date, etc.), ainsi que les paramètres de réplication (*length*, *warm-up period*, etc.)

Manipulation

On désire étudier au moyen de la simulation le comportement d'un système de production. Il est constitué d'une machine et d'un stock tampon géré en FIFO (first in first out). L'arrivée des produits suit une loi exponentielle avec une moyenne de 30 min. Le temps opératoire sur la machine suit une loi exponentielle avec une moyenne de 24 minutes. Ce système est connu dans la théorie des files d'attente sous le nom de file d'attente **M / M / 1**.

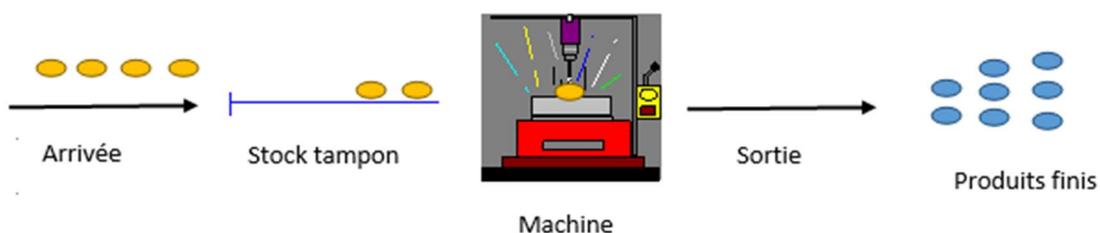


Figure 1. 7 Schéma de la manipulation

1. Faire une simulation avec le simulateur Arena de ce système sur une période de 10 ,000 minutes, ensuite prélever du rapport les paramètres suivants :
 - Les temps de production (de traitement) moyen des produits ;
 - Le taux d'utilisation de la machine ;
 - La taille moyenne du stock tampon ;
 - La durée moyenne d'attente des produits ;
2. Réaliser une animation 2D de ce système

TP°2 | TRAITEMENT EN SERIE, TRAITEMENT EN PARALLELE

Introduction

Cette manipulation est consacrée au traitement en **série** et au traitement en **parallèle** des pièces. On dit que le traitement est en série si les pièces doivent passer par plusieurs machines l'une à la suite de l'autre. On dit que le traitement est en parallèle si les pièces doivent passer par une seule ressource parmi un ensemble de machines

Manipulation

Nous prendrons pour exemple la fabrication de chaussures dans un atelier. Les demandes arrivent suivant une loi exponentielle, avec une moyenne de **1,25 min** ; la première demande arrive à partir de **t=0**.

Le traitement de chaque demande nécessite quatre étapes et se réalisent dans l'ordre présenté ci-après :

- la première étape est la découpe, elle est réalisée par l'opérateur Mohammed avec un temps de traitement = **expo(1) min**
- la seconde étape est l'assemblage, elle est réalisée par l'opérateur ALI avec un temps de traitement = **expo(1) min**
- la troisième étape consiste à faire le vernissage, elle est réalisée par l'opérateur Youcef avec un temps de traitement = **expo (2) min**
- la quatrième étape est mettre les chaussures sous emballage, Elle est réalisée par l'opérateur Riad avec un temps de traitement = **expo(1) min**

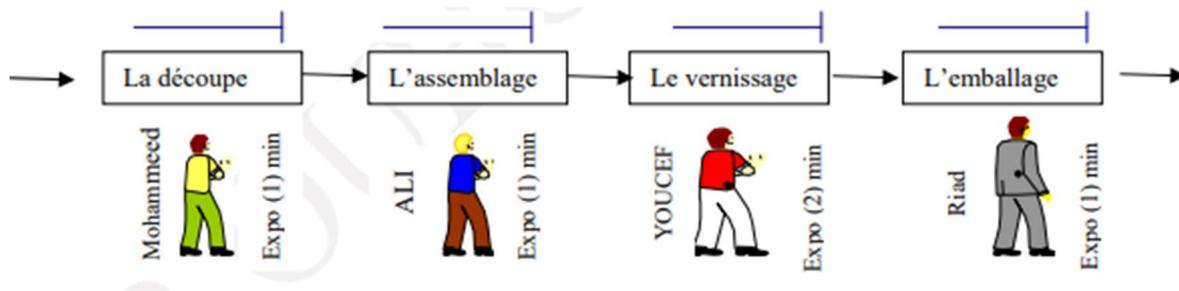


Figure 2. 1 Le Process de fabrication

Nous nous intéressons aux statistiques suivantes :

- le temps de traitement moyen, min et max pour chaque demande
- le temps de cycle moyen, min, max pour chaque demande
- le temps moyen, min, max d'attente des commandes dans chacune des files d'attente
- le nombre moyen, min, max des commandes en attente dans chacune des files d'attente
- le taux d'utilisation des ressources
- Taux de production

1. Scénario-01 :

Nous simulons le système sur **160 heures** (environ un mois de travail) ; **8 h** de travail par jour, 5 jours par semaine.

Pour avoir des statistiques concernant « le temps de cycle » et « le taux de production » on a besoin d'ajouter le module 'record' de la bibliothèque « *basic process* »



Ce module **RECORD** enregistre des informations chaque fois qu'une entité le traverse. Les options sont les suivantes :

- **Count** : augmentera ou diminuera la valeur du compteur nommé par la valeur spécifiée dans le champ 'Value'
- **Entity Statistics** : génère des statistiques générales sur l'entité, telles que des informations sur le temps, le coût et la durée.
- **Time interval** : calculera et enregistrera la différence entre la valeur d'un attribut spécifié et le temps de simulation actuel.
- **Time between** : enregistrera le temps entre les entités entrant dans le module record.
- **Expression** : enregistrera la valeur de l'expression spécifiée.

Donnez les statistiques suivantes :

Statistique	moyenne	Min	Max	Unité
Temps de traitement				
Temps de cycle				
Le temps d'attente				
WIP				
Le taux d'utilisations des ressources :				

<ul style="list-style-type: none"> • Mohammed • Ali • Youcef • Riad 				
Taux de production				

2. Scénario-02 : On introduit le principe de la période transitoire

Portez un changement en ajoutant uniquement une durée de **01heure** à la période transitoire : *warm-up period* < *replication parameters* < *setup* < *run*

Donnez les statistiques suivantes :

Statistique	moyenne	Min	Max	Unité
Temps de traitement				
Temps de cycle				
Le temps d'attente				
WIP				
Le taux d'utilisations des ressources : <ul style="list-style-type: none"> • Mohammed • Ali • Youcef • Riad 				
Taux de production				

Quelles sont vos constatations par rapport au scénario précédent ?

3. Scénario-03 : On augmente, par rapport au scénario-02, à 10 le nombre de réplifications

Portez un changement en ajoutant uniquement 10 comme nombre de réplifications : *number of replications* < *replication parameters* < *setup* < *run*

Donnez les statistiques suivantes :

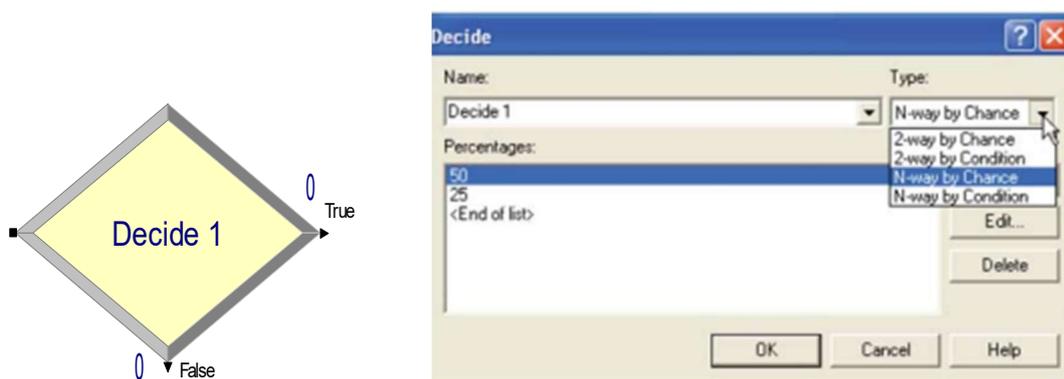
Statistique	moyenne	Min	Max	Unité
Temps de traitement				
Temps de cycle				
Le temps d'attente				

Le nombre d'attente				
Le taux d'utilisations des ressources :				
<ul style="list-style-type: none"> • Mohammed • Ali • Youcef • Riad 				
Taux de production				

Quelles sont vos constatations par rapport aux scénarios précédents ?

4. Scénario-04 : On ajoute, par rapport au scénario-03, un module Decide

On remarque qu'au niveau de la découpe (Mohammed) il y a 10% de mauvaise découpe (déchets). Pour cela on ajoute le module **decide** de la bibliothèque **basic Process**



Comme on peut le voir sur la figure, un module DECIDE comporte quatre types (2-way by chance, 2-way by condition, N-way by chance, N-way by condition).

-L'option **2-way by condition** met en œuvre une construction de base if-then-else, l'entité étant dirigée vers l'un des deux chemins indiqués selon que la condition est vraie ou fausse. Sous forme de pseudo-code, le module DECIDE à 2 voies par condition agit comme suit :

Si la condition est vraie
ALORS envoyer l'entité par le point de sortie **True** ,
else
 Envoyer l'entité par le point de sortie **false**

- L'option **by chance** du module DECIDE permet à l'entité de choisir au hasard dans la liste des points de sortie. Dans l'option **2-way by chance**, l'utilisateur spécifie la probabilité associée au

vrai (**true**) point de sortie. La probabilité restante est associée au point de sortie faux (**false**) point de sortie.

Donnez les statistiques suivantes :

Statistique	moyenne	Min	Max	Unité
Temps de traitement				
Temps de cycle				
Le temps d'attente				
WIP				
Le taux d'utilisations des ressources : <ul style="list-style-type: none"> • Mohammed • Ali • Youcef • Riad 				
Taux de production				

Quelles sont vos constatations par rapport aux scénarios précédents ?

5. Scénario-05: le parallèle

Le responsable a décidé d'ajouter un autre opérateur Mehdi qui fait aussi l'opération de vernissage avec le même temps qui le fait l'opérateur Youcef

- Choisir l'option **Random**:

Statistique	moyenne	Min	Max	Unité
Temps de traitement				
Temps de cycle				
Le temps d'attente				
Le nombre d'attente				
Le taux d'utilisations des ressources : <ul style="list-style-type: none"> • Mohammed • Ali • Youcef • Mehdi • Riad 				

Taux de production				
--------------------	--	--	--	--

Refaire le même scénario avec l'option « **cyclical** »

Statistique	moyenne	Min	Max	Unité
Temps de traitement				
Temps de cycle				
Le temps d'attente				
Le nombre d'attente				
Le taux d'utilisations des ressources : <ul style="list-style-type: none"> • Mohammed • Ali • Youcef • Mehdi • Riad 				
Taux de production				

Quelles sont vos constatations par rapport aux scénarios précédents ?

6. Scénario-0-6: le parallèle avec temps de traitement différent

Supposons que **Mehdi** fait l'opération de vernissage avec une durée = **expo (1) min**

Donnez les statistiques suivantes :

Statistique	moyenne	Min	Max	Unité
Temps de traitement				
Temps de cycle				
Le temps d'attente				
WIP				
Le taux d'utilisations des ressources : <ul style="list-style-type: none"> • Mohammed • Ali • Youcef • Riad 				

Taux de production				
--------------------	--	--	--	--

Quelles sont vos constatations par rapport aux scénarios précédents ?

7. Synthèse des résultats :

Statistique	Scén-01	Scén-02	Scén-03	Scén-04	Scén-05_1	Scén-05_2	Scén-06
Temps de traitement							
Temps de cycle							
Le temps d'attente							
Le taux d'utilisations des ressources : <ul style="list-style-type: none"> • Mohammed • Ali • Youcef • Mehdi • Riad 							
Taux de production							

Quelles est le modèle le plus performant, commentez ?

TP°3 | ETUDE DES PANNES DES MACHINES ET DES CALENDRIERS

Introduction

Ce TP aborde deux divers sujets qui peuvent améliorer vos capacités de modélisation. Le premier examine d'abord la modélisation des processus d'arrivée non-stationnaires.

Objectif

- Pouvoir modéliser les arrivées non-stationnaires avec Arena
- Être capable de modéliser l'arrêt, l'ordonnancement des opérateurs, les pannes et la réparation des ressources

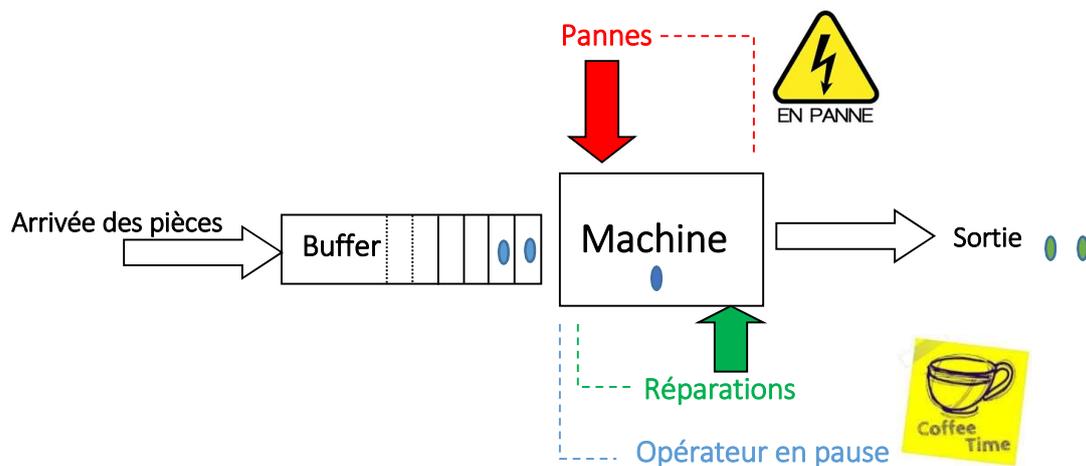


Figure 3. 1 process avec panne et pause de l'opérateur

Manipulation

1/ Calendrier d'arrivée

Considérons un système de production, constitué d'une machine et d'un stock tampon géré en FIFO. Les produits arrivent avec un taux d'une **seule** pièce chaque **10 min** et avec un taux de **deux** pièces chaque **20 min**. Le temps opératoire sur la machine est de **10 min**. La machine traite un produit à la fois.

Simuler le fonctionnement de ce système sur une période de **8 heures**. Commenter les résultats obtenus.

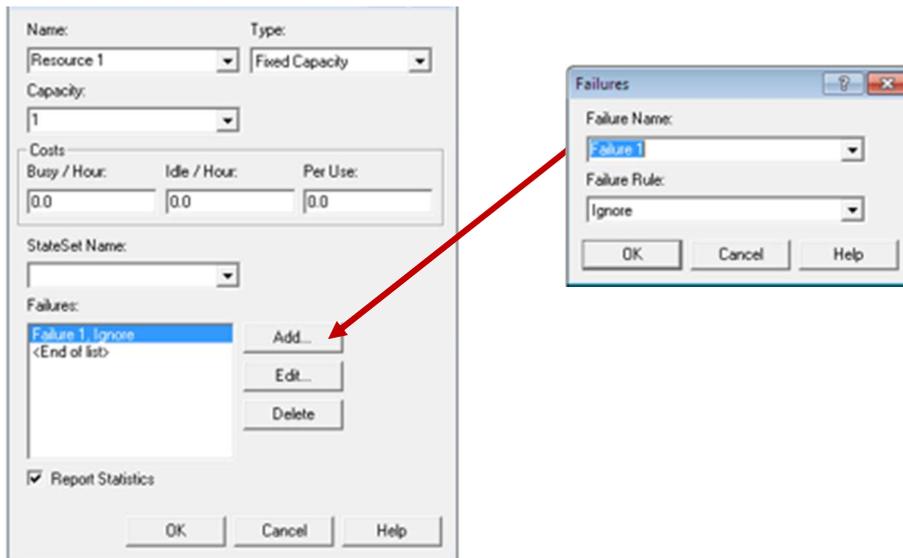
On s'intéresse aux caractéristiques suivantes :

1. Le temps de traitement des produits
2. Le taux d'utilisation de la machine
3. La taille du stock tampon moyenne
4. La durée d'attente moyenne des produits dans la file d'attente

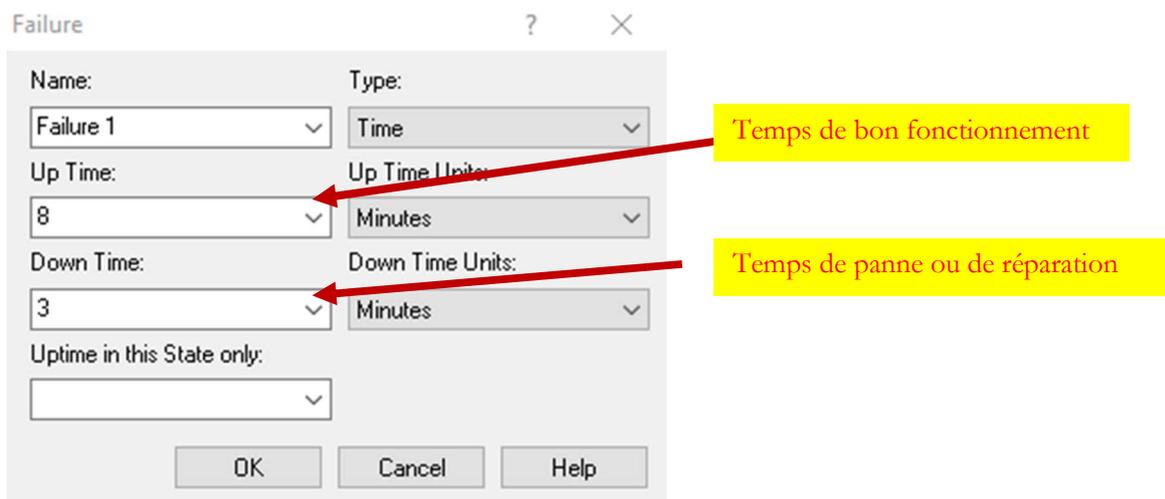
2/ panne d'une machine

Nous allons introduire des pannes pour la machine.

Ajouter à la définition de la ressource « machine » une panne telle que la durée de bon fonctionnement est de **8 minutes** et la durée de réparation est de **3 minutes**.



Pour ajouter les temps de panne et le temps de réparation > Dans la bibliothèque **advanced process** > bloc **failure**



Trois options sont possibles pour « **failure rule** ».

- **Wait :**
- **Ignore :**
- **Preempt :**

1. Choisir dans un premier temps l'option **Ignore**
2. Faire une première simulation, ou on réalise **un seul** produit.
 - Le temps de traitement du produit
 - Le taux d'utilisation de la machine

- La taille du stock tampon moyenne
- La durée d'attente moyenne du produit.
- Le temps de cycle

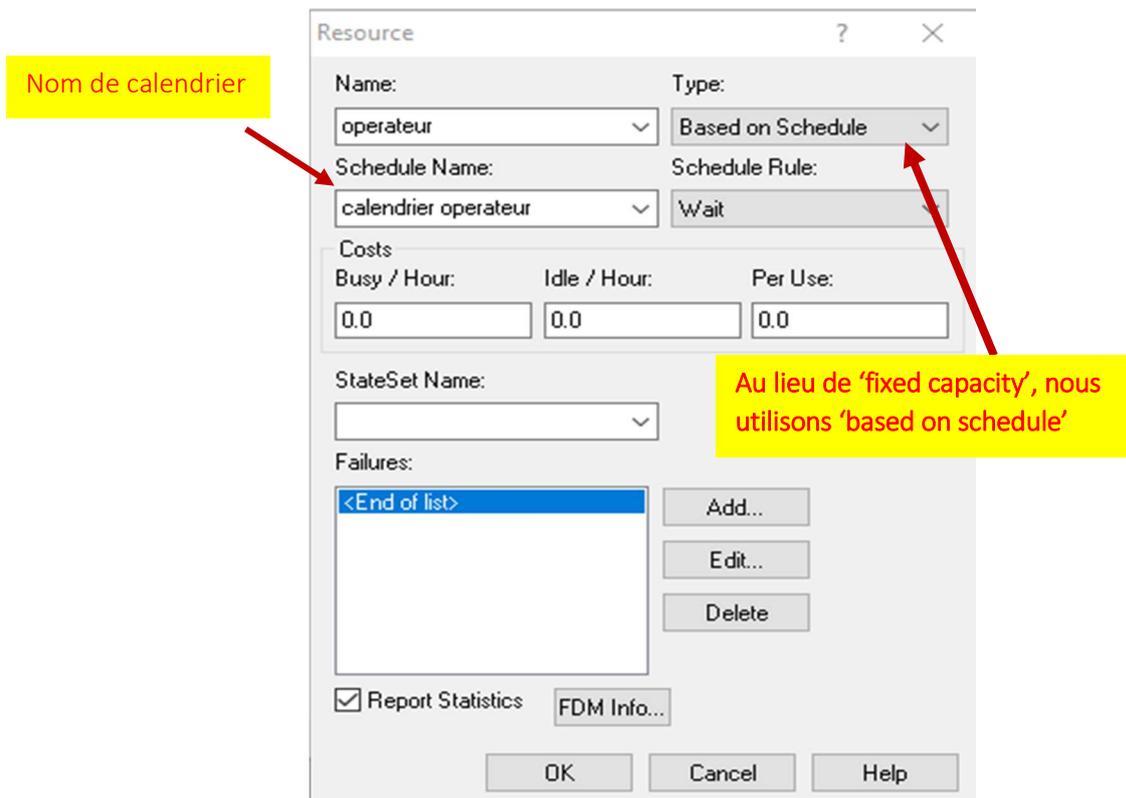
3. Refaire la simulation avec les deux autres options : **Wait et preempt**

3/ calendrier d'un opérateur

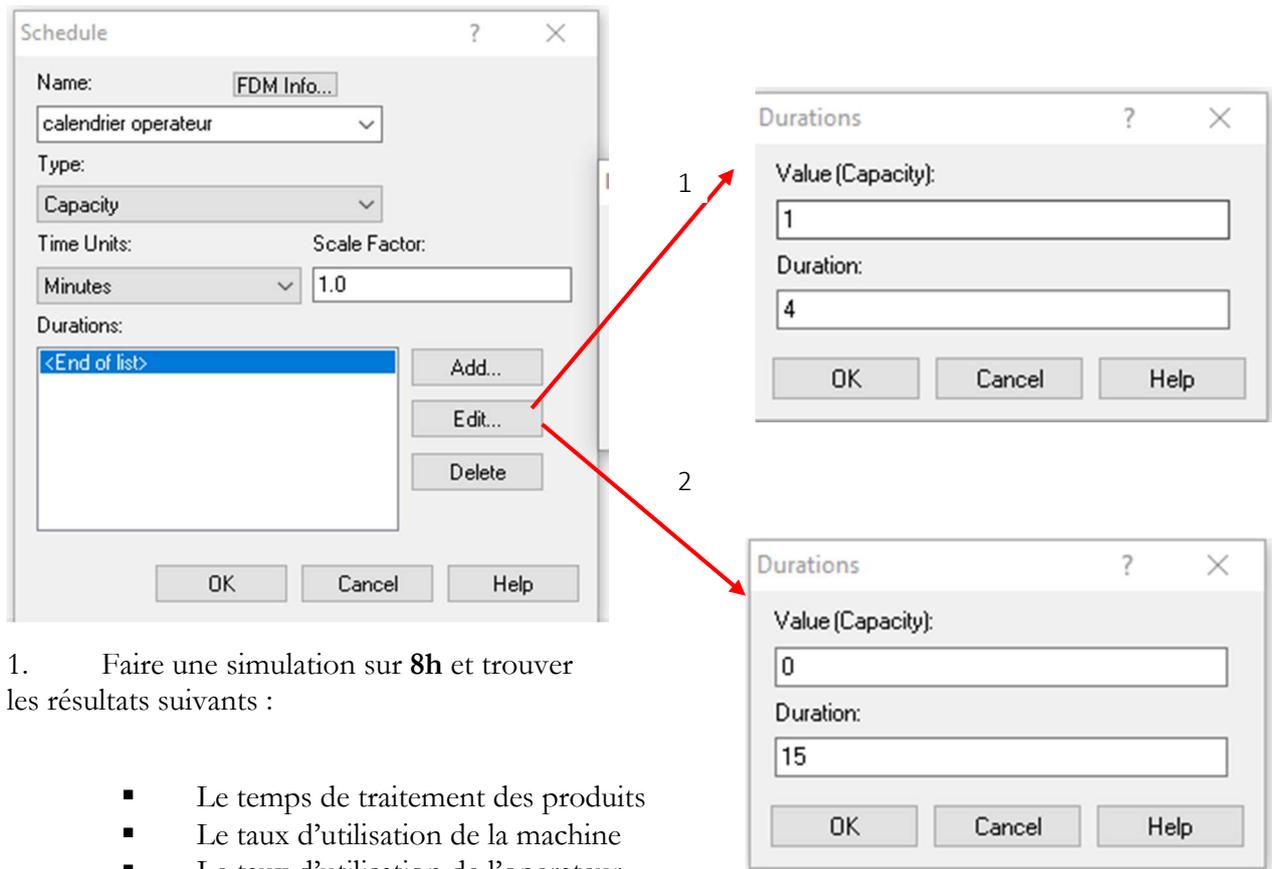
Considérons le même système de production mais cette fois les pièces doivent être traitées par la même machine et un opérateur (en même temps). La machine et l'opérateur sont tous les deux des ressources. Le temps de traitement est de **10 minutes**.

