



MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ ABDELHAMID IBN BADIS - MOSTAGANEM

Faculté des Sciences Exactes et de l'Informatique
Département de Mathématiques et d'Informatique
Filière : Informatique

MEMOIRE DE FIN D'ETUDES
Pour l'Obtention du Diplôme de Master en Informatique
Option : **Systèmes d'Information Géographique**

THEME :
Simulation d'un système de production avec ARENA.

Etudiant(e)s : **BACHIRI Imene.**

LARBI BENHADJAR Abdelkrim.

Encadrant(e) : **BELKACEM Imad.**

Année Universitaire 2015/2016.

Résumé

Résumé :

Arena simulation software est le logiciel de simulation à événements discrets le plus utilisé dans le monde. Avec Arena, nous pouvons intégrer la programmation, lire et écrire à partir des bases des données Microsoft et réaliser des animations 2D.

Alors, il sera intéressant de simuler un système de production sur mesure avec Arena, l'expérience acquise à travers cette recherche aide à construire une base solide dans la simulation à événements discrets qui s'applique à des situations du monde réel rencontrées dans le lieu de travail.

Les mots-clés : modélisation, simulation, système de production, les événements discrets.

Sommaire :

Résumé

Dédicaces

Remerciements

Sommaire i

Liste des tableaux.....v

Liste des figuresvi

Liste des abréviations et acronyme.....vii

Introduction générale.....1

I. Modélisation et simulation 2

I.1. Introduction 2

I.2. La modélisation 2

I.2.1. Définition 2

I.2.2. Les étapes de modélisation d'un système..... 2

I.2.3. A quoi sert la modélisation? 4

I.2.4. Les variables aléatoires 4

I.2.4.1. Définition 4

I.2.4.2. La densité de la variable aléatoire 4

I.2.4.3. La loi de distribution d'une variable aléatoire(x) 4

I.2.4.4. L'espérance mathématique d'une variable aléatoire (x).. 4

I.2.4.5. La variance.. 4

I.2.4.6. L'écart type 4

I.2.4.7. Le coefficient de la corrélation 4

I.2.5. Les lois discrètes 5

I.2.5.1. La loi de Bernoulli.....5

I.2.5.2.La loi de Binomiale.....5

I.2.5.3.La loi géométrique.....6

I.2.5.4. La Loi de poisson $p(\lambda)$6

I.2.6. Les lois continues classiques 7

I.2.6.1. La loi uniforme.....7

I.2.6.2.La loi triangulaire.....7

I.2.6.3. La loi exponentielle.....	8
I.2.6.4. Loi normalisé(ou de la place –gauss) : $N(m, \sigma)$	9
I.2.7. Les méthodes de la modélisation	9
I.2.7.1. Les méthodes descriptives	9
I.2.7.2. Les méthodes analytiques	9
I.2.8. Les réseaux de pétri	10
I.2.8.1. Définitions d'un RDP.	10
I.2.9. Les systèmes d'attente	11
I.2.9.1. Modèle de files d'attente.....	11
I.3. La simulation.....	13
I.3.1. Notion de la simulation	13
I.3.2. Définition de la simulation	13
I.3.3. L'utilité de la simulation.....	13
I.3.4. Simulations informatiques.....	14
I.3.5. Les domaines d'applications	14
I.3.6. Les limites de la simulation	15
I.3.7. Méthodologie générale	15
I.3.8. Création d'un modèle	16
I.3.9. Les différents types de modèle	17
I.3.10. Classification des modèles de simulation	17
I.3.11. La méthode de monte-carlo.....	18
I.3.12. Les avantages et les inconvénients de la simulation	19
I.4. Conclusion	19
II. Les systèmes de production	20
II.1. Introduction.....	20
II.2. C'est quoi un système de production	20
II.2.1. La simulation d'un système de production	20
II.3. Les systèmes à événements discrets(SED)	20
II.3.1. Les systèmes de production et ateliers flexibles	20
II.3.2. Les systèmes informatique	21
II.3.3. Réseaux de communication et de transport	21