L'opérateur prend une pause de **10 min** chaque **4h** de travail.

Pour modéliser la pause de l'opérateur nous allons lui attribuer un calendrier de travail : **basic process > Ressource**



Dans la bibliothèque **Basic process > schedule**



1. Faire une simulation sur **8h** et trouver les résultats suivants :

- Le temps de traitement des produits
- Le taux d'utilisation de la machine
- Le taux d'utilisation de l'opérateur
- La taille moyenne du stock tampon
- La durée d'attente moyenne du produit

TP°4 | POSTE D'ASSEMBLAGE

Manipulation

Nous considérons un atelier d'assemblage des circuits imprimés. L'objectif étant d'assembler des composantes électroniques. Pour ce faire, l'atelier est constitué de plusieurs machines et chaque machine est précédée d'une file d'attente de capacité infinie.

Les pièces arrivent par lots de 10 unités avec un temps entre deux arrivées est aléatoirement distribué de moyenne 2 h. Parmi les lots arrivés, 40% sont de type Partie 1, et 60% de type partie 2.

Traitement de la partie 1 :

Le traitement de la partie1 comporte les opérations suivante : une désintégration chimique suivi par un processus d'impression de circuit où le temps de désintégration est **uniforme** entre **0,5 et 2 h**, et le temps d'impression de circuits est constant de **1h** .

Traitement de la partie 2 :

Le traitement de la deuxième partie comporte un processus de nettoyage électronique suivi d'un processus d'impression de circuit (la même machine d'impression de la partie 1).les temps de traitement correspondants sont **2h et 3h** respectivement.

Un poste d'assemblage prend alors une unité de chaque partie et les assemble dans une opération qui dure exactement **1h**.

Les unités assemblées procèdent à une opération d'essais où des lots de cinq pièces sont testés simultanément. Le temps de cette opération est **triangulaire** avec les paramètres **6 , 8 et 10 h**. Après l'essai, les lots sont subdivisés en unités individuelles tels que **80 %** sont détectées bonnes et les autres défectueuses

Après chaque période de **8H**, tous le processus sont fermés pour une opération de maintenance de **1h** .

Les arrêts aléatoires suivants sont observés dans chaque processus :

machines	Type de panne	Temps de bon fonctionnement (h)	Temps de réparation(h)
Désintégration	aléatoire	Expo(20)	Unif(1,2)
Nettoyage	aléatoire	Expo(30)	Unif(3,4)
Impression	aléatoire	Expo(40)	Tria(1,3,4)
Test (essai)	déterministe	Chaque 200 unités	1h

1/ Modéliser ce système et le simuler sur un période d'une année et reporter les résultats suivants :

- le temps de traitement moyen
- les taux d'utilisation des machines
- le temps d'attente moyen pour chaque file d'attente
- Le nombre moyen des pièces dans chaque file d'attente
- le temps de cycle
- le temps entre deux sorties
- le nombre moyen des pièces dans le système

TP°5 | GESTION DES FILES D'ATTENTE

Introduction

C'est la loi qui définit la théorie des files d'attente. Elle lie le temps d'attente, les encours et le débit du procédé. Elle est utilisée pour gérer les files d'attente. Elle est aussi utilisée pour calculer le temps de traversé d'un stock dans une étude.

La loi de **Little** s'applique à tous type de processus, quel que soit sa variabilité.

- $WIP = \text{Work In Process}$ (Stock d'encours)
- T : débit par unité de temps
- LT : Lead Time (Temps de Cycle moyen passé dans le système)

$$WIP = T \times LT$$

Manipulation

1/ Partie 1

Un poste de travail permettant de fabriquer deux produits notés A et B.

- Les produits A arrivent toutes les 10 mn. Leur durée opératoire est de 4 mn.
- Les produits B arrivent toutes les 6 mn à partir de $t = 3$ mn. Leur durée opératoire est de 3 mn.
- Lorsqu'un produit arrive, si le poste est disponible, il est traité par ce dernier.
- Si le poste est occupé, les produits attendent dans une file d'attente gérée en FIFO.

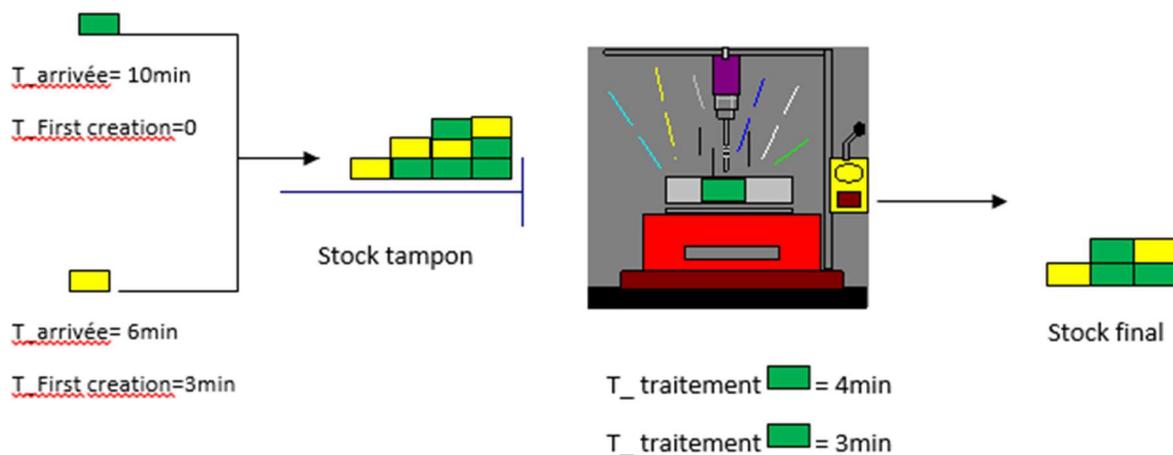


Figure 5.1 process de fabrication

Simuler le système sur une période de 8h et prélever les résultats suivant du rapport :

- Le temps de traitement moyen,
- Le taux d'utilisation de la ressource ;
- Le temps d'attente moyen dans la file d'attente,
- Le nombre moyen des pièces dans la file d'attente,
- La loi de litte est vérifiée pour le système ?justifier votre réponse
- La loi de little est vérifiée par type de produit A et B ? justifier votre réponse

Partie 2 : Modélisation du temps de changement

Pour que le système soit similaire à la réalité, on vous demande de modéliser le temps de changement de production. La production en cours est modélisée par une variable globale qui contient le type de produit en cours de réalisation. Lorsqu'un nouveau produit arrive sur le poste de travail, si son type est différent de la production en cours, on modélisera le nouveau réglage de la machine par une attente qui est évaluée à 20 mn. Celle-ci viendra se cumuler au temps opératoire.

Le simulateur permettra de mesurer, en plus des indicateurs de la question précédente, **le nombre de changement de production** qui seront intervenus.

On réalisera **trois simulations**, en faisant varier la règle de gestion de la file d'attente :

- FIFO
- Lowest Attribute Value (Type)
- Highest Attribute Value (Type)

Comparer entre les trois options par apport aux nombres de changements

Partie 3 : Lot optimal/ Process analyser

Pour améliorer le comportement du système, on réalise des lots de fabrication. La constitution des lots est modélisée de la façon suivante : **N** est une valeur numérique indiquant la taille du lot.

Un block « Batch » permet d'accumuler, au fur et à mesure de leur arrivée, N produits. Dès que ce nombre est atteint, les N entités quittent la file «Batch 1.queue» et sont transformées en une entité. Le block « Separate » permet de reconstituer le lot qui est envoyé en une seule fois dans la file « Stock tampon » (qui est alors gérée en FIFO).

Pour simplifier, on considère que les tailles des lots de produit 1 et de produit 2 sont identiques.

Faire un modèle reprenant celui de la question précédente en intégrant la constitution de lot.

Ce modèle (en plus de la mesure des indicateurs de la question précédente) permettra de tracer quatre courbes :

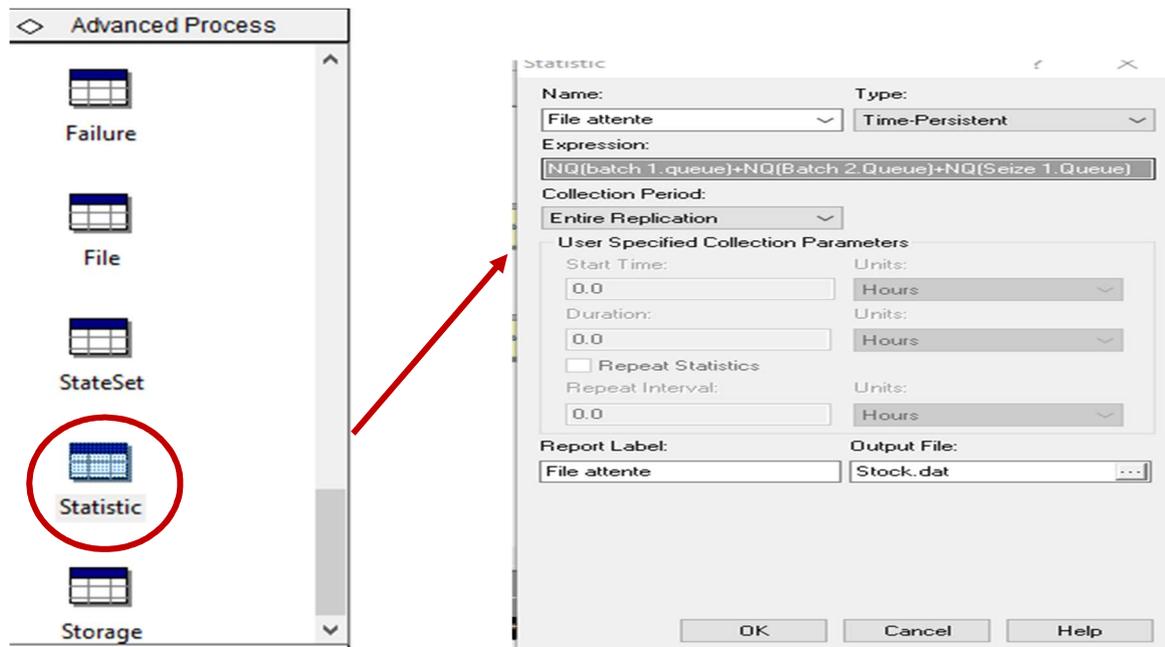
La valeur de la file d'attente du lot 1

La valeur de la file d'attente du lot 2

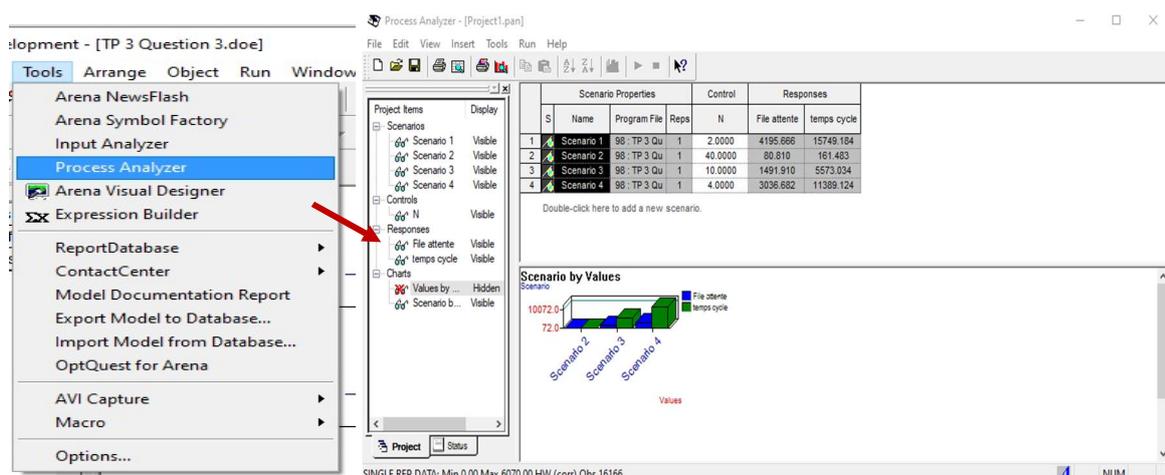
La valeur de la file Stock tampon

Le stock total des produits dans le poste (soit la somme des trois valeurs précédentes).

Pour cela, on utilisera le module « **Statistics** »



En utilisant l'outil **process analyser**, rechercher le nombre N par lot optimal (Celui qui permet de stabiliser le nombre de produits en attente dans le poste de travail tout en minimisant le temps moyen de production des produits.



La simulation sera faite sur une durée de 800 heures. Le simulateur sera placé en mode « Batch run » pour limiter la durée des simulations.

TP°6 | LIRE ET ECRIRE D'UN FICHIER / INPUT ANALYSER

Introduction

Comme n'importe quel langage de programmation, Arena™ a la capacité de lire et d'écrire des informations à partir des fichiers.

Ce TP est destiné à préparer une impression et comparer des modèles de simulation. À ce titre, on déduit les objectifs suivants :

Objectif

- Préparer un fichier texte ou sur Excel afin d'enregistrer le fichier,
- Utiliser l'**Input Analyzer** afin de déterminer la distribution de probabilités de différentes valeurs enregistrées

Manipulation

1/ Lire à partir d'un fichier Text ou Excel

Chaque jour un magasin reçoit une fiche de commande qui contient au max **100 commandes** enregistré dans un fichier text nommé «**fiche commande** ». *Voir figure 6.1*. Cette fiche contient le type de produit commandé, il existe trois types de commande : type **1**, type **2** et type **3**.

Les trois commandes passent par deux ressources nommées **Machine 1** et **machine 2** respectivement tel que le temps de traitement et le temps de panne de chacune des ressources par apport chaque type de commande sont enregistrés dans un fichier Excel nommé « traitement /panne » *voir figure 6.2*



Figure 6. 1 fiche de commande

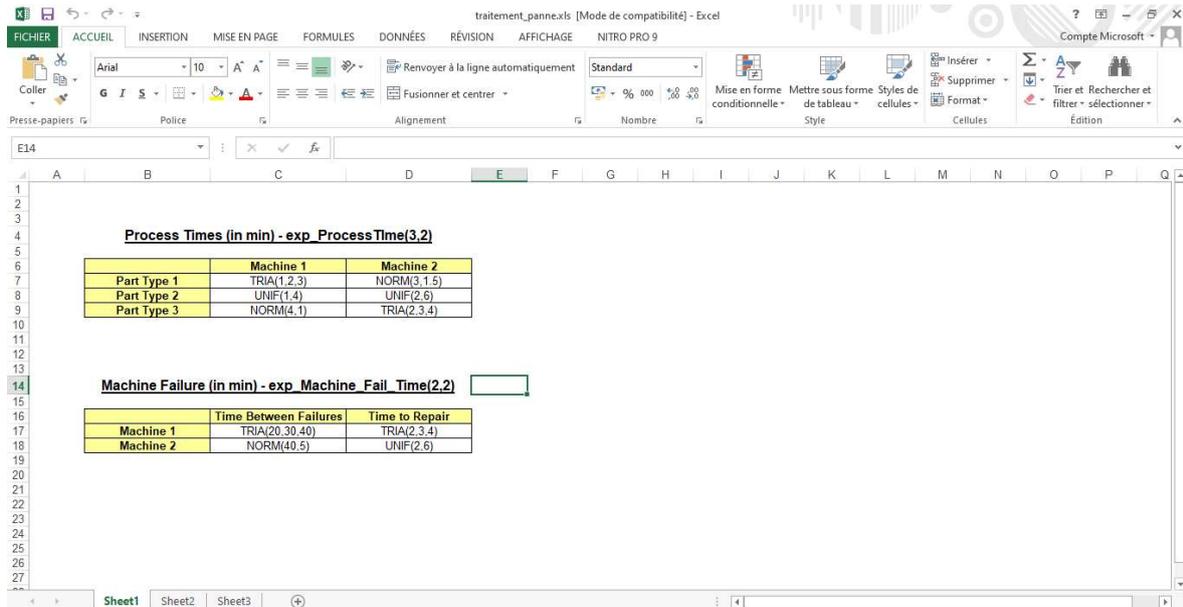


Figure 6. 2 temps de traitement et temps de panne

1. Modéliser le système
2. Prélever les résultats suivants du rapport
 - Le temps de traitement moyen des commandes
 - Le temps d'attente moyen
 - Le taux d'utilisation des machines (machine 1 et 2)
 - Le temps de séjour moyen des commandes dans le système

2/ Ecrire à partir d'un fichier Excel

Reprendre la même manipulation mais cette fois -ci, je vous demande de

- Sauvegarder le temps de cycle et le temps d'attente de chaque commande dans différents fichiers.

1. En utilisant l'outil Input analyser, déterminer la distribution de probabilité de ces temps

TP°7 | LES MOYENS DE MANUTENTION / ANIMATION 2D

Introduction

Le but de cette manipulation est de se familiariser avec les notions de base relatives au transfert des produits dans un modèle SIMAN

Manipulation

Considérons un atelier qui produit trois types d'engrenages (voir fig 7.1), G1, G2 et G3,. L'atelier est réparti géographiquement sur le sol de l'usine selon les emplacements suivants :

- Un quai d'arrivée
- Un poste de fraisage avec quatre fraiseuses
- Un poste de travail de forage avec trois machines de forage
- Un atelier de peinture avec deux cabines de peinture
- Une zone de polissage avec un seul travailleur
- Une sortie de magasin

Les distances entre les emplacements sont indiquées dans le tableau suivant.

	quai d'arrivée	Station fraisage	Station forage	Station peinture	Station polissage	La sortie
quai d'arrivée		100	100	250	250	550
Station fraisage	100		300	400	150	300
Station forage	100	300		150	400	500
Station peinture	250	400	150		300	400
Station polissage	250	150	400	300		200
La sortie	550	300	500	400	200	

Tableau 7.1 Les distances

Les engrenages arrivent par lots de 10 unités avec un temps entre deux arrivées est uniformément distribué entre 400 et 600 minutes. Parmi les lots arrivés, 50% sont de type G1, 30% de type G2 et 20% de type G3. Un engrenage arrive au quai d'arrivée et de là, est envoyé à sa séquence spécifique d'opérations de fabrication (dépend du type). Une séquence comprend un sous-ensemble d'opérations de fraisage, de forage, de peinture et de polissage. Le tableau 7.2 présente le plan d'opérations indiquant la séquence des opérations et les temps de traitement

associés pour chaque type d'engrenage. La disposition de l'atelier et les séquences d'opérations des types d'engrenage sont illustrées à la *figure 7.1*

TYPE d'engrenage	Routage ou la séquence	Le temps de traitement
G1	Fraisage	35
	Forage	20
	Peinture	55
	Polissage	15
G2	Fraisage	25
	Peinture	35
	Polissage	15
G3	Forage	18
	Peinture	35
	Polissage	15

Tableau 7.2 : le séquençage des entités

Les engrenages sont transportés entre les sites par deux camions roulant à une vitesse constante de 100 pieds / minute. Chaque camion ne peut transporter qu'un seul engrenage à la fois.

Pour analyser les performances de l'atelier, prévoyez de lancer une simulation sur une année d'exploitation.

Ces hypothèses ont été faites lors de la modélisation du système.

Hypothèses :

- -La vitesse du transporteur (camion) est la même pour les véhicules chargés et vides.
- Le transporteur affranchi reste à la gare de destination jusqu'à ce qu'une autre station le demande.
- Le Job Shop travaille 24 heures par jour en 3 équipes de 8 heures chacune.

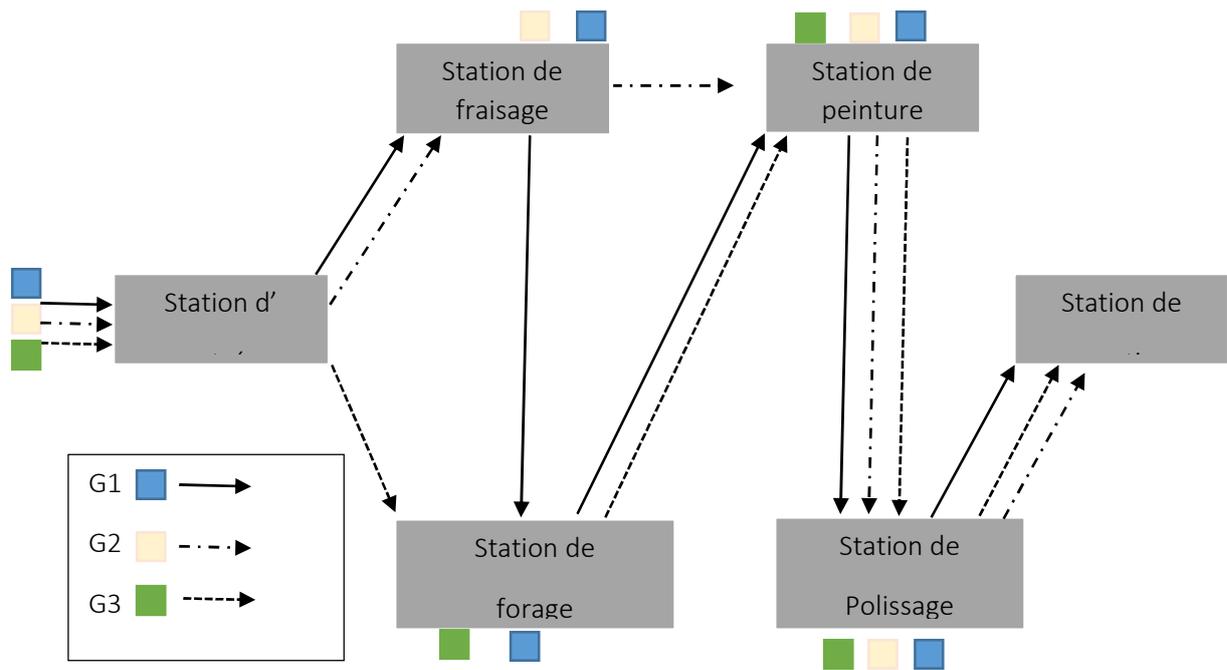


Figure 7.3 Disposition de l'atelier de travail et séquences d'opérations par type d'entité

Les statistiques suivantes sont intéressantes :

- Temps de séjour des engrenages (par type)
- Retards
- Utilisations de la machine
- Proposition d'une amélioration

TP°8 | COUT DE LA PRODUCTION /OPTQUEST

Introduction

Les attributs coûts :

Dans le logiciel ARENA, on peut associer, à chaque entité, un ensemble de coût à un instant donné. Les coûts d'une entité sont connus par l'ensemble d'attributs système suivant :

1. Entity.VACost : Cet attribut désigne le coût de la valeur ajoutée de l'entité. Ce coût caractérise l'argent dépensé utilement pour augmenter la valeur de l'entité.
2. Entity.NVACost : Cet attribut désigne le coût de la « Non-valeur ajoutée » de l'entité. Ce coût caractérise l'argent dépensé inutilement car cette dépense n'augmente pas la valeur de l'entité.
3. Entity.WaitCost : Cet attribut désigne le coût de stockage de l'entité. Ce coût caractérise l'argent dépensé lors des attentes de l'entité dans des files d'attentes qui modéliseront les opérations de stockage.
4. Entity.TranCost : Cet attribut désigne le coût de transfert de l'entité. Ce coût caractérise l'argent dépensé lors des opérations de transfert de l'entité.
5. Entity.OtherCost : Cet attribut permet de désigner les coûts non classés dans les types précédents.

Arena n'affiche pas les statistiques relatives aux coûts par défaut, pour avoir ces statistiques il faut cocher la case **costing** dans l'onglet **Project parameters** dans la fenêtre **Run Setup**.