II.3.4.planification de taches	21
II.4.Les Différents systèmes de production	21
II.4.1.La production à l'unité ou en série	21
II.4.2.La production continue ou discontinue	21
II.5. La production de service	22
II.5.1.Les spécifités de la production de service	22
II.5.2.Le front office et le back office.	22
II.6. Le choix d'un système de production	22
II.6.1.Les critères de choix	22
II.7. L'évolution des systèmes de production	22
II.8. Classification des éléments	23
II.9. Le choix d'un système de production	23
II.9.1.La performance structurelles	23
II.9.2.performance économique	24
II.10.Les travaux sur la simulation des systèmes de production(histoire)	24
II.11. L'approche orientée objets	25
II.12. Les différents logiciels de simulation	25
II.13.Conclusion	27
III.Simulation d'assemblage des pieces électronique	28
III.1. Introduction	28
III.2. Pour quoi on a choisi Arena simulation ?	28
III.3. Utilisation d'Arena	28
III.3.1. Les modules à configuré sous ARENA	29
III.4. La modélisation et la configuration de notre projet	32
III.4.1. Présentation	32
III.4.2. Implémentation	34
III.4.3. Le modèle gnérale ARENA	40
III.5. Animation 2D	41
III.5.1. Les agents	41
III.5.2. Les machines de test	42
III.5.3. Les agents verts.....	42
III.5.4. Schéma de l'atelier d'assemblage électroniques	43
III.6. Les résultats de simulation	44

III.7. Rapports de simulation	45
III.8. Application de deuxième scénario	45
III.8.1. Les résultats de simulation	46
III.9. Interface visual basic	47
III.10.Conclusion	47
 Conclusion générale.....	 48
Bibliographie.....	49

Tableau 1: Les différents types de modèles.....17

Figure I. 1 : processus de modélisation.....	3
Figure I. 2 : Un modèle sur le réseaux de pétri (RDP)	10
Figure I. 3 : Méthodologie d'une simulation	16
Figure III. 1 : Image intarface d'ARENA	29
Figure III. 2 : Basic process d'ARENA	29
Figure III. 3 : Les paramètres de create	29
Figure III. 4 : Les paramètres de dispose	30
Figure III. 5 : Les paramètres de process	30
Figure III. 6: Les paramètres de Decide	31
Figure III. 7: Les paramètres de Bach	31
Figure III. 8 : Les paramètres de separate	31
Figure III. 9: Les paramètres de Assign & Record.....	32
Figure III. 10 : Assemblage de ABCD	33
Figure III. 11 : Assemblage de ABCD2	33
Figure III. 12 : configuration de create & process(les arrives A,B,C,D)	34
Figure III. 13 : Assemblage de A avec B et C avec D	34
Figure III. 14 : configuration de Batch & configuration de process	35
Figure III. 15 : Machine de test AB et CD & configuration de Decide	35
Figure III. 16 : configuration de process	35
Figure III. 17: Assemblage de ABCD&configuration de Batch	36
Figure III. 18: Configuration de process	36
Figure III. 19 : Configuration de Decide & Process.....	37
Figure III. 20: Regroupement des plaques & configuration de Batch	37
Figure III. 21 : Séparer la pièce A de la plaque ABCD& test de la pièce A	37
Figure III. 22 : Configuration de process & configuration de separte	38
Figure III. 23: Dessouder B	38
Figure III. 24: Dessouder Cet D	38
Figure III. 25 : Test si les pièces est valide	38
Figure III. 26 : Assoumblage & test	39
Figure III. 27 : Assoumblage & test & regroupement	39
Figure III. 28 : Le model général	40
Figure III. 29 : Agent rouge	41
Figure III. 30 : Agent bleu	41
Figure III. 31 : Machine de test	42
Figure III. 32 : Agent vert	42
Figure III. 33 : schéma général de l'atelier	43
Figure III. 34 : configuration de Record.....	44
Figure III. 35 : configuration de Record(count)	44
Figure III. 36 : interface avec Visual basic.....	47

Liste des abréviations et acronymes.