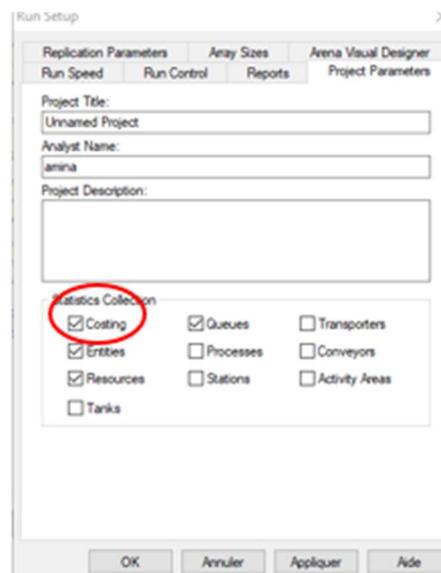


Figure 8. 1 les paramètres du projet

Le bloc déclaratif ENTITY de la bibliothèque basic process

Le module ENTITY permet entre autre de définir les coûts initiaux des entités créées

Mais il permet aussi de définir le « Holding cost » ou coût de maintien : c'est le coût dépensé pour une heure de séjour de la pièce dans le système.

Figure 8. 2 données des entités

Dans le module **Ressource** il est possible d'introduire :

- A. Le coût de l'utilisation de la ressource par heure (busy/hour)
- B. Le coût du non utilisation de la ressource par heure (Idle/hour)
- C. Le coût par utilisation (Per use).

Figure 8. 3 donnée de la ressource

Manipulation

Considérons une ligne de production, composée de deux postes de travail en série. Les deux postes de travail sont précédés d'un espace tampon limité max de 10 pièces. Les pièces arrivent par lot de 10, Le temps entre deux arrivées successives est de 3.125 heures. Le temps de traitement sur le premier poste de travail est de 15 minutes, tandis que le traitement sur le second poste de travail se fait par lot de 5 pièces et prend 30 minutes. Le holding cost est de 1 unité monétaire.

Le coût de l'utilisation de la ressource est de 0.8 unité monétaire par heure.

Le coût du non utilisation de la ressource est de 0.5 unité monétaire par heure

Le coût par utilisation est de 1 unité monétaire par heure.

1. Simuler ce système sur une période de 10 000 minutes,

Calculer théoriquement

- La taille moyenne de chacun des stocks tampons ;
 - Le temps moyen d'attente dans chacun des stocks tampon ;
 - Le taux d'utilisation de chacun des postes de travail ;
 - Le coût de valeur ajouté ;
 - Le coût valeur non ajouté ;
 - Le coût d'attente ;
 - Le coût total
2. En utilisant l'outil Optquest de arena, trouvez la capacité minimale de la machine qui me donne le cout total min

SOLUTION des TPs

Solution TP°1 INITIATION A LA SIMULATION AVEC LE LOGICIEL ARENA

La simulation passe par trois étapes importantes qui sont la *modélisation*, *simulation* et *résultats*

1. Modélisation

La simulation du système de production ci-dessus nécessite les actions suivantes :

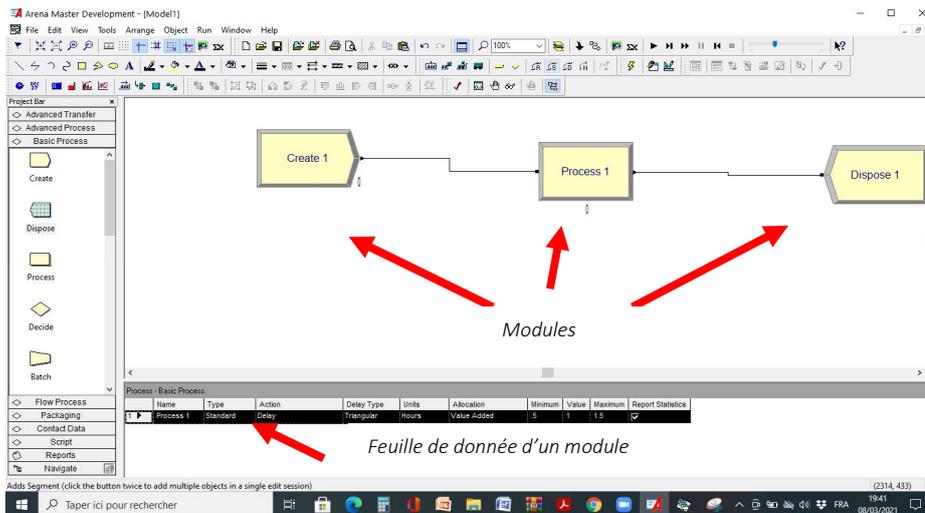


Figure1. 1 modélisation

- Les pièces ou les produits ou les entités sont créés, un à la fois, avec un temps d'arrivée déterministe. Les entités qui arrivent sont envoyés au poste de travail (machine).
- Si la machine est occupée par le traitement d'un autre produit, l'entité qui 'arrivée est mis en file d'attente (stock tampon).
- Lorsqu'un produit avance vers la tête du stock tampon, il saisit la machine pour le traitement une fois qu'il devient disponible, et le maintien pendant une période de temps appelée temps de traitement (temps opératoire).
- À la fin du traitement, le produit quitte la machine et il est supprimé du système (mais pas avant que sa contribution aux statistiques des mesures de performance requises ne soit calculée.)

1.1 la création des entités

Les transactions (entités Arena) sont créées à l'aide du module **create** de la bibliothèque **Basic Process**. La spécification d'arrivée du module **Create** est affichée dans la boîte de dialogue de la Figure 1.2.

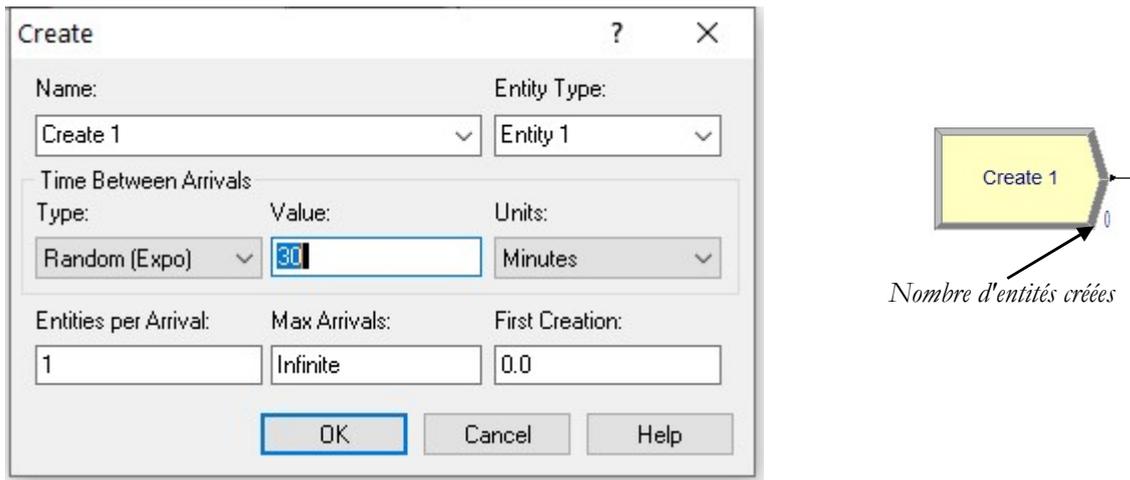


Figure 1. 2 Boîte de dialogue pour le module Create

La boîte de dialogue contient des informations sur le temps entre deux arrivées (champ *time between Arrivals*), la taille du lot (champ *Entités per arrival*), le nombre maximal d'arrivées des e (champ *Max Arrivals*), l'heure de la première création d'entité (champ *First Creation*), etc... Un menu déroulant *Type* pour le champ *time between Arrivals* propose les options suivantes :

- Random (temps entre deux arrivées est exponentielles avec moyenne donnée dans le champ *Value*).
- Schedule (permet à l'utilisateur de créer des temps d'arrivée à l'aide du module Schedule de la bibliothèque *Basic Process*).
- Constant (le temps entre deux arrivées des produit est fixes (déterministe)).
- Expression (tout type de temps entre deux arrivées spécifié par une expression Arena, y compris les distributions Arena).

1.2 Le traitement des entités

Le traitement des pièces (y compris les priorités) est spécifié dans le module Process, dont la boîte de dialogue est illustrée à la *Figure 1.3*

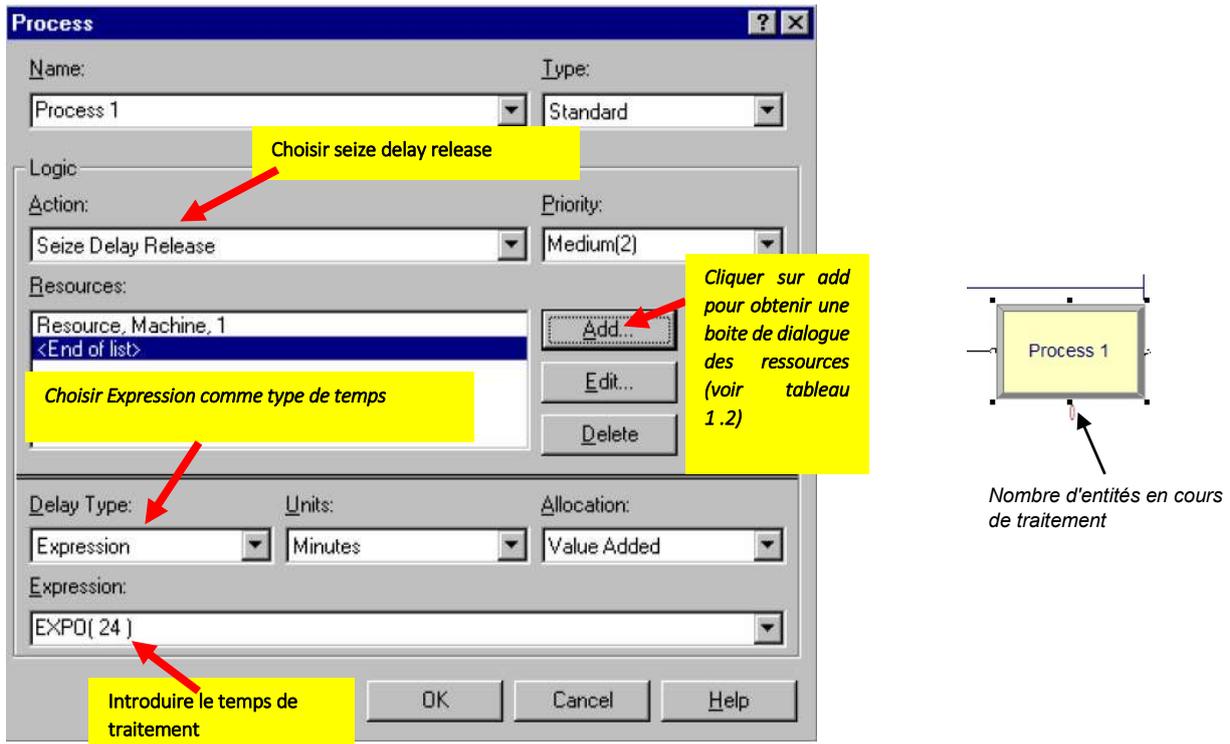


Figure1. 3 Boîte de dialogue pour le module Process

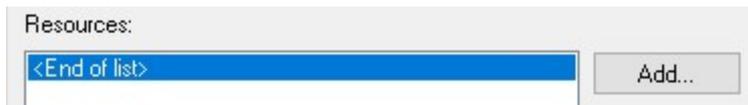
Chaque champ du module *process* correspond à une instruction bien déterminé le tableau suivant résume toutes ces instructions :

Champs	Description
Name	Le nom de module
Type	Le traitement standard signifie que toute la logique sera stockée dans le module Process et définie par une action particulière. Le submodel indique que la logique sera définie de manière hiérarchique dans un sous-modèle pouvant inclure n'importe quel nombre de modules.
Action	Type de traitement qui se produira dans le module. Le type delay indique qu'un délai de traitement sera encouru sans aucune contrainte de ressources. Le type seize Delay indique qu'une ou plusieurs ressources seront allouées dans ce module et qu'un délai se produira, mais que la libération de ressources se produira ultérieurement. Le type seize delay release indique qu'une ou plusieurs ressources seront allouées, suivies d'un délai de traitement, puis la ou les ressources allouées seront libérées. Le type delay release indique qu'une ou plusieurs ressources ont été précédemment allouées et que l'entité va simplement rester un délai et libérer la ou les ressources spécifiées.
Priority	Valeur de priorité de l'entité attendant au niveau de ce module la ou les ressources spécifiées si une ou plusieurs entités attendent la ou les mêmes ressources n'importe où dans le modèle. Non visible lorsque Action est Delay ou Delay Release ou lorsque Type est submodel.
Resources	Répertorie les ressources ou les ensembles de ressources utilisés pour le traitement des entités. Ne s'applique pas lorsqu'Action est delay ou lorsque Type est Submodel.
Delay Type	Type de temps de traitement. Constant et Expression nécessitent des valeurs uniques, tandis que Normal, Uniforme et Triangulaire nécessitent plusieurs paramètres.

Units	Unité de temps de traitement
allocation	Détermine comment le temps de traitement et les coûts de processus seront alloués à l'entité. Le processus peut être considéré comme étant à valeur ajoutée, sans valeur ajoutée, transfert, attente ou autre et le coût associé sera ajouté à la catégorie appropriée pour l'entité et le processus.
Value	Le temps de traitement

Tableau 1.1 boîte de dialogue *process*

Pour ajouter les ressources on clique sur add



Une boîte de dialogue s'ouvre

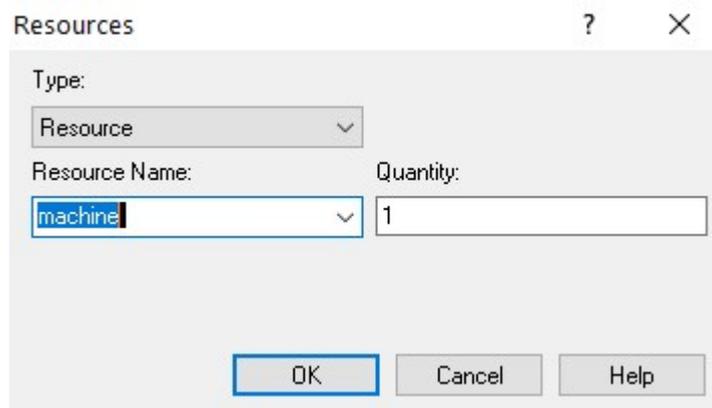


Figure1. 4 boîte de dialogue pour ressource

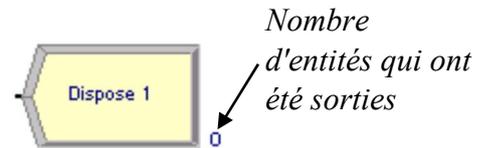
Type	Spécification d'une ressource particulière ou sélection d'un ensemble de ressources (set).
Resource Name	Nom de la ressource qui sera saisie et / ou libérée. S'applique uniquement lorsque Type est Resource.
Set Name	Nom de l'ensemble de ressources dont un membre sera saisi et / ou libéré. S'applique uniquement lorsque le type est set
Quantity	Nombre de ressources d'un nom donné ou d'un ensemble donné qui seront saisies / libérées. Pour les ensembles (set).
Selection rule	Méthode de sélection parmi les ressources disponibles dans un ensemble (set). Cyclical parcourra les membres des ressources disponibles (par exemple, 1er membre - 2e membre - 3e membre - 1er membre - 2e membre - 3e membre). Random sélectionnera un membre au hasard. Preferred order sélectionnera toujours le premier membre disponible (1er membre, si disponible ; puis 2ème membre, si disponible ; puis 3ème membre, etc.). specific member nécessite une valeur d'attribut d'entrée pour spécifier quel membre de l'ensemble (précédemment enregistré dans le champ save attribute). Largest Remaining Capacity et Smallest Number Busy sont utilisés pour les ressources à capacité multiple. S'applique uniquement lorsque le type est défini.
Save attribute	Nom d'attribut utilisé pour enregistrer le numéro d'index dans l'ensemble de ressources. Cet attribut peut être référencé ultérieurement avec la règle de sélection de membre spécifique. Ne s'applique pas lorsque la règle de sélection est specific member . Si Action est spécifiée comme Delay Release, la valeur

	spécifiée définit le membre (le numéro d'index) de l'ensemble à libérer. Si aucun attribut n'est spécifié, l'entité libérera le membre de l'ensemble qui a été saisi en dernier.
Set index	Le numéro d'index dans l'ensemble du membre demandé. S'applique uniquement lorsque la règle de sélection est specific member . Si Action est spécifiée comme Seize Delay Release, la valeur spécifiée définit quel membre (le numéro d'index) de l'ensemble doit être saisi et libéré.

Tableau 1.2 description boîte de dialogue *ressources*

1.3/ la sortie du système

La sortie du système est modélisée par le module *dispose*.

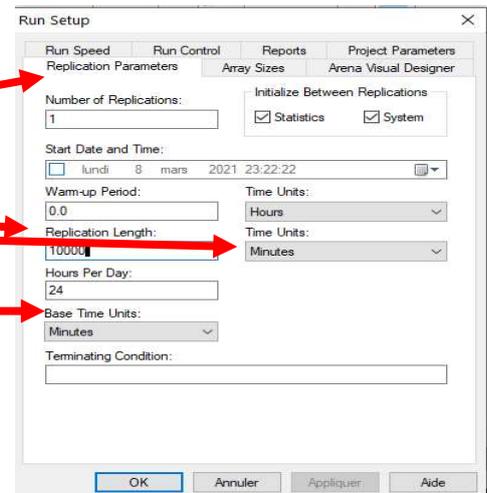


Ce module est conçu comme le point de fin pour les entités dans un modèle de simulation. Les statistiques d'entité peuvent être enregistrées avant la cession de l'entité.

2/ Simulation

La durée de simulation est 10 ,000 minutes, pour entrer cette valeur

- clique sur Run dans la barre de menu
- clique *setup* une boîte de dialogue s'ouvre
- Choisissez l'onglet *Replication Parameters*
- Entrez la durée de simulation dans le champ *Replication length* avec unité du temps
- Pour avoir des résultats de base en min, changez l'unité dans le champ *base time units*



- Lancer la simulation en utilisant

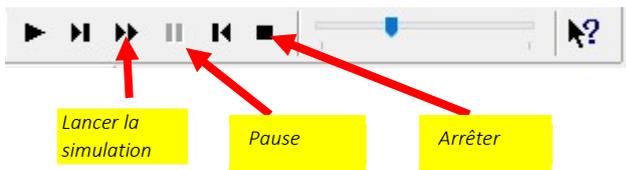


Figure1. 5 paramètres de simulation

3/ Le rapport

Le résultat final d'une simulation est un ensemble de statistiques requises, telles que les temps de traitement moyen, temps d'attente moyens, la taille et le temps d'attente moyen dans les stocks tampons, etc. Ceux-ci seront appelés résultats d'exécution. Arena fournit un nombre considérable de statistiques par défaut dans un rapport, généré automatiquement à la fin d'une simulation. Des statistiques supplémentaires peuvent être obtenues en ajoutant des modules de collecte de

statistiques dans le modèle, tels que *Record* (bibliothèque *Advanced Process*) et statistique (panneau de modèle de processus avancé).

Les résultats de l'analyse d'une seule réplication du modèle de l'exemple sont affichés dans la figure 1.6

- Lorsque je clique sur *resource* les résultats concernant les ressources s'affiche voir figure 1.7
- Lorsque je clique sur *queue* les résultats concernant queue s'affiche voir figure 1.8
- Lorsque je clique sur *Entity* les résultats concernant queue s'affiche voir figure 1.9

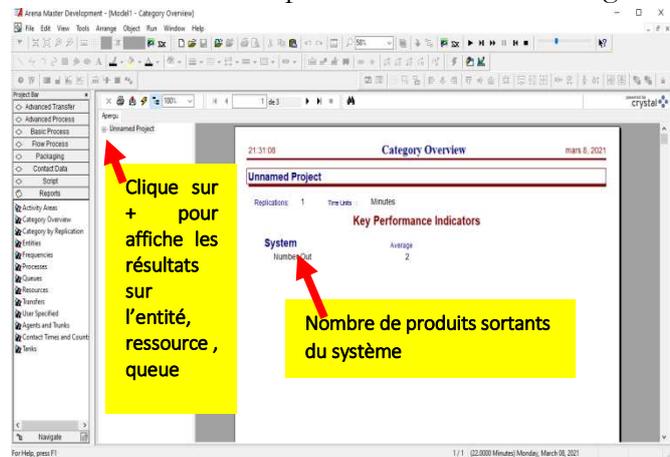


Figure1. 6 résultat de la réplication

Resource				
Usage				
Instantaneous Utilization	Average	Half Width	Minimum Value	Maximum Value
machine	0.8005	0.000587264	0.00	1.0000
Number Busy	Average	Half Width	Minimum Value	Maximum Value
machine	0.8005	0.000587264	0.00	1.0000
Number Scheduled	Average	Half Width	Minimum Value	Maximum Value
machine	1.0000	(Insufficient)	1.0000	1.0000
Scheduled Utilization	Value			
machine	0.8005			
Total Number Seized	Value			
machine	20005282			

Figure1. 7 rapport de la ressource

La figure 1.7 affiche la section *Ressources*, qui comprend des statistiques persistantes dans le temps d'utilisation des ressources. Observez que l'élément *Number Busy* fait référence au nombre d'unités occupées de la ressource, tandis que l'élément *number Scheduled* fait référence au nombre des ressources disponibles. L'élément *scheduled Utilization* se rapporte à l'utilisation par unité de ressource, à savoir *Number Busy* divisé par *Number Scheduled*.

Queue				
Time				
Waiting Time	Average	Half Width	Minimum Value	Maximum Value
Process 1.Queue	96.2379	0.530684313	0.00	1951.88
Other				
Number Waiting	Average	Half Width	Value	Maximum Value
Process 1.Queue	3.2088	0.018632555	0.00	72.0000

Figure1. 8 rapport de la file d'attente

La figure 1.8 affiche la section Files d'attente, qui comprend des statistiques orientées pièces (moyennes des clients), qui incluent les temps d'attente moyens dans les files d'attente y compris les tailles moyennes des files d'attente.

Entity				
Time				
VA Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	24.0078	0.012541577	0.00000095	440.45
NVA Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	0.00	0.000000000	0.00	0.00
Wait Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	96.2379	0.530684313	0.00	1951.88
Transfer Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	0.00	0.000000000	0.00	0.00
Other Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	0.00	0.000000000	0.00	0.00
Total Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	120.25	0.536964834	0.00000155	1959.22

Figure1. 9 rapport de l'entité

La figure 1.9 affiche la section Entity , qui comprend des statistiques orientées pièces, qui incluent le temps de traitement moyensdes pièce(*va time*) , temps d'attente (*wait time*)et aussin le temps de cycle ou le temps de séjour (*total time*) .

4/ Etude mathématique

Ce système de file d'attente à serveur unique est une situation très courante dans la pratique. En fait, cette situation a été étudiée mathématiquement à travers une branche de la recherche opérationnelle appelée théorie des files d'attente. Pour des situations de modélisation spécifiques, des formules pour le calcul des performances des systèmes peuvent être développées. Cet exemple de file d'attente

M / M / 1. Le premier M représente les arrivées de Markov¹, le second M représente le temps de traitement de Markov et le 1 représente un serveur unique. le nombre attendu de clients en file d'attente, Lq , pour le modèle M / M / 1 est donné par l'équation suivante :

$$Lq = \frac{\rho^2}{1-\rho}$$

$$\rho = \frac{\lambda}{\mu}$$

$$\text{Tel que } \lambda = \text{le taux d'arrivée} = \frac{1}{30}$$

$$\text{Et } \mu = \text{le taux de service} = \frac{1}{24}$$

De plus, le temps d'attente attendu dans la file d'attente Wq est donné par

$$Wq = \frac{Lq}{\lambda}$$

Application numérique

$$\text{le taux d'utilisation de la machine } \rho = \frac{\lambda}{\mu} = \frac{24}{30} = 0.8 * 100 = 80\%$$

$$\begin{aligned} \text{le nombre moyen des pièces dans la file d'attente } Lq &= \frac{\rho^2}{1-\rho} = \frac{0.8^2}{1-0.8} \\ &= 3,2 \text{ pièces} \end{aligned}$$

$$\text{le temps d'attentemoyen dans la file d'attente } Wq = \frac{Lq}{\lambda} = \frac{3.2}{1/30} = 96 \text{ min}$$

En comparant ces résultats analytiques avec les résultats de simulation, vous pouvez voir qu'ils correspondent à une erreur statistique. Ces résultats analytiques sont disponibles pour ce cas particulier car les temps d'arrivée et de traitement sont exponentielles ; cependant, des résultats analytiques simples ne sont pas disponibles pour de nombreuses distributions courantes, par exemple loi-normales.

Avec la simulation, vous pouvez facilement estimer les mesures de performances ci-dessus ainsi que de nombreuses autres mesures de performances intéressantes pour de nombreuses situations de file d'attente. Par exemple, grâce à la simulation, vous pouvez facilement estimer la probabilité qu'il y ait trois pièces ou plus en attente.

¹ Markov était un mathématicien célèbre qui a examiné la distribution exponentielle et ses propriétés. Selon la théorie de la file d'attente.

5/ Animation Graphique

Il est possible de modifier le visuel des entités et objets du système. Pour ce faire, il faut aller sur le petit tableau « Entity » de la section « Basic Process ». Une entité est déjà définie (Entity 1). On peut changer l'image associée à cette entité. Pour avoir balle rouge, il suffit de sélectionner *Picture.Red Ball* dans le menu déroulant correspondant, voir figure 1.10

Entity - Basic Process									
Entity Type	Initial Picture	Holding Cost / Hour	Initial VA Cost	Initial NVA Cost	Initial Waiting Cost	Initial Tran Cost	Initial Other Cost	Report Statistics	
1	Entity 1	Picture.Red Ball	0.0	0.0	0.0	0.0	0.0	0.0	<input checked="" type="checkbox"/>

Double-click here to add a new row.

Figure1. 10 donnée de l'entité

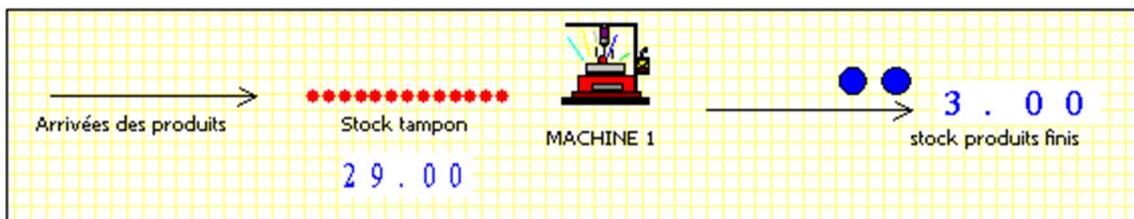
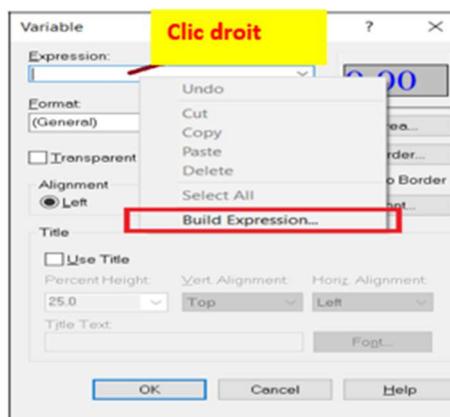
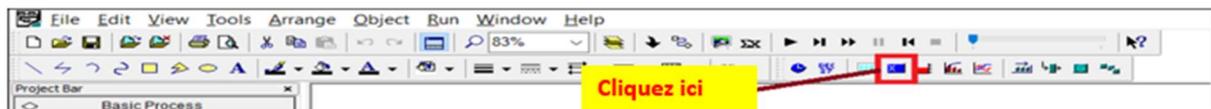


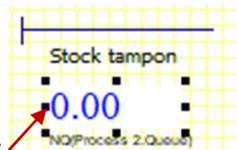
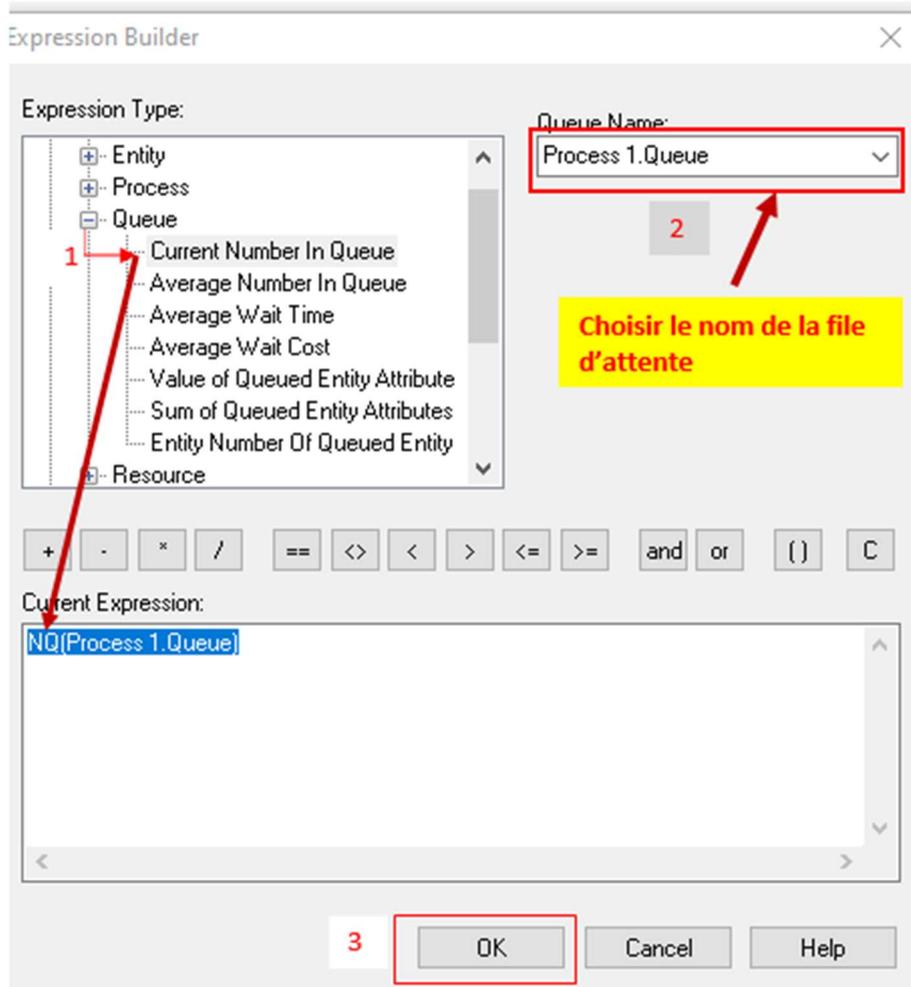
Figure1. 11 animation graphique 2D

Comment puis-je regarder une variable dans la simulation (comme NQ) étape par étape ?

NQ (nom_file d'attente) cette variable calcule le nombre des entités dans la file d'attente nommée « nom_file d'attente»

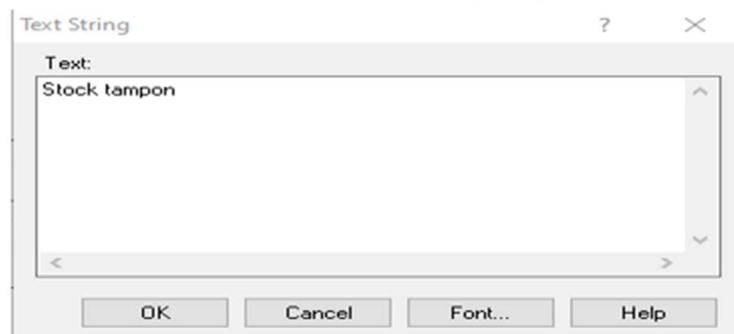
Suivez ces étapes :



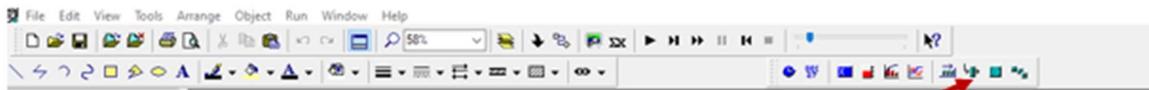


En fin dessiner un carré

Pour ajouter un text cliquer ici



Il est possible aussi de faire une animation 2D pour les ressources pour la faire suivre ces étapes :



CLIQUER ICI



Identifier: Resource 1
 State: Busy
 Picture ID: Machine.Drill

Current Library: c:\...\machines.plb

Identifier la ressource

2

Choisir le statut par exemple idl.

1 Choisir une image

3

Size Factor: 1 Auto Scale

Effects

When multiple pictures are defined for the same resource state, use this simulation timing to animate a series of pictures:

Hours per picture

Rotate By Expression:

Seize Area

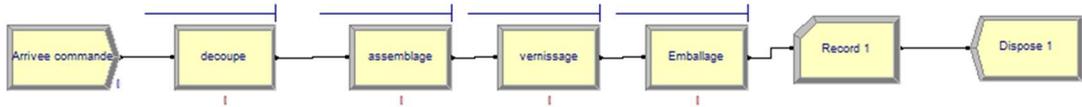
OK Cancel Help

Refaire les mêmes étapes pour le statut busy

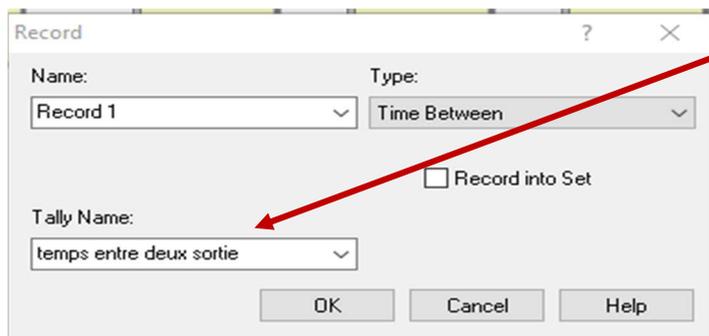
Solution TP°2 | TRAITEMENT EN SERIE, TRAITEMENT EN PARALLELE

1. Scénario-01 :

1/Modélisation



Pour avoir des statistiques sur le temps entre deux sorties, nous ajoutons un module *Record* avec type Time between. Qui sera enregistré dans tally Name « *temps entre deux sortie* »



Et le **taux de production** = $1 / \text{temps entre deux sortie}$ qui représente la quantité de produit fabriqué par unité du temps (Taux de production)

2/ simulation

Nous simulons le système sur une durée de 160 h avec 8H par jour, voir figure1.5

3/ Résultat de la simulation

Statistique	moyenne	Min	Max	Unité
Temps de traitement	5.0310	0.3144	0.3144	min
Temps de cycle	1743.89	1.5470	3440.93	min
Le temps d'attente	1738.86	0.00	3432.36	min
WIP	1381.64	0.00	2774.00	pièces
Le taux d'utilisations des ressources :				
• Mohammed		80,55 %		
• Ali		80 ,05%		
• Youcef		99 ,99%		
• Riad		52 ,27%		
Taux de production	1/1 ,9724=0,5 pièce /min			

2. Scénario-02 : On introduit le principe de la période transitoire

Si vous avez spécifié une période transitoire, les données collectées pendant cette période ne sont pas utilisées dans le calcul de l'intervalle de confiance.

Dans notre cas la période du régime transitoire égale à 1h.

A travers le menu Run/Setup/Replication Parameters, on peut fixer :

- la période du régime transitoire (champ Warp-up period) voir figure1.5

Statistique	moyenne	Min	Max	Unité
Temps de traitement	5.0291	0.3144	18.7389	min
Temps de cycle	1753.54	12.6679	3440.93	min
Le temps d'attente	1748.51	4.7132	3432.36	min
WIP	1390.29	9.0000	2774.00	pièces
Le taux d'utilisations des ressources :				
• Mohammed		80,15 %		
• Ali		80 ,62%		
• Youcef		100%		
• Riad		52 ,2%		
Taux de production	1/1 ,9724=0,5 pièce /min			

3. Scénario-03 : On augmente, par rapport au scénario-02, à 10 le nombre de réplifications

Dans Arena (lorsque le nombre de réplification est supérieur à 1), la demi-largeur (half width) se rapporte à un intervalle de confiance dont le seuil est égal à 95%.

A travers le menu *Run/Setup/Replication Parameters*, on peut notamment fixer :

- le nombre de réplifications (champ Number of Replications),

Statistique	moyenne	Min	Max	Unité
Temps de traitement	5.3641	0.01014	5.6932	Min
Temps de cycle	14.330	1.5469	21.354	Min
Le temps d'attente	8.9667	.00000	18.239	Min
WIP	7.6330	.00000	12.000	pièce

Le taux d'utilisations des ressources :	
• Mohammed	64,768 %
• Ali	69,256%
• Youcef	81 ,683%
• Riad	59 ,730%
Taux de production	1 /2.1835=0.45pièce/min

4. Scénario-04 : On ajoute, par rapport au scénario-03, un module Decide

Modélisation

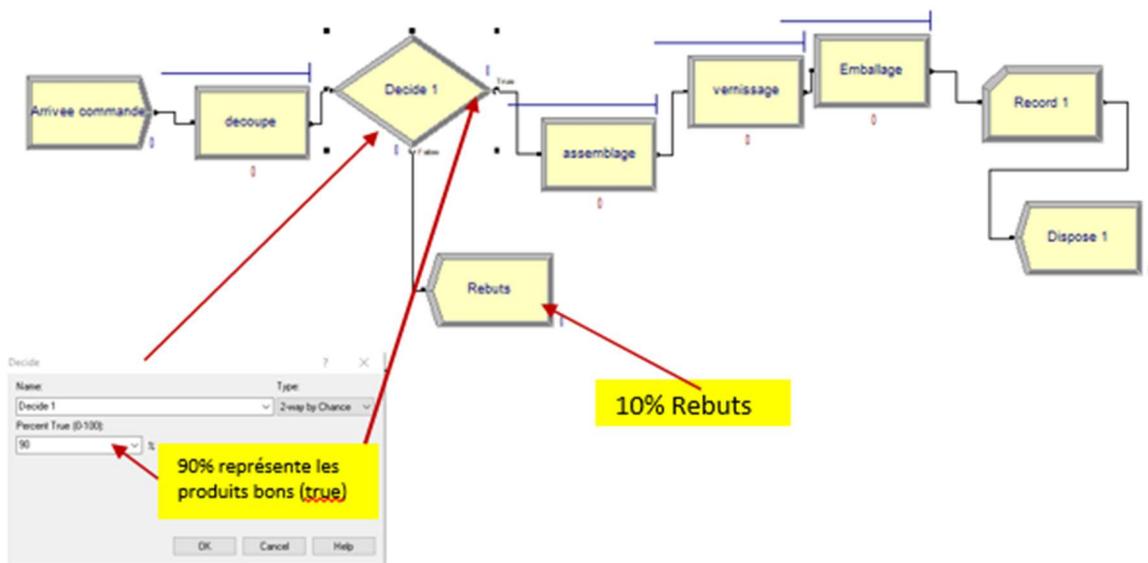


Figure2. 1 Modélisation scénario 4

Résultats

Statistique	moyenne	Min	Max	Unité
Temps de traitement	5.0240	0.12489	14.491	Min
Temps de cycle	15.023	0.13742	31.079	Min
Le temps d'attente	9.9993	0	27.540	Min
WIP	12.908	0	25.000	Pièce
Le taux d'utilisations des ressources :				
• Mohammed		87,622%		
• Ali		71 ,262%		

• Youcef	90 ,521
• Riad	37,659%
Taux de production	1/2 ,21=0 .45 pièce/min

5. Scénario-05: le parallèle

Modélisation



Figure2. 2 modèle ressource parallèle avec option Random



A travers le bloc « *Set* » de la bibliothèque « *Basic process* » nous introduisons les deux ressources Youcef et Mehdi qui sont dans l'ensemble nommé set name « *ensemble ressource* » voir figure 2.3

Figure2. 3 bloc set

Résultats :Random

Statistique	moyenne	Min	Max	Unité
Temps de traitement	5.0356	0.12489	9.9874	min
Temps de cycle	9.2528	0.13742	14.527	min
Le temps d'attente	4.2171	0.00	10.212	min
WIP	6.0701	0.00	14.000	pièces
Le taux d'utilisations des ressources :				
• Mohammed			64,668 %	
• Ali			70,629%	
• Youcef			66,201%	
• Riad			61,152%	
• Mehdi			60,997%	
Taux de production	1/ 1.6854=0.59 pièce /min			

Résultats : Cyclical

Statistique	moyenne	Min	Max	Unité
Temps de traitement	4.3332	0	10.706	Min
Temps de cycle	9.3592	0.13742	18.457	Min
Le temps d'attente	5.0260	0	13.557	Min
WIP	5.6660	0	13.000	Pièce
Le taux d'utilisations des ressources :				
• Mohammed			60,69%	
• Ali			58,90%	
• Youcef			42,718%	
• Mehdi			53,61%	
• Riad			42,58%	
Taux de production	1/2.2146=0.45 pièce/min			

6. Scénario-0-6: le parallèle avec temps de traitement différent

Modélisation

Le même modèle sauf que nous allons ajouter un attribut nommé « *num* » qui va sauvegarder l'indice de la ressource dans set. Pour ajouter cet attribut voir figure 2.4

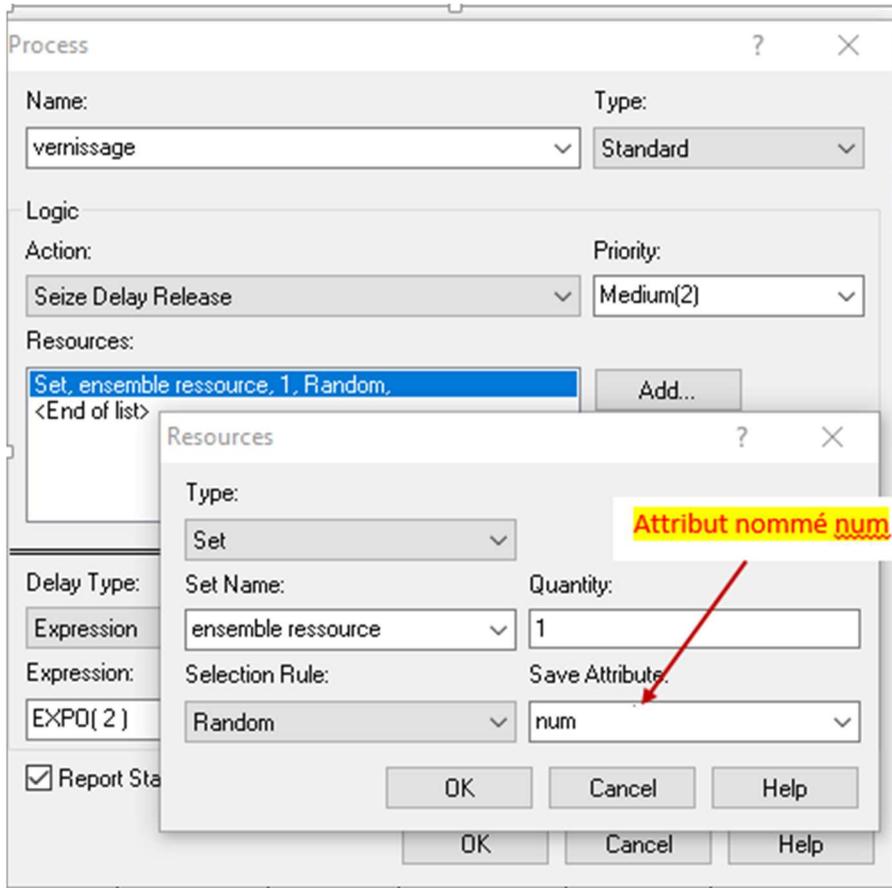


Figure2. 4 attribut dans ressources

Afin d'attribuer le temps pour chaque ressource nous utilisons une expression booléenne dans le champ delay de process 'vernissage'

Process - Basic Process										
	Name	Type	Action	Priority	Resources	Delay Type	Units	Allocation	Expression	Report Statistics
1	decoupe	Standard	Seize Delay Release	Medium(2)	1 rows	Expression	Minutes	Value Added	EXPO(1)	<input checked="" type="checkbox"/>
2	assemblage	Standard	Seize Delay Release	Medium(2)	1 rows	Expression	Minutes	Value Added	EXPO(1)	<input checked="" type="checkbox"/>
3	vernissage	Standard	Seize Delay Release	Medium(2)	1 rows	Expression	Minutes	Value Added	(num==1)/2+(num==2)/1	<input checked="" type="checkbox"/>
4	emballage	Standard	Seize Delay Release	Medium(2)	1 rows	Expression	Minutes	Value Added	EXPO(1)	<input checked="" type="checkbox"/>

Tel que num ==1 représente la première ligne dans SET

num == 2 représente la deuxième ligne dans SET

} Voir figure 2.4

Résultats

Statistique	moyenne	Min	Max	Unité
Temps de traitement	4.1001	0	22.041	Min
Temps de cycle	13.974	0.00507	47.613	Min

Le temps d'attente	9.8740	0	42.312	Min
WIP	11.085	0	43.000	Pièce
Le taux d'utilisations des ressources :				
• Mohammed	80,17%			
• Ali	71,81%			
• Youcef	57,88%			
• Mehdi	71,68%			
• Riad	43,75%			
•				
Taux de production	$1/1.3953=0.71$ pièce /min			

Solution TP°3 ETUDE DES PANNES DES MACHINES et DES CALENDRIERS

Partie 1 : Calendrier d'arrivée

Nos TP's ont jusqu'à présent supposé que les phénomènes aléatoires (par exemple, les arrivées, les services, etc.) sont modélisés comme des variables à partir d'une loi de probabilité fixe qui ne change pas dans le temps, de sorte que le processus sous-jacent est stationnaire. Cependant, il est assez courant dans la pratique que la loi de probabilité varie dans le temps, auquel cas le processus est non stationnaire (dépendant du temps).