RDP : réseaux de pétri.

SED : Les systèmes à événements discrets.

GPSS: General Purpose Simulation System.

CSL: Continuous Simulation Langage.

DEVS: Discrete Event System Specification.

VIMS: Visual Interactive Modelling Systems.

GPAO : gestion de la production assistée par ordinateur.

SDP : Système De Production

PAPS : Premier qui Arrivée, Premier qui Sort

Introduction générale

De nos jours le monde connaît des développements et des avancements terribles dans les domaines de la simulation.

La modélisation est la première étape de la simulation. Cette étape passe par l'analyse de comportement et définit les problèmes d'un système ou d'un phénomène quelconque, le modèle numérique n'est qu'une approximation d'un phénomène réel ou hypothétique rédigé pour la simulation.

La simulation est la traduction réelle du phénomène du processus en présence d'un ensemble d'équations - mathématiques - dans les programmes informatiques développés pour l'étude et l'évaluation.

La simulation par ordinateur c'est la création d'un modèle numérique à l'aide des logiciels pour résoudre des problèmes d'une situation existante ou hypothétique ou un problème inattendu, ce qui facilite le processus d'étudier le comportement du système et ça par la modification sur les hypothèses ou l'influence de comportement des variables du système, et ce type de simulation a connu un grand développement au cours des dernières décennies et est actuellement en vigueur dans les domaines de la recherche pour améliorer la performance d'une entreprise ou d'une usine.

Le succès d'une entreprise ou d'une usine est lié à la bonne gestion de son système de production, ce dernier est un ensemble des composants matériels (les machines de production, les camions...) et immatériels (la main-d'œuvre).

Le système de production est un système à événement discret et pour le simuler nous utilisons un logiciel adapté avec ce type de système.

Il existe plusieurs logiciels de simulation chacun a des configurations plus ou moins que d'autres logiciels et le plus performant à notre connaissance et adapté avec les systèmes à événements discrets est le simulateur **ARENA** dans le système d'exploitation **Windows** qui va nous aider à consister notre but de simuler, évaluer et améliorer la performance d'un système de production.

I. Modélisation et simulation :

I.1. Introduction :

La simulation sur ordinateur est un outil très utile pour bien comprendre le principe de fonctionnement d'un système. Elle aide à observer et à comprendre le fonctionnement en temps réel et à analyser plusieurs scénarios du fonctionnement.

Dans ce chapitre nous présentons la modélisation et la simulation d'un système de production et leurs avantages et leurs conventions et aussi les modèles et leur différent type et définir quelque notion de réseau pétri.

I.2. La modélisation :

I.2.1. Définition :

La modélisation consiste à définir les points suivants : Le système, Le modèle, L'objectif, Un critère de rentabilité [1].

- ✓ **Le système** : il y a de nombreuses définitions ont été attribuées au system
 - Ensemble de composants reliés entre eux.
 - Ensemble organisé d'éléments fonctionnels.
 - Assemblage d'éléments fonctionnant de manière unitaire et en interaction Permanente [1].
- ✓ **Le modèle** : structure simplifiée de la réalité qui ne contient pas que les caractéristiques estimées importantes pour l'étude du système [1].
- ✓ **L'objectif** : le but pour lequel de modèle a été élaboré [1].
- ✓ **Un critère de rentabilité** : un critère économique qui justifie l'utilisation d'un modèle [1].

I.2.2. Les étapes de modélisation d'un système :

L'étape de modélisation est une phase essentielle à la simulation. Différents points doivent être abordés:

- Définir l'objectif de la modélisation
- Définir les éléments du système (via la réalisation d'une fonction, ou d'un processus) et les limites du système (les entrées, les sorties).
- Définir les interactions entre ces éléments (hiérarchie).
- Définir la dynamique du système (entités qui circulent entre les éléments, comportement du système au cours du temps).
- Abstraction (choisir les éléments du système pertinents pour l'étude).
- Formalisation, conceptualisation: Modèle mathématique (les lois discret et les lois continues), modèle logiciel (Simulink, Siman Arena), modèle graphique (réseaux de Pétri) [2].