1/Modélisation

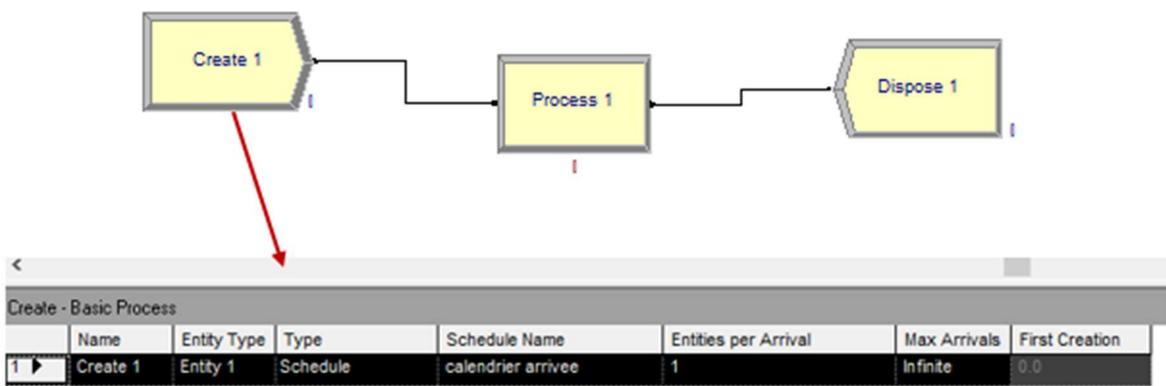


Figure3. 1 modélisation TP3 partie 1

Ce module de données peut être utilisé en conjonction avec le module *Ressource* pour définir un programme d'exploitation d'une ressource ou avec le module *create* pour définir un programme d'arrivée. Voir figure 3.2

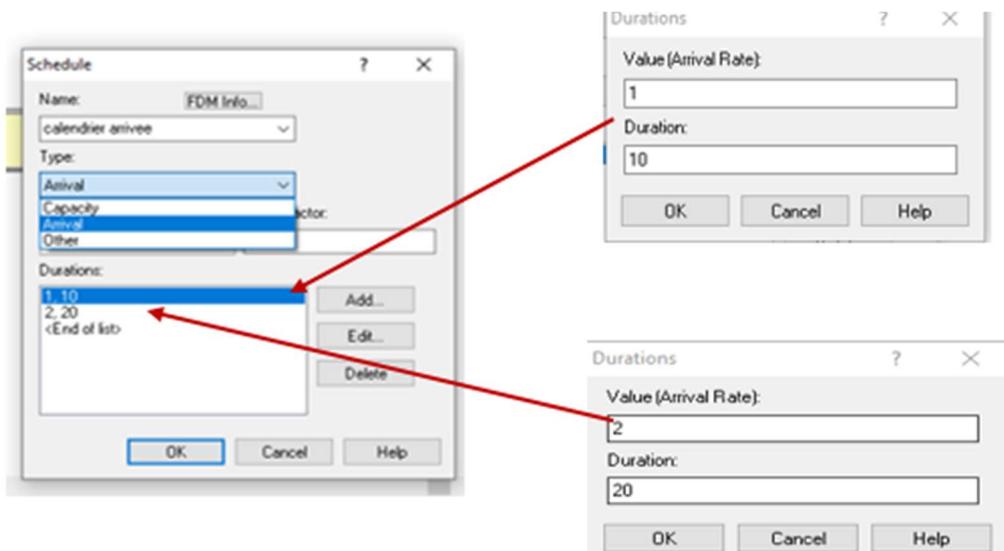


Figure3. 2 les données de bloc schedule

Simulation

Nous simulons le système sur une période de 8H

Résultats

- Le temps de traitement des produits =**10min**
- Le taux d'utilisation de la machine =**25%**
- La taille du stock tampon moyenne =**0 .10 pièce**
- La durée d'attente moyenne des produits dans la file d'attente = **4.03 min**

Partie 2 : Panne des machines

Nous ajoutons à la définition de la ressource une panne tel que le temps de bon fonctionnement =**8 min** et le temps de panne = **3min**

Modélisation

La modélisation de la panne est bien décrite dans l'énoncé de TP

Résultats

1 /Choix de l'option « **Ignore** »

- Le temps de traitement du produit= 10min
- Le taux d'utilisation de la machine =**33 .33%**
- La taille du stock tampon moyenne=**0 pièce**
- La durée d'attente moyenne du produit. =**0 min**
- Le temps de cycle =**10 min**

2/ Choix de l'option « **Wait** »

- Le temps de traitement du produit= 10min
- Le taux d'utilisation de la machine =**33,33%**
- La taille du stock tampon moyenne=**0**
- La durée d'attente moyenne du produit. =**0**
- Le temps de cycle =**10 min**

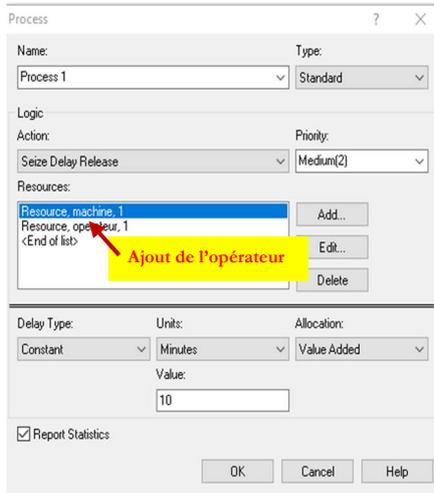
3/ Choix de l'option « **Preempt** »

- Le temps de traitement du produit= 10min
- Le taux d'utilisation de la machine =**33,33%**
- La taille du stock tampon moyenne=**0**
- La durée d'attente moyenne du produit. =**0**
- Le temps de cycle =**13min**

Partie 3 : Pause de l'employeur

Temps entre deux arrivées = expo(10) min

Ajouter une nouvelle ressource nommée operateur dans le même process .voir figure 3.3



- **NB** : Pour ajouter la pause de l'opérateur voir l'énoncé de TP

Figure3. 3 Module Process

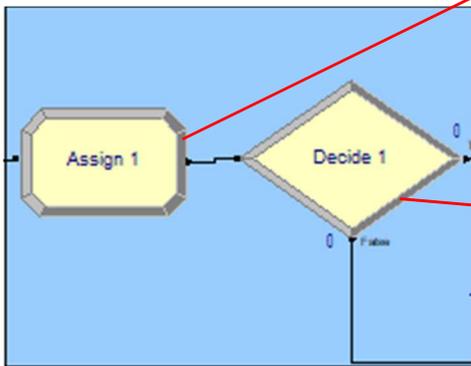
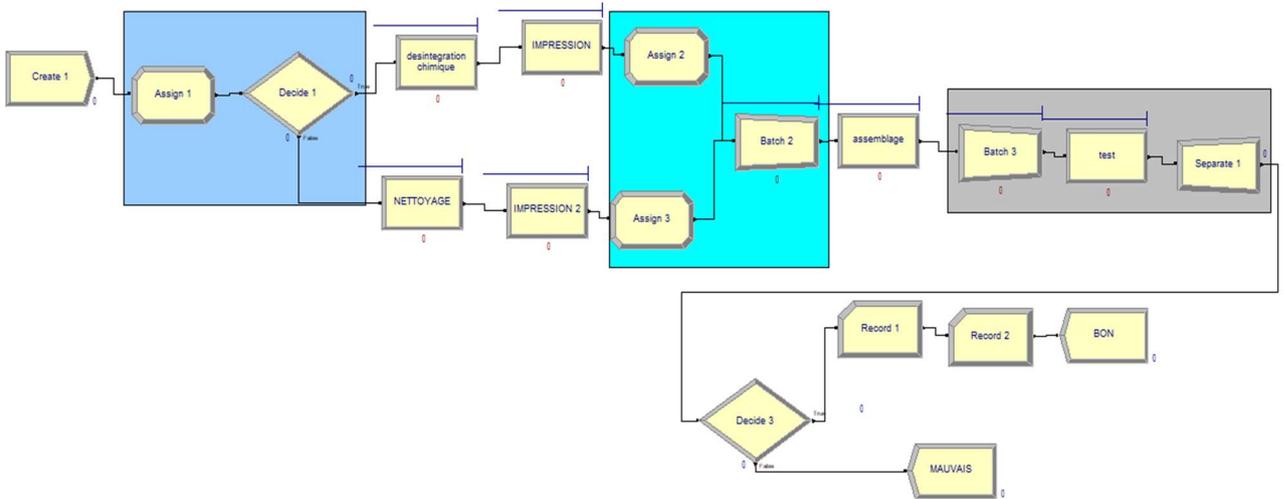
Résultats

Après simulation sur une période de 8H

- Le temps de traitement des produits = 10 min
- Le taux d'utilisation de la machine =70,63%
- Le taux d'utilisation de l'opérateur =70,63%
- La taille moyenne du stock tampon =4.5929 pièces
- La durée d'attente moyenne du produit=55.6792 min

Solution TP°4 | POSTE D'ASSEMBLAGE

Modélisation



Assign

Name: Assign 1

Assignments:

- Attribute, TYPE, Disc[0.4.1.1.2]
- Attribute, temps de cycle, tnow
- <End of list>

Buttons: Add..., Edit..., Delete, OK, Cancel, Help

Decide

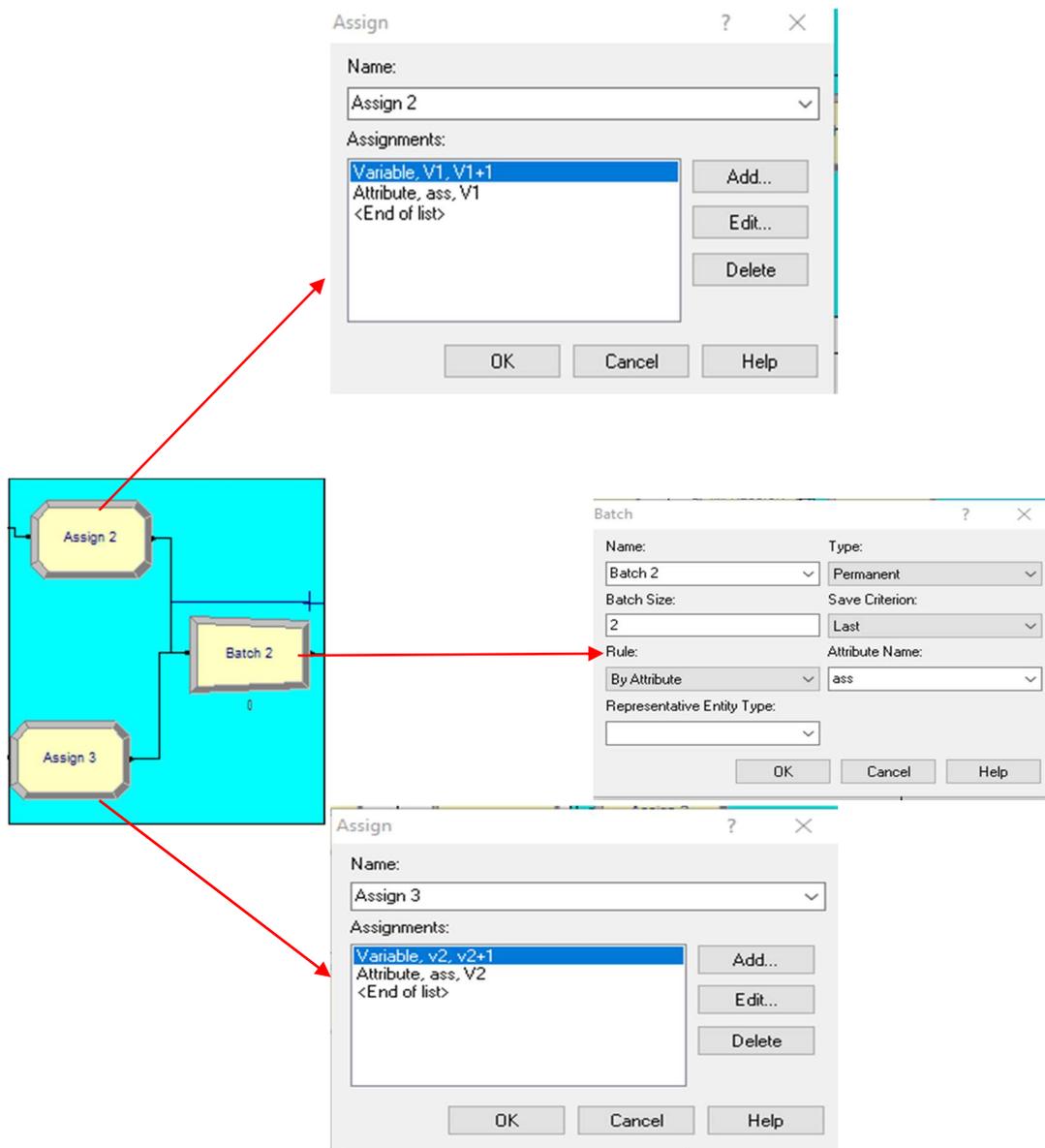
Name: Decide 1 Type: 2-way by Condition

If: Attribute Named: TYPE Is: ==

Value: 1

Buttons: OK, Cancel, Help

Cette partie pour modéliser l'assemblage



Résultats

- le temps de traitement moyen = 16.1904 H

VA Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	16.1904	0.078842492	13.6455	18.7170

- les taux d'utilisation des machines :

Instantaneous Utilization	Average	Half Width	Minimum Value	Maximum Value
desintegration machine	0.8268	(Correlated)	0.00	1.0000
IMPRIMANTE	0.8535	0.005855935	0.00	1.0000
machiene de nettoyage	0.7958	0.009186905	0.00	1.0000
machine assemblage	0.1824	0.002299044	0.00	1.0000
machine essai	0.2893	(Correlated)	0.00	1.0000

- le temps d'attente moyen pour chaque file d'attente

Waiting Time	Average	Half Width	Minimum Value	Maximum Value
assemblage.Queue	0.05361462	0.010905716	0.00	0.9714
Batch 2.Queue	890.59	(Correlated)	0.00	3554.45
Batch 3.Queue	10.9710	0.167427631	0.00	31.5002
desintegration chimique.Queue	2907.65	(Correlated)	0.00	5709.77
IMPRESSION 2.Queue	2370.57	(Correlated)	0.6761	4745.96
IMPRESSION.Queue	2372.21	(Correlated)	0.00	4745.75
NETTOYAGE.Queue	3789.59	(Correlated)	0.00	7613.08
test.Queue	0.00	(Insufficient)	0.00	0.00

- le nombre moyen des pièces dans chaque file d'attente :

assemblage.Queue	0.00978038	0.002184589	0.00	1.0000
Batch 2.Queue	542.73	(Correlated)	0.00	1084.00
Batch 3.Queue	1.9998	0.020878214	0.00	5.0000
desintegration chimique.Queue	5627.61	(Correlated)	0.00	11282.00
IMPRESSION 2.Queue	943.94	(Correlated)	0.00	1887.00
IMPRESSION.Queue	1577.89	(Correlated)	0.00	3153.00
NETTOYAGE.Queue	11102.00	(Correlated)	0.00	22186.00
test.Queue	0.00	(Insufficient)	0.00	0.00

- le temps de cycle

Interval	Average	Half Width	Minimum Value	Maximum Value
cycle time	4106.00	(Correlated)	36.9882	8216.48

- le temps entre deux sorties

Between	Average	Half Width	Minimum Value	Maximum Value
temps entre deux sorties	6.8094	0.284701839	0.00	38.5000

- le nombre moyen des pièces dans le système

WIP	Average	Half Width	Minimum Value	Maximum Value
Entity 1	19808.96	(Correlated)	0.00	39603.00

Solution TP°5 | GESTION DES FILES D'ATTENTE

Partie ° 1 :

Modélisation

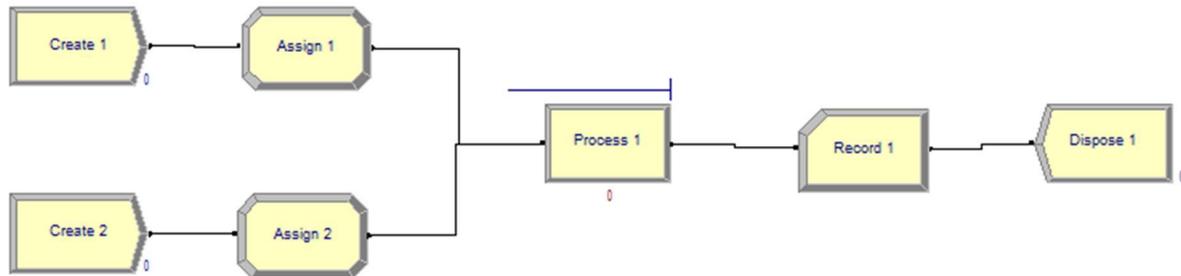


figure5. 1 Modèle partie 1

Résultats

Le temps de cycle moyen de la pièce A est de 4.6667 minutes

Le temps de cycle moyen de la pièce A est de 4.0000 minutes

Le temps de cycle moyen tout type de pièces confondu est de 4.2500 minutes

Le taux d'utilisation de la machine est de 0.9000

La taille moyenne du stock 0.2333 Pièces

Question N° 2 :

Modélisation

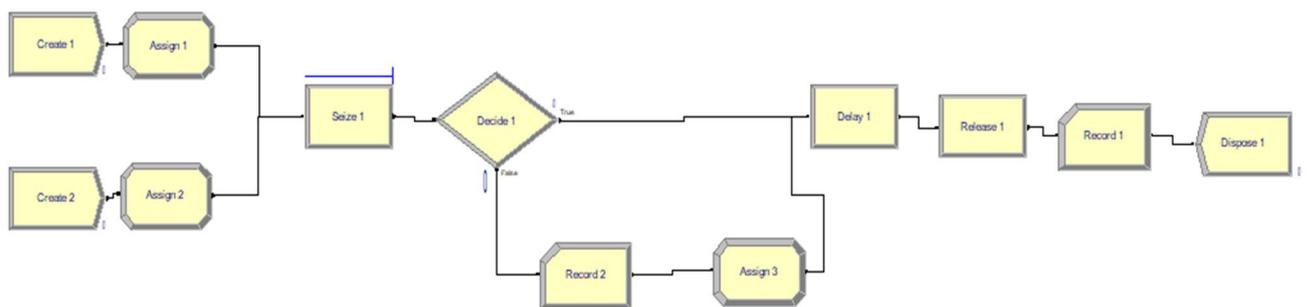


figure5. 2 Modèle partie 2

Résultats

Le temps de cycle moyen de la pièce A est de 199.50 minute

Le temps de cycle moyen de la pièce A est de 199.00 minute

Le temps de cycle moyen tout type de pièces confondu est de 199.20 minutes

Le taux d'utilisation de la machine est de 1.0000

La taille moyenne du stock 51.2188 pièces

Le nombre de changement est de 20

Question N° 3 :

Modélisation

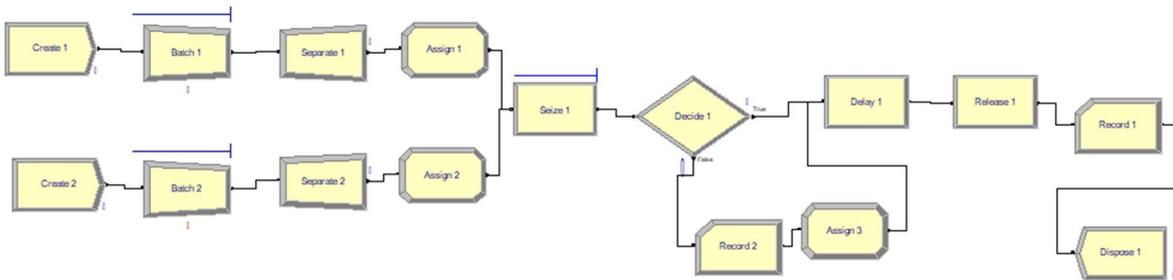


figure5. 3 Modèle partie 3

Résultats

En utilisant le process analyser (voir énoncé)

- **Définition process analyser** : Le Processus Analyser (PAN) est un assistant pour l'évaluation des solutions lors de l'exécution des différents scénarii du modèle de simulation. Il est très utile pour la simulation des modèles développés tout en donnant des résultats précis et en un temps très court. Son utilité apparait surtout lorsqu'il s'agit de prendre des décisions qui ont un impact sur l'avenir de l'entreprise.

Le PAN est axé principalement sur la comparaison des post-modèles issus des modèles précédents. A partir de là ; on peut dire que la simulation du modèle est complète, validée, et correctement configurée pour qu'elle soit par la suite prête à être utilisée par le processus PAN. Par conséquent, le processus PAN a pour rôle de réaliser la comparaison entre les différentes sorties à travers les modèles validés à partir des différentes entrées du modèle.

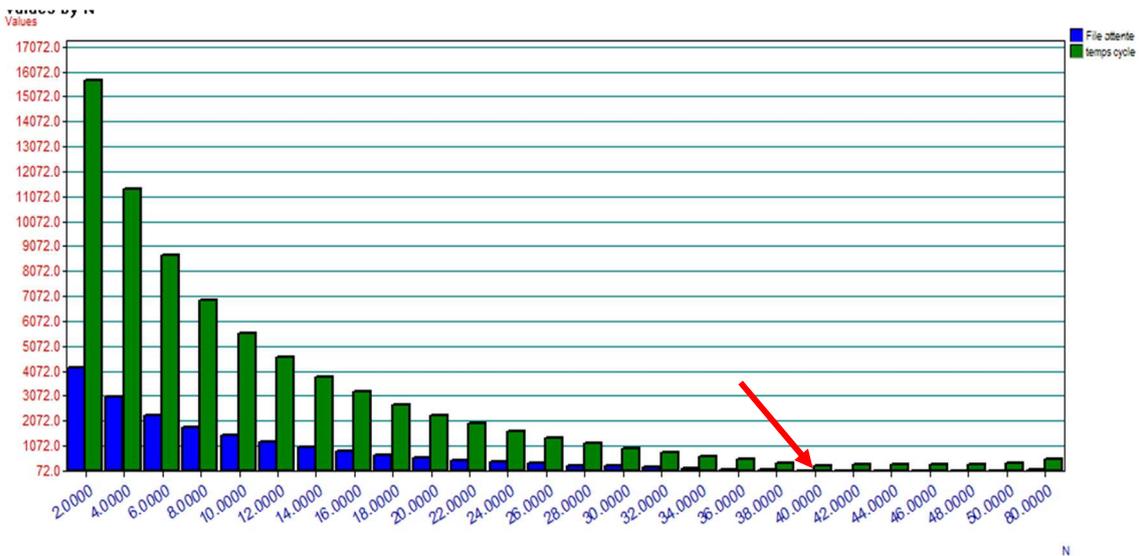
Ces dernières sont initiées par la commande « controls » quant aux sorties, elles sont initiées par la commande « responses ».

• **Définition des termes liés au PAN**

- **Controls** : ce sont les entrées affectant le fonctionnement du modèle de sorte que la sortie du modèle soit à la fois visualisée et contrôlée.
- **Responses** : ce sont les sorties représentant les mesures du comportement du modèle pendant son exécution.
- **Scénario** : un ensemble de commandes de types « controls » et « responses » appliqué à un modèle de simulation donné.

La figure suivante représente l'interface du processus Analyzer qui permet de visualiser la comparaison entre les différents scénarii

Scenario Properties					Control	Responses	
S	Name	Program File	Reps	N		File attente	temps cycle
1	Scenario 1	S: TP 3 Que	1	2.0000		4195.686	15752.934
2	Scenario 2	S: TP 3 Que	1	4.0000		3036.682	11400.375
3	Scenario 3	S: TP 3 Que	1	6.0000		2323.209	8725.811
4	Scenario 4	S: TP 3 Que	1	8.0000		1840.193	6913.048
5	Scenario 5	S: TP 3 Que	1	10.0000		1491.910	5606.790
6	Scenario 6	S: TP 3 Que	1	12.0000		1229.198	4620.034
7	Scenario 7	S: TP 3 Que	1	14.0000		1024.222	3852.491
8	Scenario 8	S: TP 3 Que	1	16.0000		860.030	3236.019
9	Scenario 9	S: TP 3 Que	1	18.0000		725.737	2731.937
10	Scenario 10	S: TP 3 Que	1	20.0000		613.980	2312.685
11	Scenario 11	S: TP 3 Que	1	22.0000		519.670	1958.300
12	Scenario 12	S: TP 3 Que	1	24.0000		439.108	1655.169
13	Scenario 13	S: TP 3 Que	1	26.0000		369.589	1394.354
14	Scenario 14	S: TP 3 Que	1	28.0000		310.325	1171.807
15	Scenario 15	S: TP 3 Que	1	30.0000		259.300	980.005
16	Scenario 16	S: TP 3 Que	1	32.0000		214.465	811.416
17	Scenario 17	S: TP 3 Que	1	34.0000		174.965	662.568
18	Scenario 18	S: TP 3 Que	1	36.0000		139.895	530.548
19	Scenario 19	S: TP 3 Que	1	38.0000		108.688	412.862
20	Scenario 20	S: TP 3 Que	1	40.0000		80.810	307.671
21	Scenario 21	S: TP 3 Que	1	42.0000		64.086	320.068
22	Scenario 22	S: TP 3 Que	1	44.0000		67.382	332.330
23	Scenario 23	S: TP 3 Que	1	46.0000		90.637	344.685
24	Scenario 24	S: TP 3 Que	1	48.0000		93.937	357.105

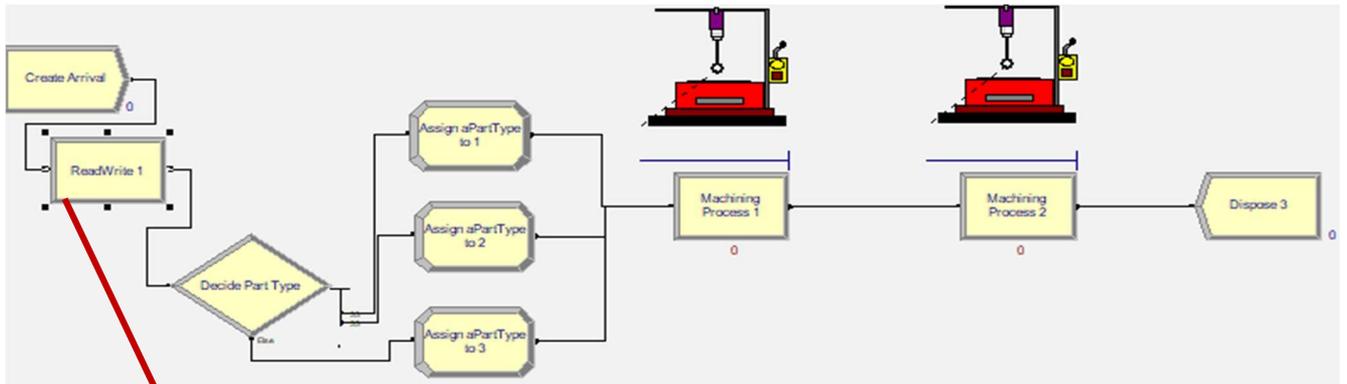


D'après l'analyse des scénarii effectuée par le PAN et les histogrammes confiés, on a constaté que **40** c'est lot optimal qui nous donne le min temps de cycle et le min nombre dans la file d'attente

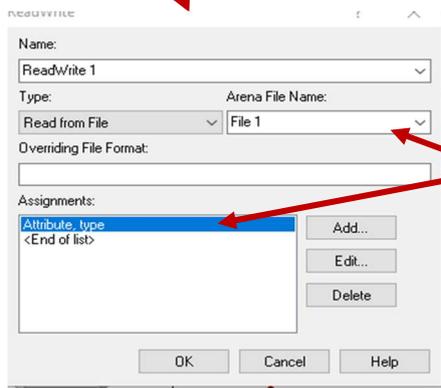
Solution TP°6 | LIRE / ECRIRE A PARTIR D'UN FICHIER

Partie 1 : lire à partir du fichier

Modélisation



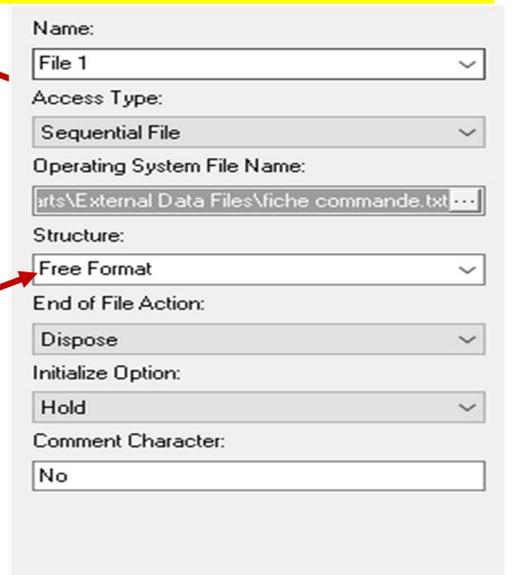
1. lecture à partir d'un fichier type txt



Chaque entité va lire le type de la commande qui est sauvegardé dans un fichier 'file 1' l'attribut nommé 'type'

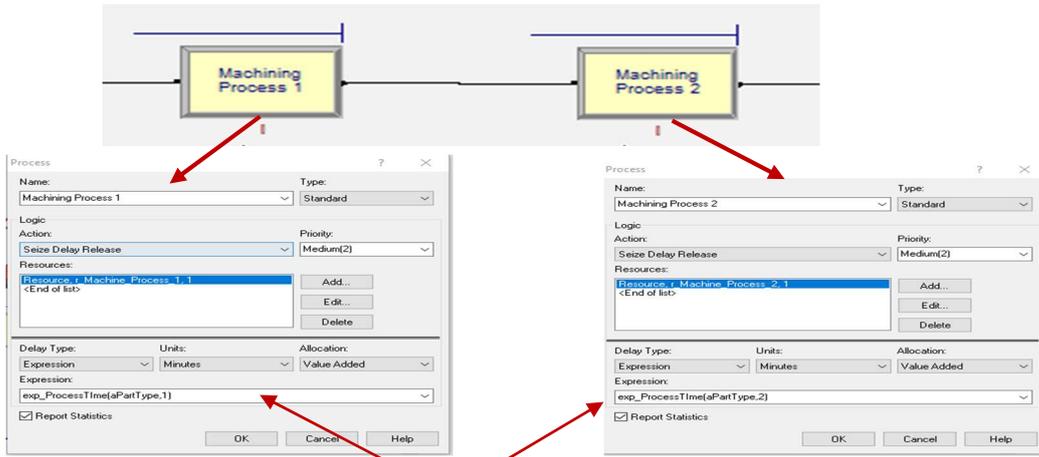
Dans la le module 'file' de la bibliothèque 'advanced proceses' on trouve le nom de fichier qui déjà déclaré dans le module **readwrite 1**.

Le chemin d'enregistrement



2. Lecture à partir d'un fichier Excel

Le temps de traitement et le temps de panne sont sauvegardés dans un fichier nommé 'traitement_panne.xls'



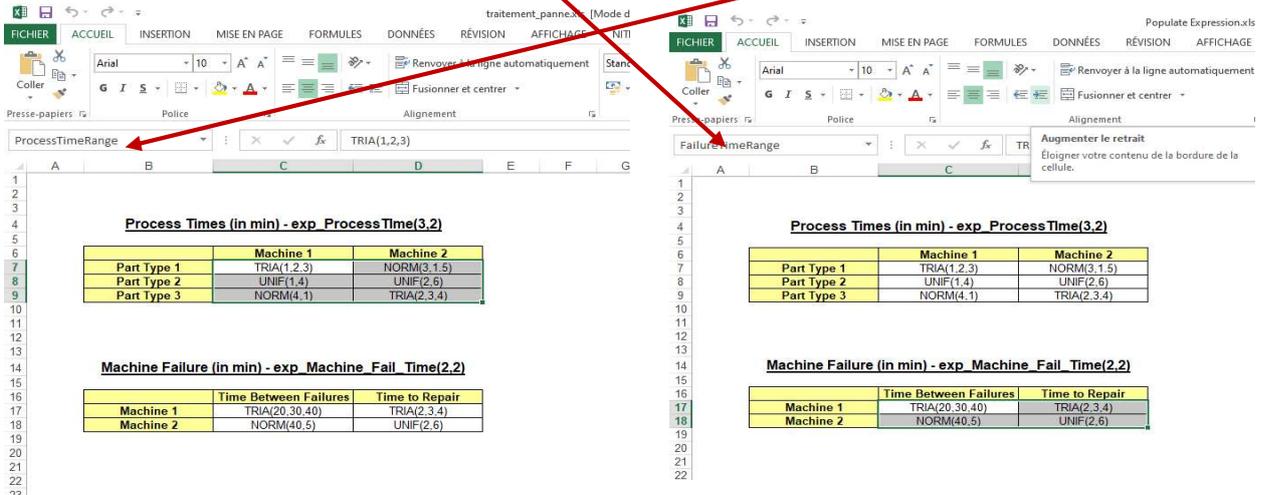
L'expression du temps est 'ex_processTime' est déclaré dans le bloc 'expression' de la bibliothèque 'advanced process' voir figure ci-dessus

	Name	Rows	Columns	Data Type	File Name	Recordset
1	exp_ProcessTime	3	2	Native	File Read Expressions From Excel	Recordset ProcessTimeRange
2	exp_Machine_Fail_Time	2	2	Native	File Read Expressions From Excel	Recordset FailureTimeRange

Les valeurs du temps de traitement et le temps de panne sont lu à partir d'un fichier excel nommé 'File Read Expressions From Exce' dont le recordset est nommé 'Recordset ProcessTimeRange' pour le temps de traitement et Recordset FailureTimeRange pour le temps de panne

Chemin d'enregistrement du fichier traitement_panne.xls

Le champ 'Named range' contient le nom donné dans le fichier Excel 'ProcessTimeRange' pour le temps de traitement et 'FailureTimeRange' pour le temps de panne



Résultats : unité en min

- Le temps de traitement moyen des commandes

VA Time		Average	Half Width	Minimum Value	Maximum Value
Part		5.9874	(Insufficient)	1.6016	9.3771

- Le temps d'attente moyen

Waiting Time		Average	Half Width	Minimum Value	Maximum Value
Machining Process 1.Queue		134.03	(Insufficient)	0.00	274.62
Machining Process 2.Queue		59.2917	(Insufficient)	0.00	104.66

- Le taux d'utilisation des machines

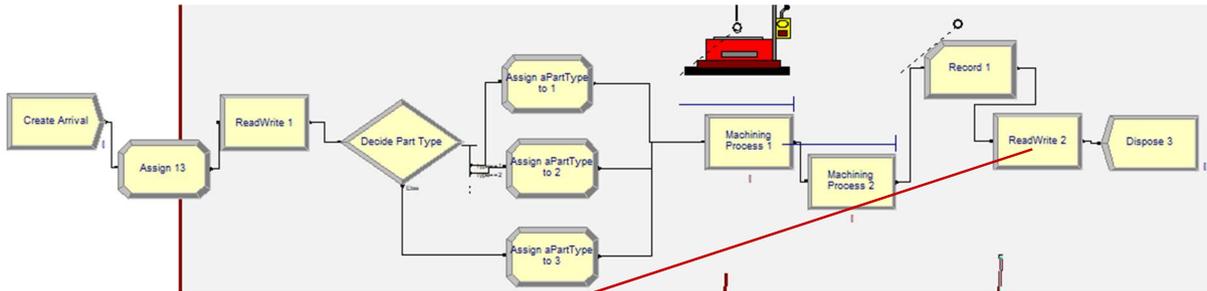
Instantaneous Utilization		Average	Half Width	Minimum Value	Maximum Value
r_Machine_Process_1		0.6124	(Insufficient)	0.00	1.0000
r_Machine_Process_2		0.8361	(Insufficient)	0.00	1.0000

- Le temps de séjour moyen des commandes dans le système =

Total Time		Average	Half Width	Minimum Value	Maximum Value
Part		199.91	(Insufficient)	2.6609	381.20

Partie 2 : Ecrire dans un fichier

Modélisation



On a ajouté un module 'Readwrite' pour écrire (sauvegarder) le temps de cycle de chaque entité (commande) dans un fichier type txt nommé cycle time. Ce dernier est calculé dans le module Record.

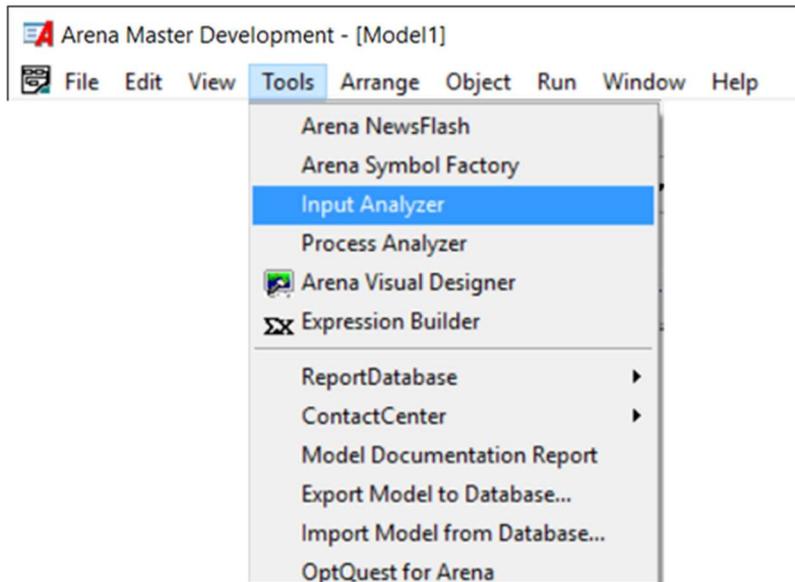
Résultats

➤ **Definition Input Analyser**

Est un composant standard de l'environnement Arena. Il est utilisé pour déterminer **la distribution de probabilité à des données.**

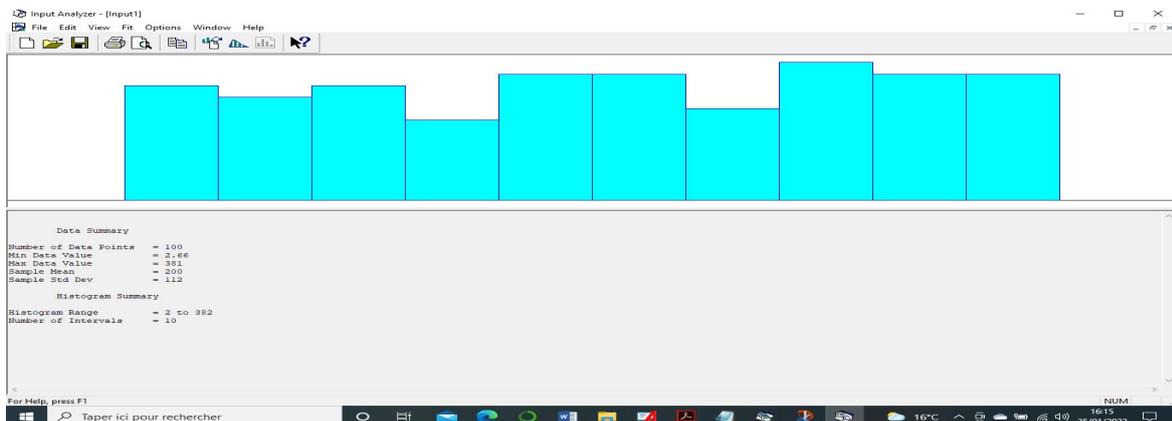
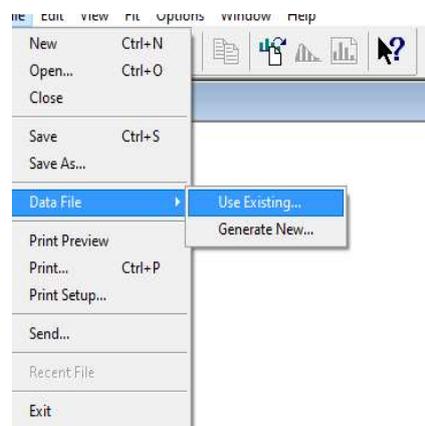
En plus **Input Analyser** peut générer des ensemble

Pour ouvrir **Input Analyser** : Tools >> **Input Analyser**

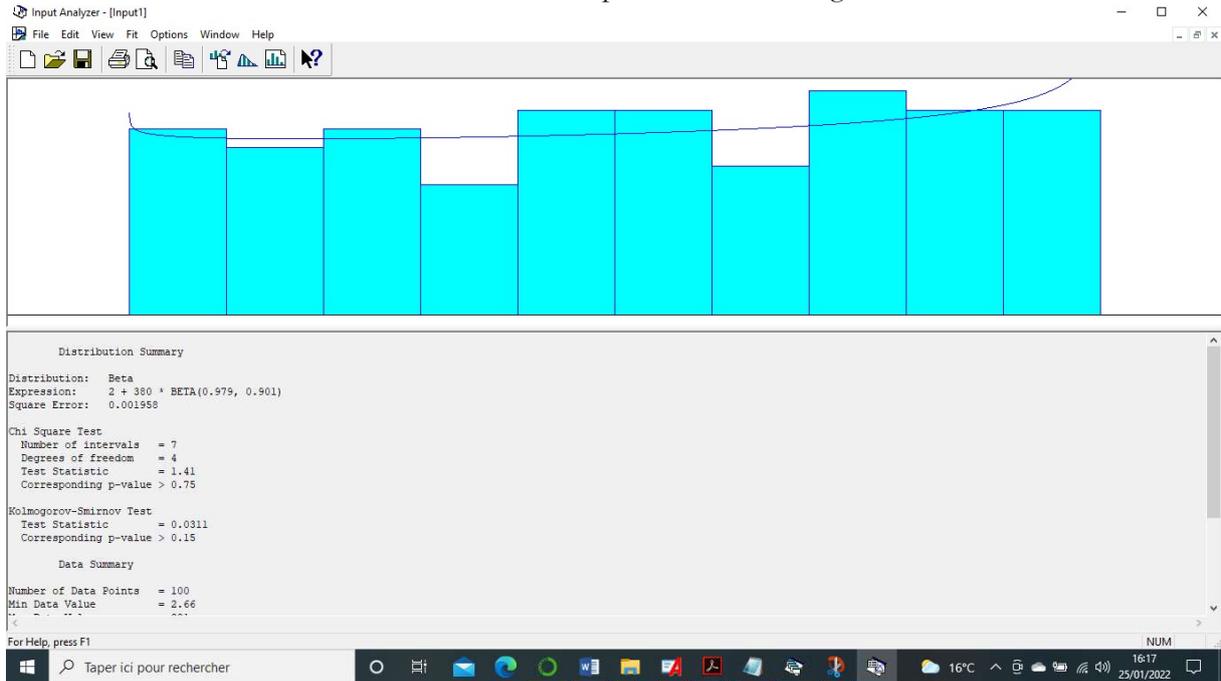


Pour déterminer la distribution de proba des données dans notre cas le temps de cycle s qui sont déjà existé dans un fichier nommé cycle time

File >> Data File >> Use Existing >> choisir le fichier 'cycle time '



Pour trouver la meilleure distribution correspondant à cet histogramme, faire : *Fit/Fit All*



Solution TP°7 | MOYEN DE MANUTENTION / ANIMATION 2D

1. Modélisation

Le système donné est modélisé dans le logiciel ARENA. La figure 7.1 représente modèle le pour l'atelier d'engrenages, composé de trois segments principaux :

1. Arrivées des entités
2. Transport des entités
3. Traitement des entités
4. Départ de l'entité

Voici une présentation du modèle, étape par étape

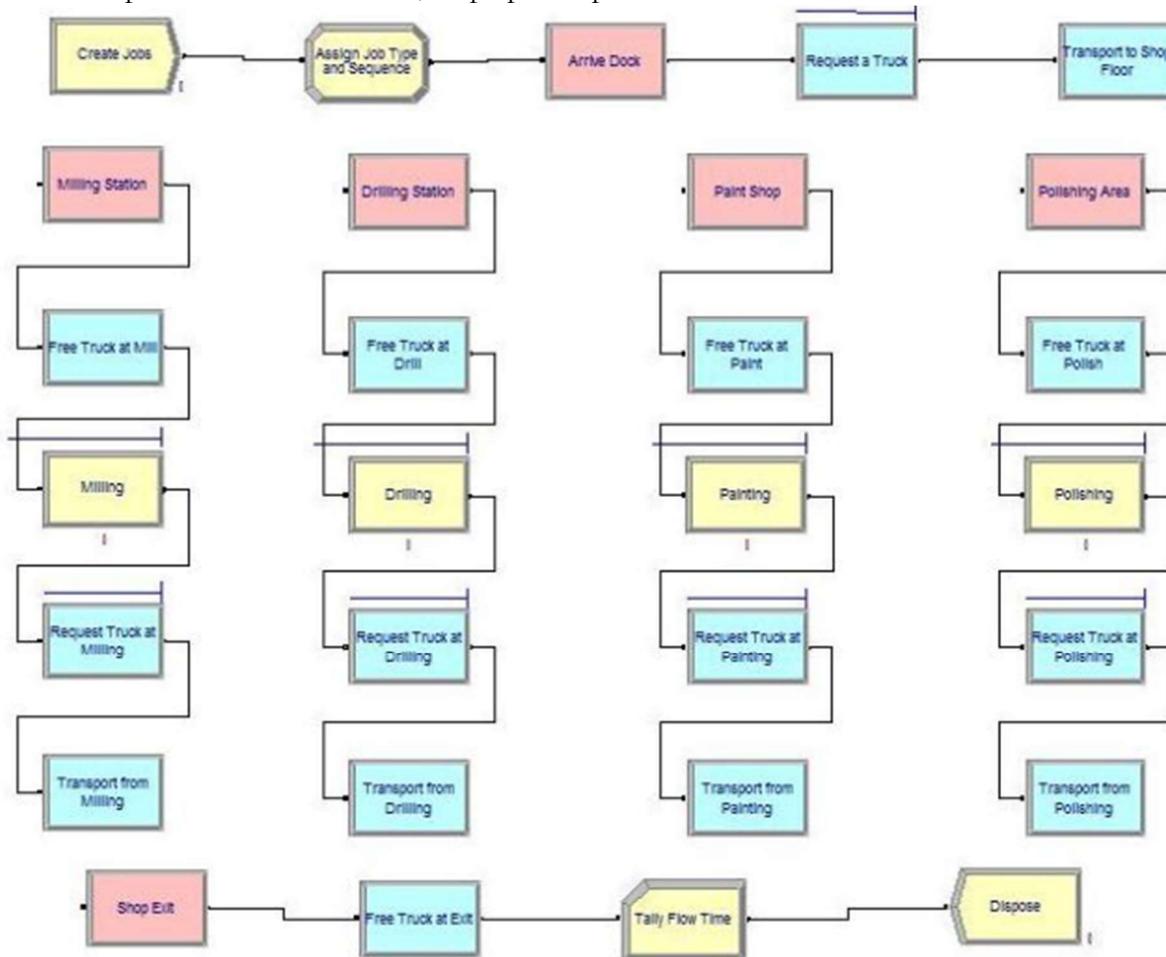


figure7. 1 modèle Arena TP 7

1.1 Arrivée des entités

Cette partie comprend la section d'arrivée des engrenages (Gear Jobs). Le sous modèle est illustré dans la figure 7.2.

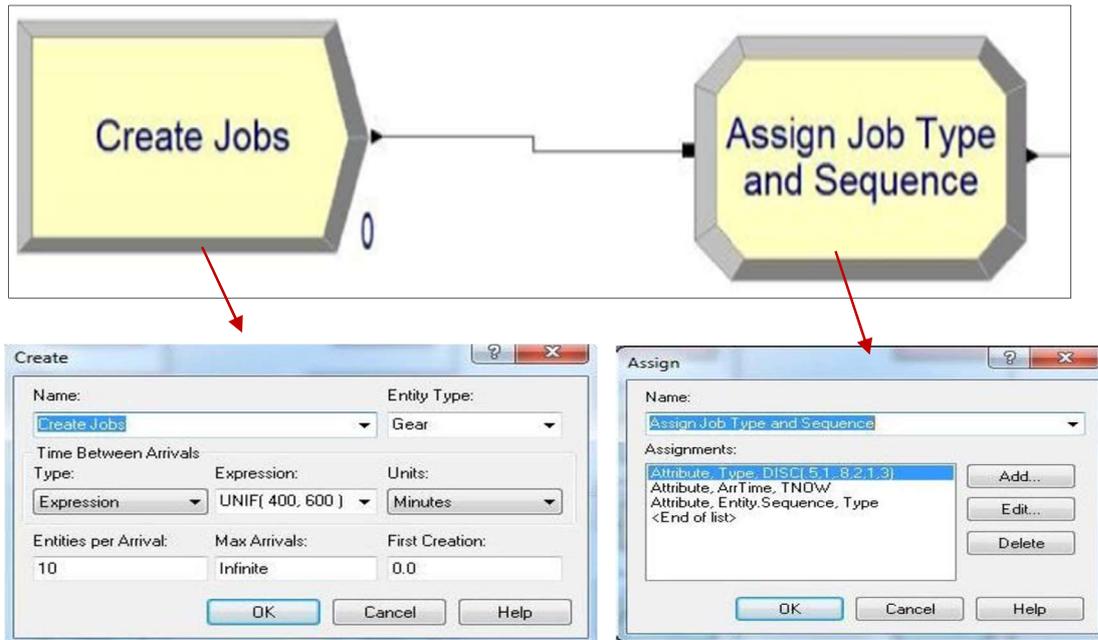


Figure 7.2 modèle des arrivées des entités arrivées

Les entités d'engrenage sont créées avec le module *create*, appelé '*create jobs*', dont la boîte de dialogue est illustrée à la figure 7.3. Le champ indique que les travaux d'engrenage arrivent par lots de 10, le champ '**time between arrivals**' spécifie que les temps d'inter-arrivée des lots doivent être uniformément répartis entre 400 et 600 min. Après l'arrivée, chaque entité d'engrenage entrante est traitée comme une entité distincte.

Une entité d'engrenage entrant entre ensuite dans le module Assign pour assigner le type et la séquence de travail. Une entité d'engrenage se voit affecter un type en l'échantillonnant à partir d'une distribution discrète et en enregistrant le code de type (1 ou 2,3) dans son attribut Type. En outre, l'attribut *ArrTime* reçoit la valeur de l'horloge de simulation, *Tnow*, qui sera utilisée ultérieurement pour calculer le temps de séjour de l'entité engrenage. Enfin, l'attribut défini par *Arena Entity.Sequence* se voit attribuer la valeur de l'attribut Type. Cet attribut agit comme un index qui associe un type d'engrenage à la séquence d'opérations correspondante.