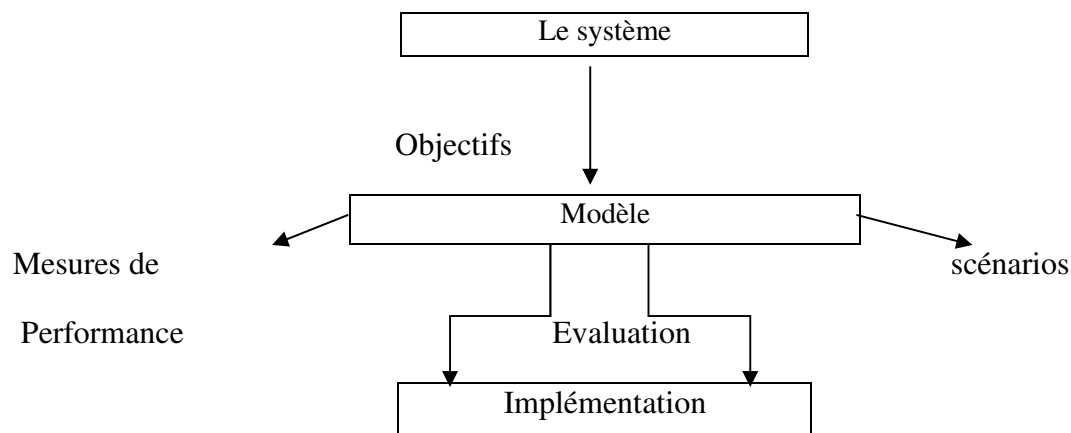


Figure I.1 : processus de modélisation.

➤ **Exemple sur la modélisation d'un système :**

- a. **Système :** on a un système d'assemblage électronique, il fabrique 100 pièces dans une période de 6 heures dans un jour avec l'arrivée de 800 composants (chaque pièce contient de huit composants)
- ✓ **Objectifs :** élever le nombre des pièces arrivées vers 1200 pièces et fabriquer de 150 Dans une période de 8 heures.
- b. **Les modèles :** Créer un modèle numérique avec un logiciel le même que le modèle réel.
- ✓ **Les scénarios :**

1^{er} scénario : on augmente la durée de travail des salariés.

2^{ème} scénario : on minimise la durée de fabrication de chaque pièce.

3^{ème} scénario : acheter des nouvelles machines.

✓ **Les mesures de performances :**

- Si le responsable sur l'usine souhaite le changement sur le système ne soit pas coûteux alors le 3^{ème} scénario n'est pas valable.
- Si les machines peuvent s'adapter avec l'augmentation de la vitesse de production alors le 2^{ème} scénario est valide si non, alors il reste d'appliquer le 1^{er} scénario.

c. **Implémentation :** On applique le meilleur scénario sur le système.

I.2.3. A quoi sert la modélisation ? :

- **Compréhension de la réalité:** le modèle pourrait apporter une connaissance additionnelle sur les mécanismes de fonctionnement d'un phénomène (comportements oscillatoires, états d'équilibre, stabilité, instabilité) [3].
- **Génération de solutions:** les solutions du modèle fournissent des exemples de comportement du système. Parfois une solution peut être obtenue de manière analytique mais dans les cas les plus généraux il faut avoir recours à la simulation numérique [3].
- **Étude de dépendances structurelles:** il est intéressant étudier et analyser le comportement de solutions en fonction des paramètres ou de la structure du modèle (analyse de la stabilité, sensibilité et robustesse) [3].
- **Contrôle:** conception de règles ou d'un système de contrôle capable d'amener le système vers un état désiré.

I.2.4. Les variables aléatoires :

I.2.4.1. Définition :

Une variable aléatoire X est une fonction de l'ensemble fondamental Ω à valeurs dans $\mathbb{R}$, $X : \Omega \rightarrow \mathbb{R}$. Lorsque la variable X ne prend que des valeurs discrètes, on parle de variable aléatoire discrète [4].