Les séquences d'opérations pour les types d'engrenages sont spécifiées dans le module *Séquence* de la bibliothèque '*advanced transfert*'. Chaque séquence se compose d'un nom de séquence (colonne Nom) et d'une série d'étapes (colonne Étapes), énumérées dans l'ordre du Assignations). Un clic sur le bouton correspondant fait apparaître la feuille de dialogue Affectations associée. L'affectation du temps pour l'opération de fraisage est illustrée dans la figure 7.3

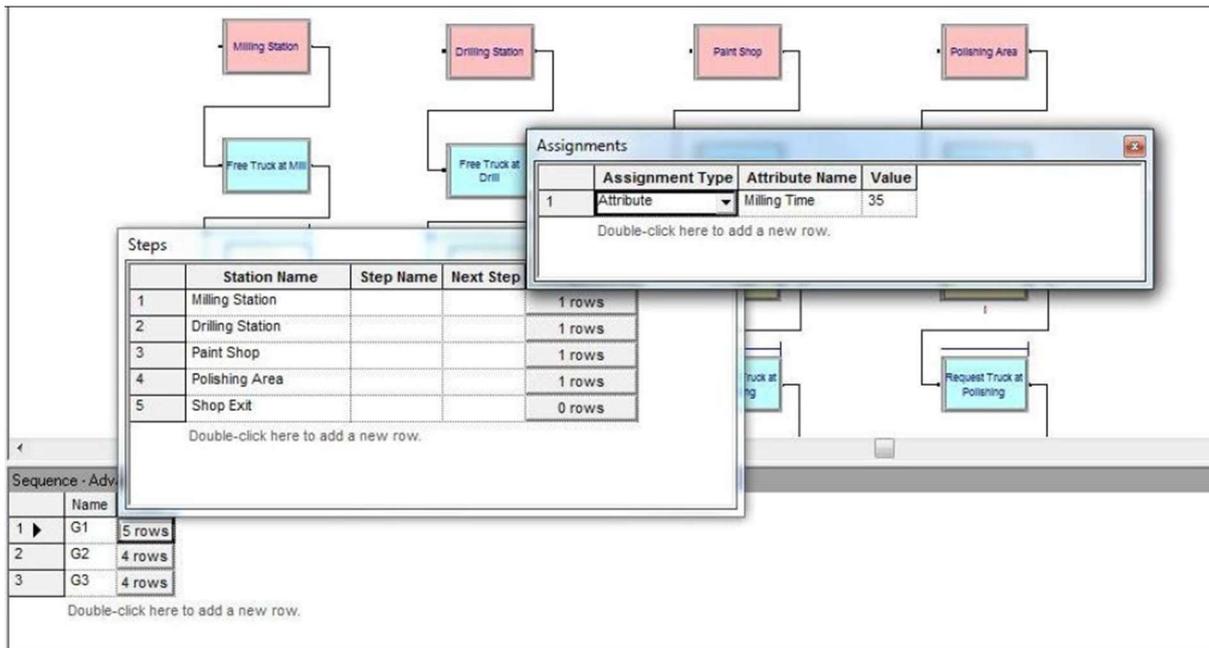


Figure 7.3 boîte de dialogue du module séquence

Les emplacements des ateliers sont modélisés comme des modules de **station**. En conséquence, chaque entité d'engrenage se dirige vers le module **Station**, appelé **Arrive_Dock**, pour modéliser son arrivée physique au quai d'arrivée de l'atelier. De là, les engins seront transportés vers l'atelier pour commencer la première étape de leur séquence d'opérations.

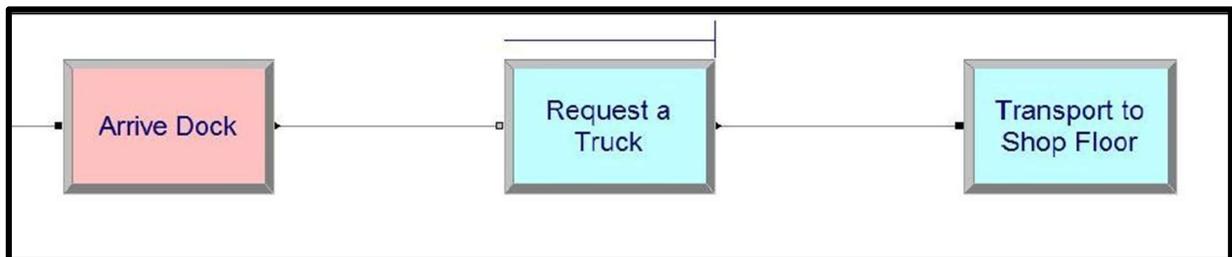


Figure 7.4 segment de transport

À cette fin, une entité engin entre dans le module **Request** (à partir de la bibliothèque Advanced transfer), appelée *request truck*, dont la boîte de dialogue est affichée dans la Figure 7.5. Le champ Nom du transporteur indique une demande de transporteur de type truck à fourche. Si plusieurs transporteurs sont disponibles, le modélisateur peut préciser comment en sélectionner un dans le champ *selection rule*. Ces sélections peuvent être cycliques, aléatoires, par ordre préférentiel (comme indiqué dans le module Transporteur), par la plus petite distance, la plus grande distance ou un transporteur spécifique. Ici, la règle de sélection demande le transporteur le plus proche du quai d'arrivée. En outre, le champ *save attribute* l'attribut spécifie que l'ID du transporteur sélectionné doit être enregistré dans l'attribut Truck_ID de l'engin demandeur.

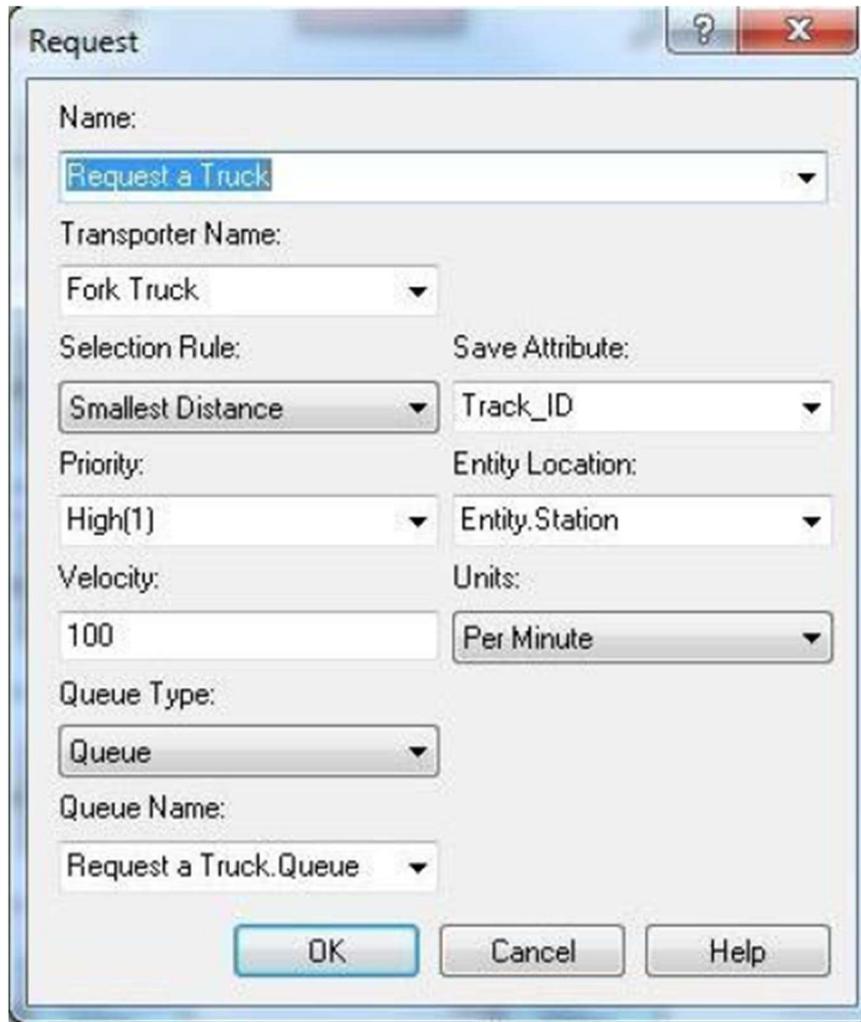


Figure 7.6 boîte de dialogue REQUEST

Les champs Name.transportor et unit number spécifient le type et l'ID du transporteur sélectionné, qui est ici le *fork truck* dont l'ID est conservé dans l'attribut `Track_ID` de l'entité d'engrenage requérante. La destination du transporteur/de l'engin est spécifiée dans le champ 'Type de destination de l'entité comme l'option Par séquence, indiquant que la destination est déterminée par le numéro de séquence de l'entité engin. Ce champ peut également spécifier un nom de module de station, en utilisant l'option Station. Il peut également spécifier un ou expression. L'entité engin et le transporteur se déplacent en tant qu'entité groupée à une vitesse 100 pieds /min, comme indiqué dans le champ *Velocity*. Notez que la vitesse peut dépendre du type de déplacement, de sorte qu'un camion vide et un camion chargé peuvent être amenés à se déplacer à des vitesses différentes.

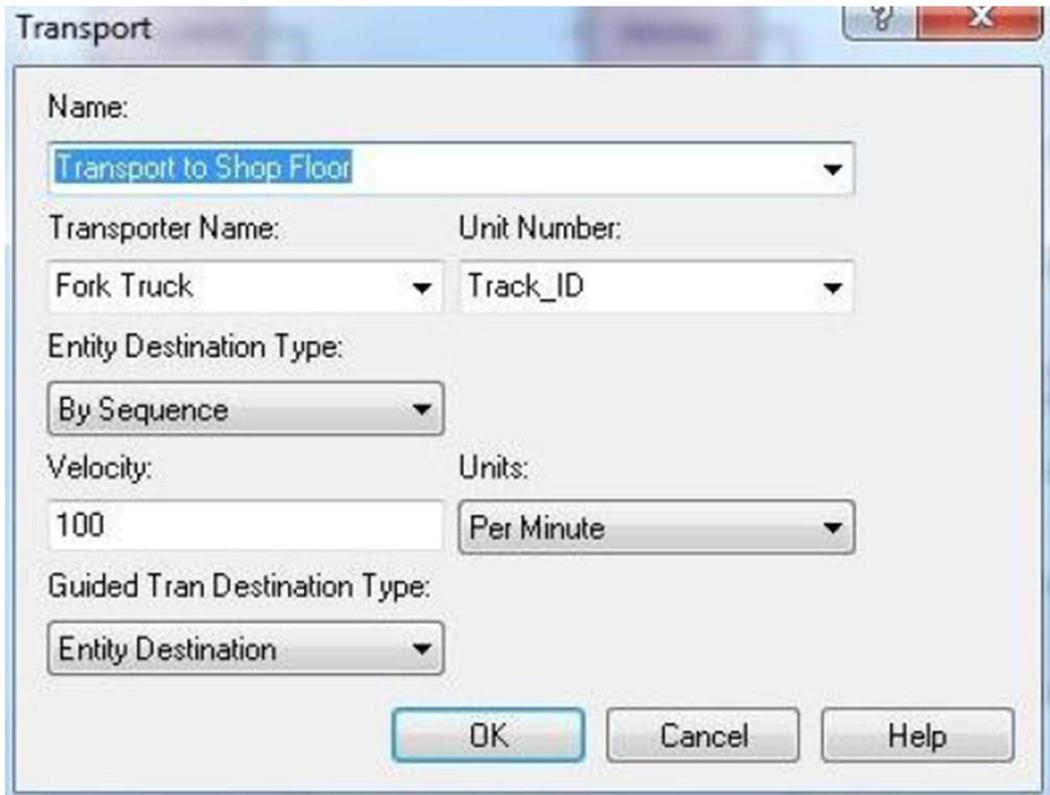


Figure 7.6 boîte de dialogue transport

Ensuite, les distances entre les différentes stations sont spécifiées. La figure montre 7.7 la feuille de dialogue du module *Distance* (à gauche), ainsi que la feuille de dialogue correspondante Stations (à droite), qui s'affiche en cliquant sur un bouton sous la colonne Stations du module *Distance*.

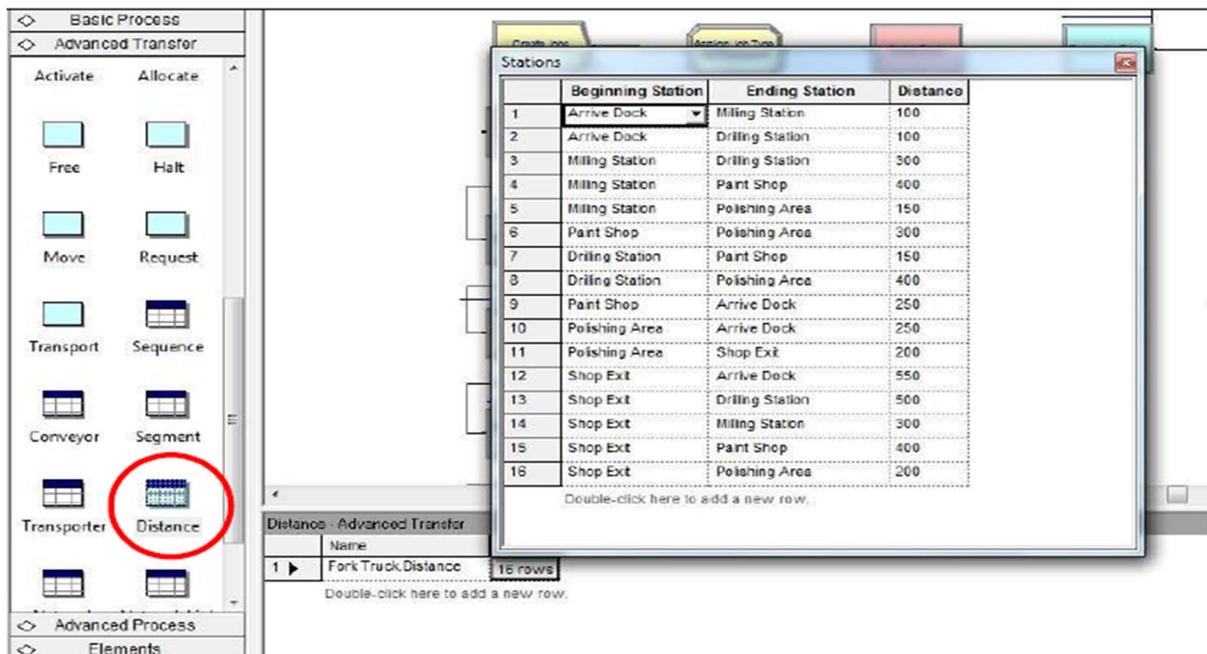


Figure 7.7 boîte de dialogue distance

Résultats

Le modèle ARENA a été simulé pendant un an.

Nous supposons que l'usine travaille pendant heures24, c'est-à-dire trois équipes de huit heures chacune. Une fois la simulation terminée, un rapport sera généré

Le rapport de sortie résultant est affiché à la figure 7.2 La section temps par entité répertorie les statistiques des temps d'attente des entités pour chaque opération. Comme prévu, le temps d'attente moyen à l'atelier de peinture est très élevé par rapport aux autres ateliers, car les temps de pulvérisation sont assez longs. Les résultats sont présentés sous forme de graphiques dans les figures 7.8, 7.9 et 7.10.

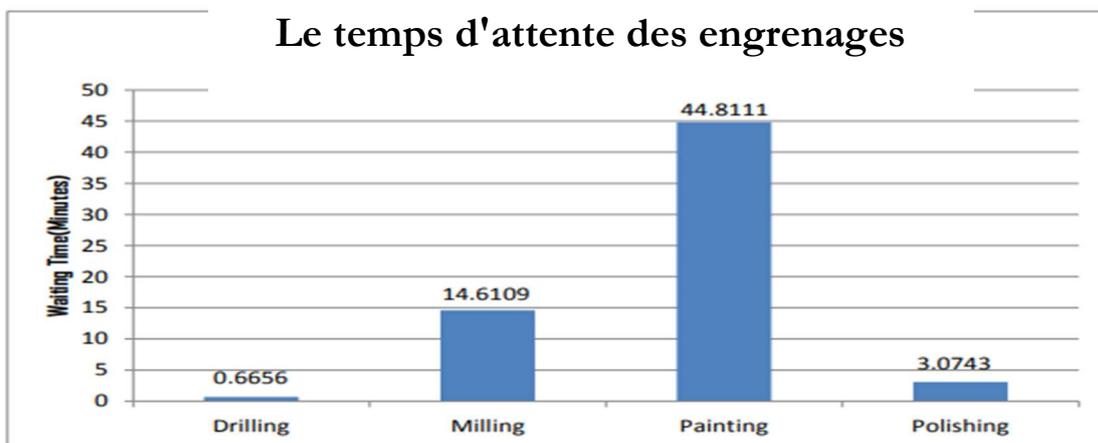


figure7. 8 Le temps d'attente des engrenages

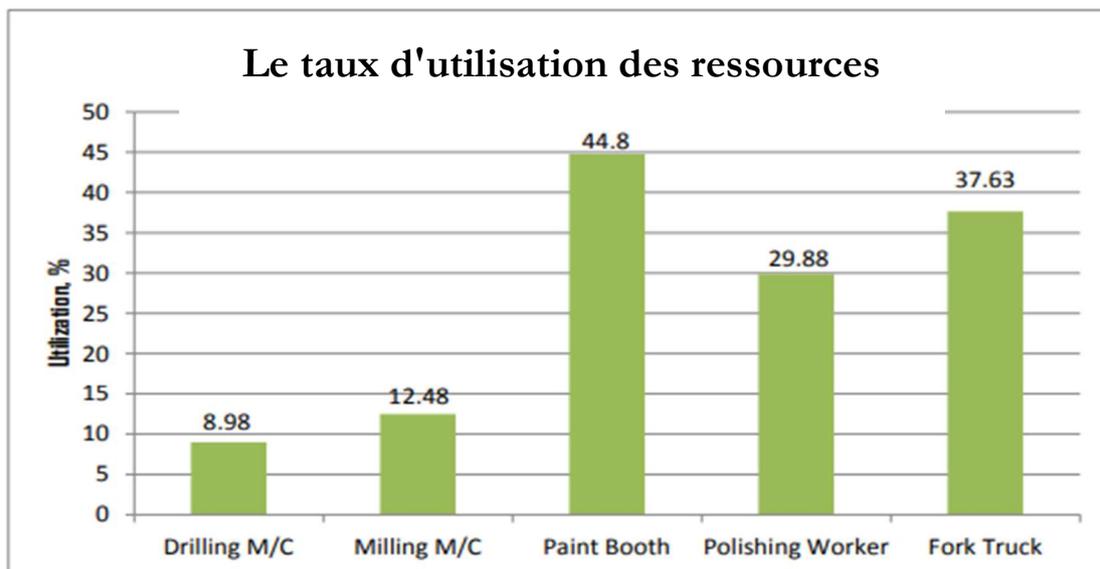


figure7. 9 le taux d'utilisation des ressources

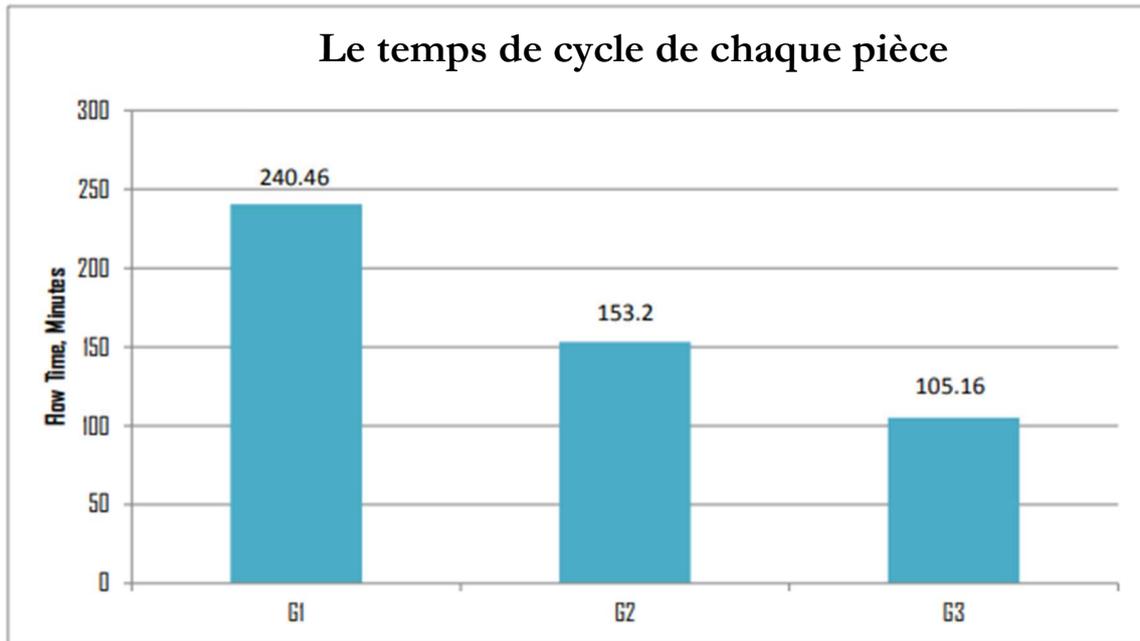


figure7. 10 le temps de cycle de chaque pièce

Replications: 1		Time Units: Minutes		
Key Performance Indicators				
System	Average			
Number Out	10,470			
Wait Time Per Entity				
	Average	Half Width	Minimum Value	Maximum Value
Drilling	0.6656	0.058478552	0.00	16.0000
Milling	14.6109	0.283086009	0.00	62.0000
Painting	44.8111	0.502319726	0.00	156.50
Polishing	3.0743	0.087881937	0.00	15.0000
Usage				
Instantaneous Utilization				
	Average	Half Width	Minimum Value	Maximum Value
Drilling Machine	0.08984399	0.001432065	0.00	1.0000
Milling Machine	0.1248	0.001909548	0.00	1.0000
Painting Section	0.4480	0.004240685	0.00	1.0000
Polishing Section	0.2988	0.002281493	0.00	1.0000
Time Persistent				
	Average	Half Width	Minimum Value	Maximum Value
Fork Truck Utilization	0.3763	0.003317102	0.00	2.0000
Usage				
None				
	Average	Half Width	Minimum Value	Maximum Value
G1 Flow Time	240.46	0.542674482	141.00	380.00
G2 Flow Time	153.20	1.50766	91.5000	326.00
G3 Flow Time	105.16	0.972138539	81.0000	224.00

figure7. 11 Rapport Arena

Conclusion

Le système Job Shop donné a été modélisé dans le logiciel Arena et les résultats ont été générés. Après analyse des résultats, il a été remarqué que le temps d'attente dans l'atelier de peinture était plus élevé que dans les autres stations. Il a donc été suggéré d'augmenter le nombre de cabines de peinture d'une unité.

Et il a été constaté que le temps d'attente dans l'atelier de peinture augmente mais que l'utilisation de cette station diminue. L'acceptation ou le rejet de cette suggestion dépend des politiques de l'organisation. Si elle accepte de sacrifier l'utilisation d'une machine au profit du temps d'attente, elle peut aller de l'avant avec cette suggestion. Elle doit prendre en compte le coût de fonctionnement de la machine et la perte due à un temps d'attente important dans une station. C'est là que se trouve le compromis.

Encore une fois, de nombreuses suggestions peuvent être faites dans ce système, en supprimant certaines machines dans une station qui a moins d'utilisation de sorte que nous pouvons augmenter l'utilisation et nous pouvons économiser de l'argent dans le fonctionnement de la machine. Nous pouvons également envisager des changements dans l'hypothèse 1" et "2".

Modélisation

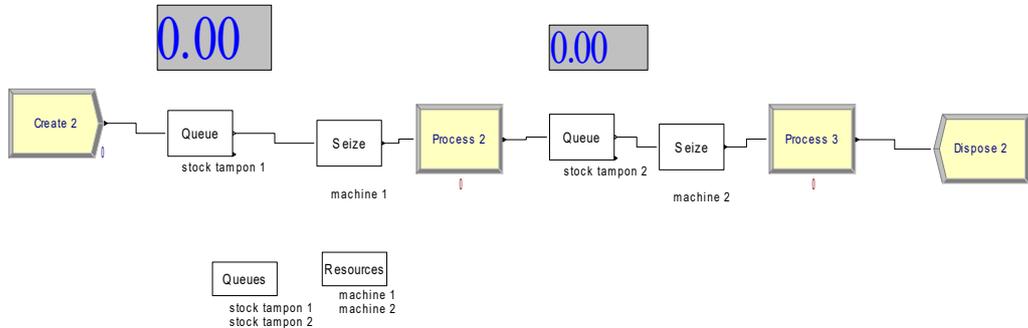


Figure 8.1 modélisation de TP8

Nous remarquons bien que la capacité de la file d'attente = 10 pour cela nous avons utilisé le module « *queue* » de la bibliothèque « *blocks* » et « *queues* » de la bibliothèque « *elemnets* ». Ce dernier est utilisé pour la déclaration.

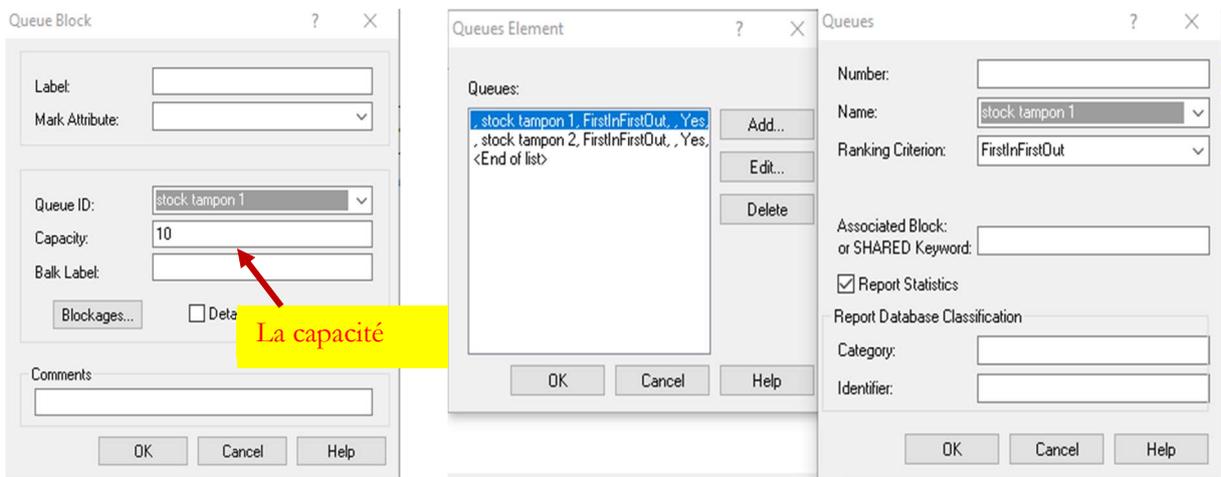


Figure 8.2 boîte de dialogue queue et queues

Dans les modules « *seize* » de la bibliothèque « *Queue* » nous avons réservé les ressources *machine1* et *machine 2* qui sont déclarées dans le module « *ressources* » de la bibliothèque « *Elements* » voir figure 8.3

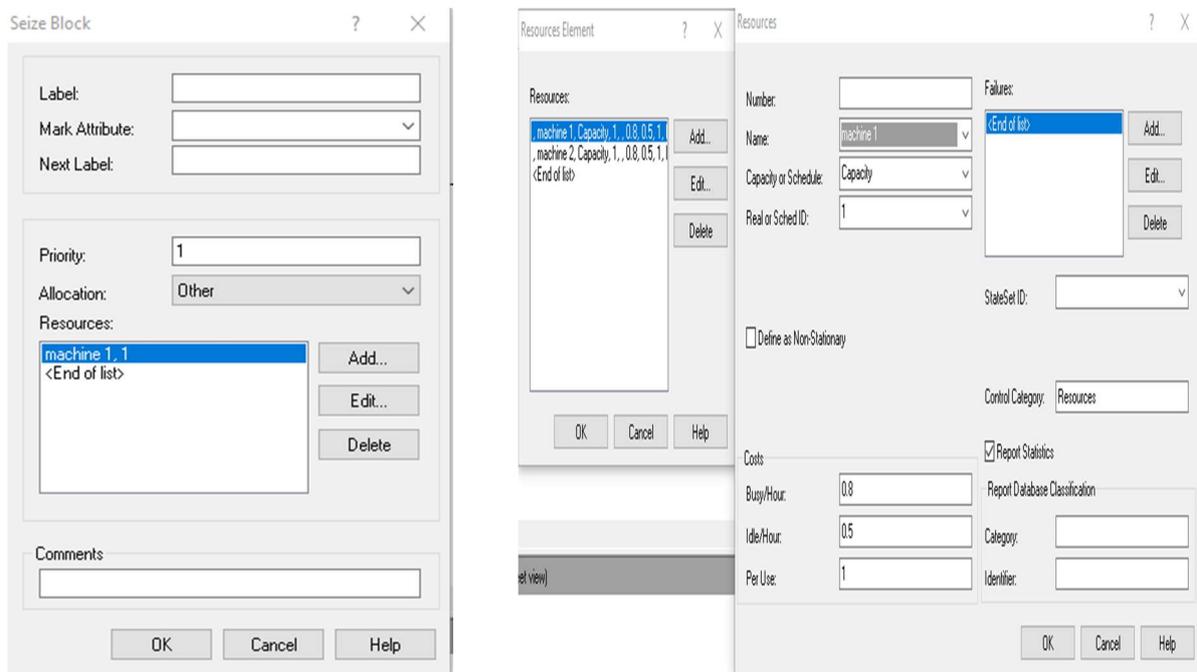


Figure 8.3 boîte de dialogue seize block et Ressources Elements

Pour les couts voir l'énoncé de TP

Les résultats :en min

La taille moyenne de chacun des stocks tampons =

Number Waiting				
	Average	Half Width	Minimum Value	Maximum Value
stock tampon 1	9.8559	(Correlated)	0.00	10.0000
stock tampon 2	9.4465	(Correlated)	0.00	10.0000

- Le temps moyen d'attente dans chacun des stocks tampon ;

Waiting Time				
	Average	Half Width	Minimum Value	Maximum Value
stock tampon 1	146.63	(Correlated)	0.00	150.00
stock tampon 2	279.23	(Correlated)	0.00	300.00

- Le taux d'utilisation de chacun des postes de travail ;

Instantaneous Utilization				
	Average	Half Width	Minimum Value	Maximum Value
machine 1	1.0000	(Insufficient)	0.00	1.0000
machine 2	0.9985	(Insufficient)	0.00	1.0000

- Le coût de valeur ajouté ;

Value Added Cost	599
• Le coût valeur non ajouté Non-Value Added Cost	0
• Le coût d'attente Wait Cost	3,180
• Le coût total Total Cost	1,266

Key Performance Indicators

All Entities

	Average
Non-Value Added Cost	0
Other Cost	1,000
Transfer Cost	0
Value Added Cost	599
Wait Cost	3,180
Total Cost	4,778



All Resources

	Average
Busy Cost	266 *
Idle Cost	0
Usage Cost	1,000 *
Total Cost	1,266



* these costs are included in Entity Costs above.

System

	Average
Total Cost	4,778
Number Out	332

Figure 8.4 Rapport sur les différents coûts

2. Optimisation

Pour cela nous utilisons l'outil *Optquest*

- **Définition** : OptQuest surmonte cette limitation en recherchant automatiquement les solutions optimales dans les modèles de simulation Arena. L'idée est de définir le problème d'optimisation dans OptQuest puis de le programmer pour rechercher les valeurs indiquées des contrôles qui maximisent (ou minimisent) un objectif donné.

• L'interface d'OptQuest

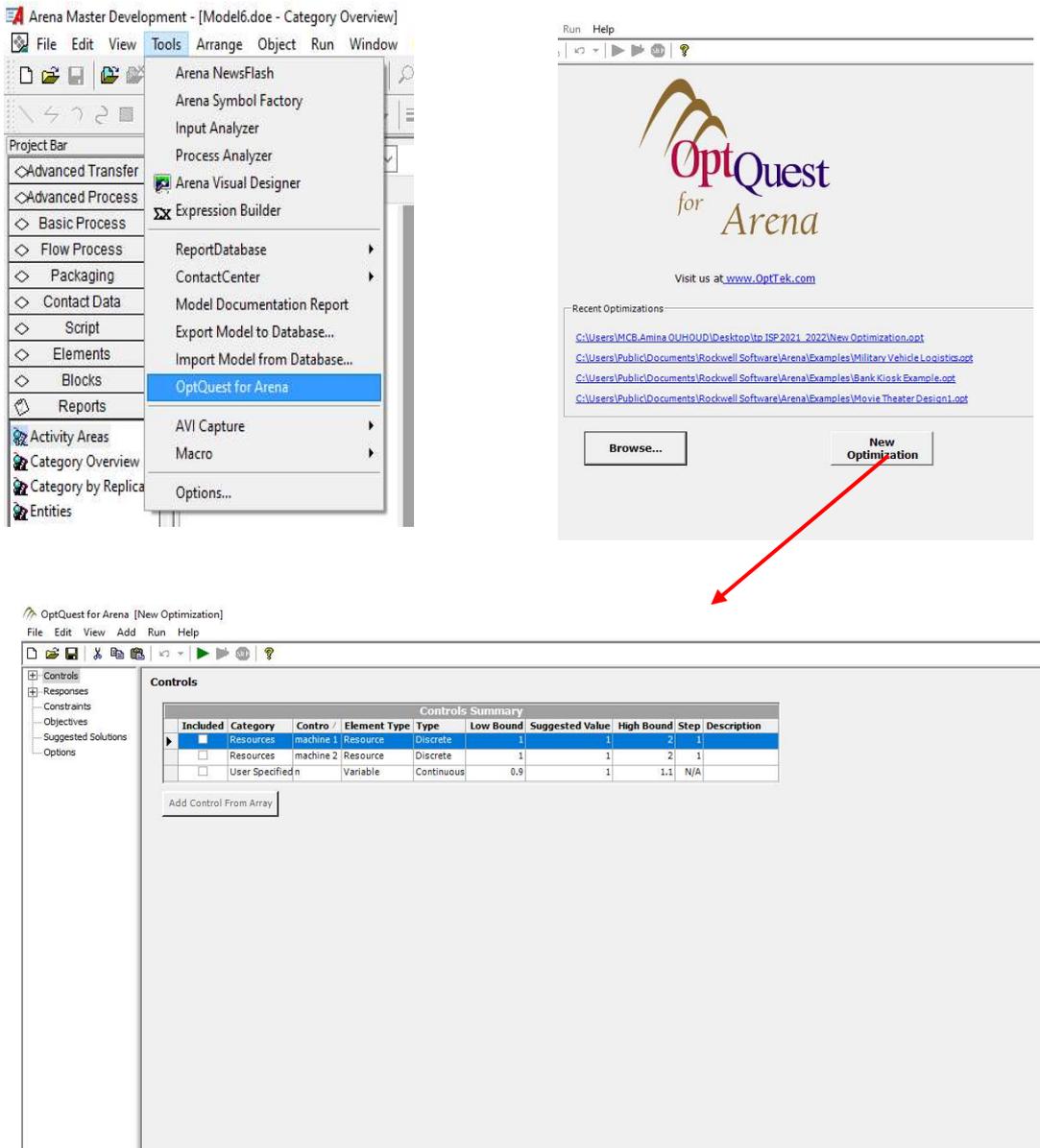


Figure 8 .5 Interface OPTQUEST

Une fois le problème d'optimisation conçu (les contrôles ou variables de décision, l'objectif et les contraintes ont été sélectionnés), le modèle dans Arena est testé avec différentes valeurs de contrôle choisies pour être évaluées. La méthode d'optimisation OptQuest évalue les résultats d'une exécution particulière, les analyse et les compare aux exécutions précédentes, et détermine un nouvel ensemble de valeurs pour les contrôles, qui seront ensuite évalués en exécutant à nouveau le modèle Arena. Ce processus itératif conduira éventuellement à une amélioration des résultats avec chaque nouvel ensemble de valeurs de contrôle. Le processus se poursuit jusqu'à ce qu'un certain critère de terminaison soit satisfait (arrêt après un nombre fixe de simulations, ou lorsque OptQuest détermine que la fonction objectif a cessé de "s'améliorer"). Une fois l'application

OptQuest fermée, les commandes du modèle Arena reprennent leurs valeurs d'origine. Le modèle n'est pas affecté par l'optimiseur.

Dans notre modèle les :

- **controls** sont les ressources c-a-d la capacité avec lower band == 1 et upperbound== 5 pour les deux ressources

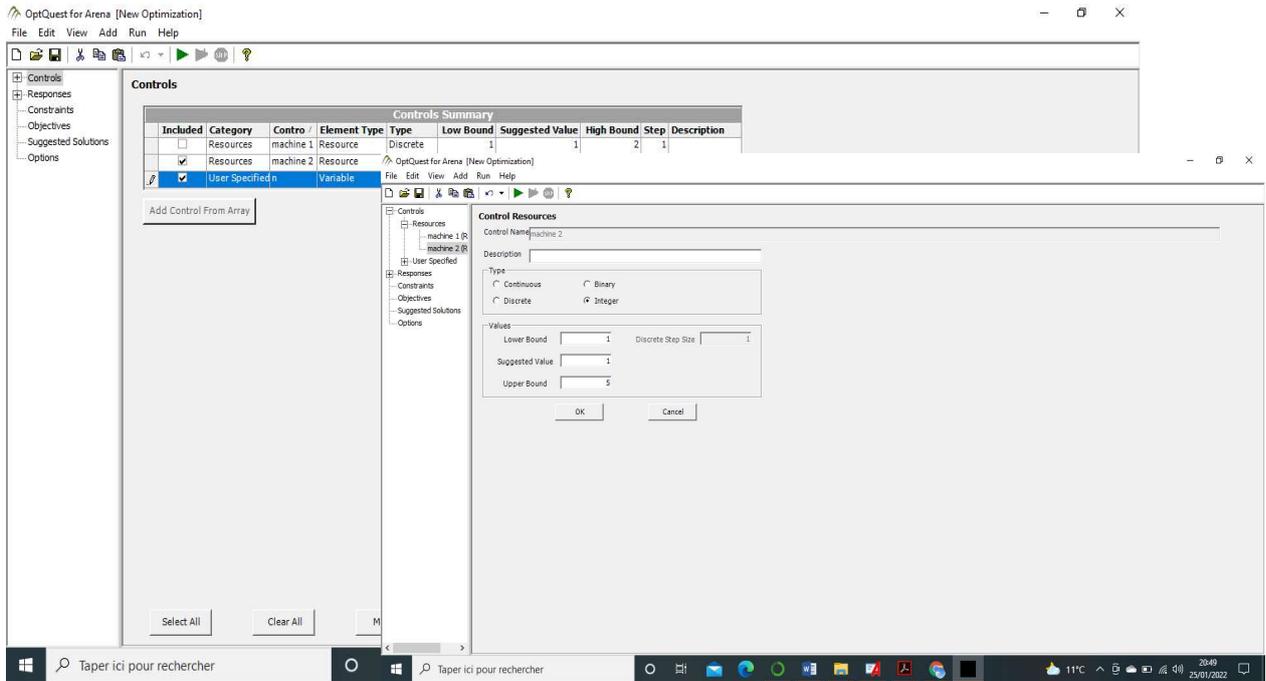


Figure 8.6 controls de l'optimisation

- **responses** est le total cost

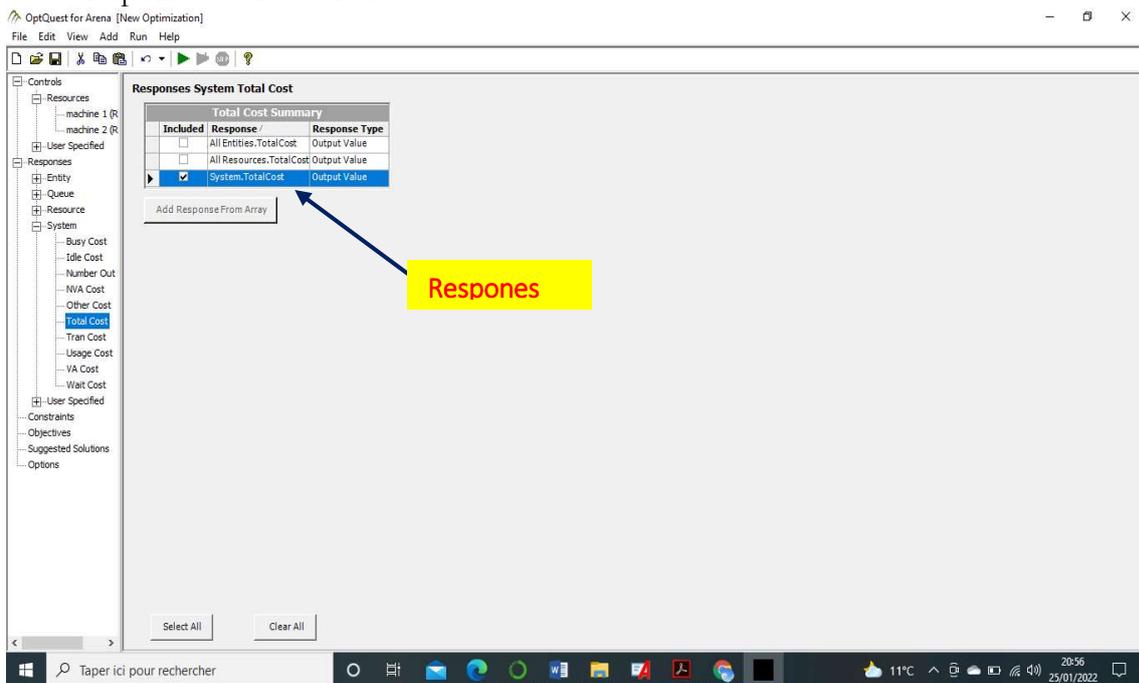


Figure 8.7 fenêtre de réponses

➤ La fonction objectif

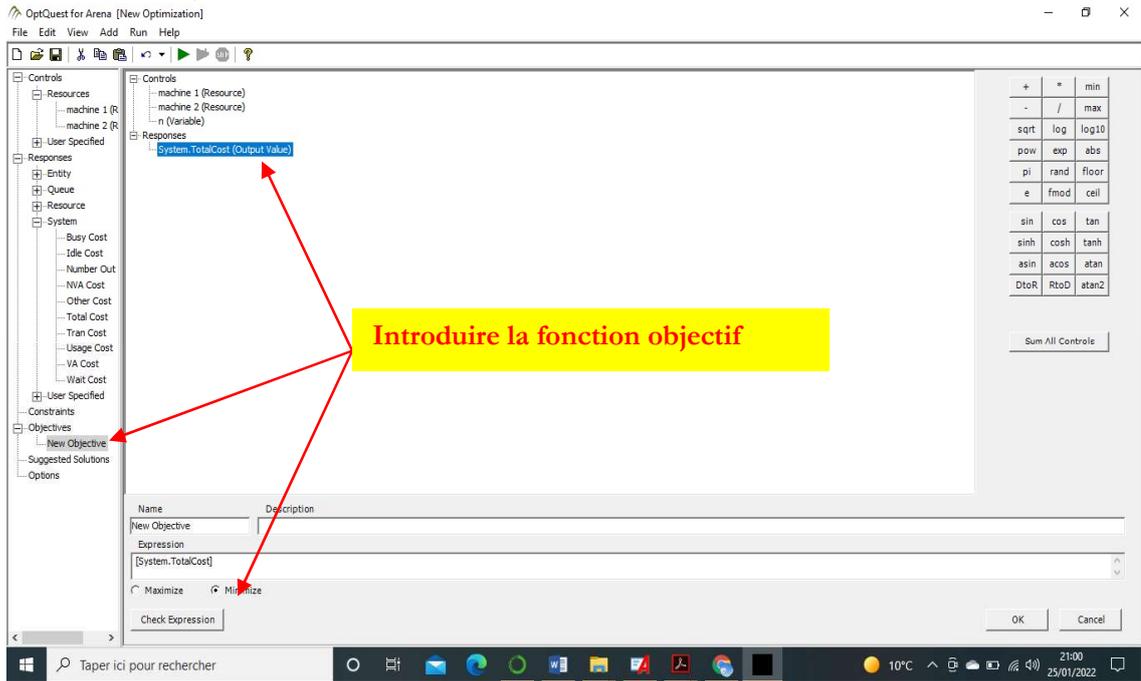


Figure 8.8 fenêtre de la fonction objectif

➤ Résultat d'optimisation

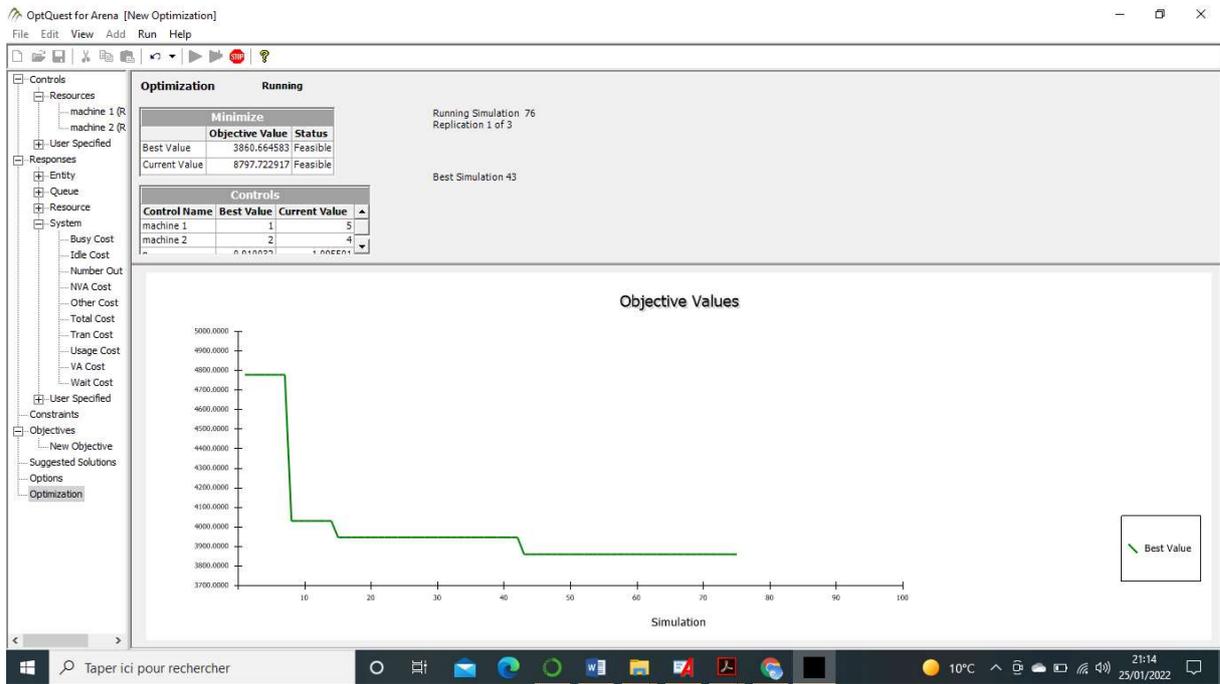


Figure 8.9 Résultat d'optimisation

Les résultats montrent que la machine 1 avec capacité = 5 et la machine 2 avec capacité = 4 nous donne le min total cost = 3860.66

CONCLUSION

La simulation est un outil important pour la modélisation de systèmes complexes. Elle est utilisée pour représenter des systèmes de fabrication, de transport et de services dans un programme informatique afin de réaliser des expériences. La représentation du système par le biais d'un programme informatique permet de tester les changements de conception technique sans perturber le système modélisé. La modélisation par simulation fait appel à des éléments de la modélisation des systèmes, la programmation informatique, les probabilités et les statistiques, ainsi que la conception technique. Étant donné que la modélisation de simulation fait appel à ces sujets individuellement difficiles, l'enseignement et l'apprentissage de la modélisation de simulation peuvent être difficiles pour les étudiants. L'enseignement et l'apprentissage de la modélisation par simulation peuvent être difficiles, tant pour les instructeurs que pour les étudiants. Les instructeurs doivent présenter les concepts de programmation informatique, la modélisation des probabilités et l'analyse statistique dans le contexte de l'enseignement de la modélisation de systèmes complexes tels que les usines et les chaînes d'approvisionnement. En outre, en raison de la complexité associée à la modélisation de la simulation, des langages informatiques spécialisés sont nécessaires et doivent donc être enseignés aux étudiants pour qu'ils puissent les utiliser pendant le processus de construction du modèle. Ce document a pour but d'aider les étudiants dans cette tâche ardue.

Ce polycopié est conçu comme un manuel des travaux pratiques de simulation des systèmes de production destiné aux étudiants de deuxième cycle spécialité génie industriel à un niveau d'introduction. Il vise les aspects pratiques de la simulation, ainsi que le fonctionnement de l'environnement de simulation Arena. Cependant, le polycopié n'est pas structuré comme un manuel d'utilisation pour Arena, et nous recommandons fortement aux lecteurs de consulter les fonctions d'aide d'Arena pour plus de détails sur les constructions des modèles avec Arena.

Références Bibliographiques

Références Bibliographiques :

[LAW, KEL, 00] Law, A.M. and Kelton, D.W. (2000), “Simulation Modeling and Analysis”, 3^{ème} ed., Edition McGraw-Hill, New York 2000.

[LAW, KEL, 91] Law, A. M. & Kelton, D. W. (1991) , “Simulation Modeling and Analysis, Industrial Engineering and Management Science”, 2^{ème} ed, Edition McGraw-Hill Inc.1991.

[YUR, JEL, 05] Yuri Merkurjev. Jelena Pecherska, support de cours « Discrete-Event Simulation: Methodology and Practice » Department of Modelling and Simulation Riga Technical University Riga, Latvia, 2005, p41.