I.2.4.2. La densité de la variable aléatoire :

est défini par : $f(x) = p[(X) = x]$.

I.2.4.3. La loi de distribution d'une variable aléatoire(x) :

$F(x) = p[(X) \leq x] \rightarrow$ fréquence cumulée.

I.2.4.4. L'espérance mathématique d'une variable aléatoire (x) :

Est noté :

$$E(x) = \sum_{i=1}^n p_i x_i \rightarrow \text{cas discret}$$

Dans le cas continu :

$$E(x) = \int_{-\infty}^{+\infty} x f(x) dx$$

↖
Densité

I.2.4.5. La variance :

$$V(x) = \sigma^2(x) = E(x^2) - [E(x)]^2$$

I.2.4.6. L'écart type :

La racine de la variance.

$$T(x) = \sqrt{v(x)} = \sqrt{\sigma^2(x)} = \sqrt{E(x^2) - [E(x)]^2}.$$

I.2.4.7. Le coefficient de la corrélation :

$$e(x) = v(x) / E(x).$$

I.2.5. Les lois discrètes :

Par définition, les variables aléatoires discrètes prennent des valeurs entières discontinues sur un intervalle donné. Ce sont généralement le résultat de dénombrement [5].

I.2.5.1. Loi de Bernoulli :

On appelle variable de Bernoulli, la variable aléatoire X telle que :

$$X : \Omega \rightarrow \mathbb{R}$$

Cette loi utilisée lorsqu'on a une expérience où on a deux cas :

$X(\Omega) = \{0,1\}$ car $\Omega = \{E, S\}$ et S pour succès et E pour échec.

S est réalisé, $X = 1$ sa probabilité p

Si E est réalisé, $X = 0$ sa probabilité $q = 1 - p$

$E(x) = p$ | l'espérance de x est la probabilité de succès

$V(x) = p \cdot q$ | la variance de x est la probabilité de succès fois la probabilité de l'échec.

La loi de probabilité associée à la variable de Bernoulli est appelée loi de Bernoulli notée **B (1, p)** [5].

✓ **Exemple d'utilisation :** un jet d'une pièce de monnaie (non truquée)

Probabilité de pile = 0.5

Probabilité de face = $1 - 0.5 = 0.5$

Si la pièce de monnaie est truquée alors la probabilité d'avoir une face ou une pile sera divers

Si pile et l'échec alors la face et la réussite

I.2.5.2. La Loi Binomiale :

Cette loi utilisée dans le cas de l'existence de deux possibilités (0,1)

- 0 c'est l'échec q

- 1 c'est la réussite p

Soit la répétition de n épreuves indépendante de Bernoulli de paramètre p .

La loi de la variable aléatoire X égale au nombre de succès dans la répétition de ces n épreuves est dite binomiale :

$$P(X=k) = C_n^k p^k (1-p)^{n-k}$$

$E(x) = n \cdot p$ | l'espérance de x est $[n : \text{le nombre de succès fois sa probabilité } p]$

$V(x) = n \cdot p \cdot q$ avec $p + q = 1$ | la variance de x est $[n : \text{le nombre de succès fois sa probabilité } p \text{ fois la probabilité de l'échec}]$ [5].

✓ **Exemple d'utilisation :** un jet d'une pièce de monnaie (non truquée)

Probabilité de pile = 0.5

Probabilité de face = 1 - 0.5 = 0.5

Exemple : Si x suit une loi binomiale de paramètre $n = 20$, $p = 1/4$

$$P(x = k) = C_{20}^k p^k (1 - p)^{n-k}$$

Par exemple : la probabilité $P(x \leq 2)$

$$\begin{aligned} P(x \leq 2) &= C_{20}^0 P^0 (1 - P)^{20} + C_{20}^1 P^1 (1 - P)^{19} + C_{20}^2 P^2 (1 - P)^{18} \\ &= C_{20}^0 \left(\frac{1}{4}\right)^0 \left(1 - \frac{1}{4}\right)^{20} + C_{20}^1 \left(\frac{1}{4}\right)^1 \left(1 - \frac{1}{4}\right)^{19} + C_{20}^2 \left(\frac{1}{4}\right)^2 \left(1 - \frac{1}{4}\right)^{18} \\ C_h^k &= \frac{n!}{k! (n - k)!} \Rightarrow C_{20}^0 = \frac{20!}{0! (20 - 0)!} \end{aligned}$$

à la fin on obtient la probabilité de succès avec un $x \leq 2$

I.2.5.3. La loi géométrique :

Est une répétition de la loi de Bernoulli jusqu'à obtenir un succès.

La loi de la variable aléatoire discrète X porte le nom de loi de Pascal ou loi géométrique de paramètre p telle que [5] :

$$P(x=k) = q^{k-1} p \quad \text{et : } k \in \mathbb{N}^*$$

$k-1$ échecs réussite

$E(x) = 1/p$ | l'espérance de x est un sur la probabilité de succès

$V(x) = 1/p^2$ | la variance de x est un sur la probabilité de succès au carré

On note $x \sim \text{geom}(p)$

✓ **Exemple d'utilisation :**

On reprend l'expérience de jet d'une pièce de monnaie, si on considère que le pile est un échec et face est un succès, alors on jette la pièce de monnaie jusqu'à obtenir face

I.2.5.4. La Loi de poisson $p(\lambda)$:

C'est l'approximation de la loi Binomiale pour un grand nombre de tirages

$n \rightarrow +\infty$ [6].

$$p(x = k) = e^{-\lambda} \frac{\lambda^k}{k!}, \quad x \in \mathbb{N}$$

λ : le nombre moyen de succès correspondant à l'intervalle de temps (constante positive).

On note : $x \sim \text{poisson}(\lambda)$.

Et on a : $E(x) = V(x) = \lambda$.

✓ **Exemple d'utilisation :** si X suit une loi de poisson

λ : Le nombre moyenne d'arrivée des clients ou des produits dans un intervalle de temps (10 Produit /2 Heurs)

Si on cherche la probabilité d'avoir 20 produit alors $X=20$

I.2.6. Les lois continues classiques :

Par définition, les variables aléatoires continues prennent des valeurs continues sur un intervalle donné.

I.2.6.1. Loi uniforme :

La variable aléatoire X suit une loi uniforme est la loi exacte de phénomènes continus uniformément sur le segment $[a, b]$ avec $a < b$ si sa densité de probabilité est donnée par [5] :

$$f(x) = \begin{cases} \frac{1}{b-a} & , x \in [a, b] \\ 0 & , \text{si non} \end{cases}$$

$$E(x) = \frac{a+b}{2} \quad , \quad V(x) = \frac{(b-a)^2}{12}$$

On note : $x \sim \text{uniforme}(a, b)$.

✓ **Exemple d'utilisation :**

Six suit une loi uniforme avec $x \rightarrow [-2, 3]$

$$f(x) = \begin{cases} \frac{1}{5} & \text{sur } [-2; 3] \\ 0 & \text{sinon} \end{cases}$$

Quel est la probabilité pour que l'intervalle soit entre 0,5 et 0,7

$$\Rightarrow \int_{0.5}^{0.7} \frac{1}{5} dx = \left[\frac{x}{5} \right]_{0.5}^{0.7} = \frac{0.7 - 0.5}{5} = 0.04$$

0,04 c'est la probabilité pour x soit entre 0,5 et 0,7.

I.2.6.2. Loi triangulaire :

La Loi triangulaire continue sur le support $[a, b]$ et de mode m est définie sur

La densité suivante sur $[a, b]$:

$$f(x) = \begin{cases} \frac{2(x-a)}{(m-a)(b-a)} & \text{si } a \leq x \leq m \\ \frac{2(b-x)}{(b-m)(b-a)} & \text{si } m \leq x \leq b \\ 0 & \text{si non} \end{cases}$$

$$E(x) = \frac{a+b+m}{3}, \quad V(x) = \frac{a^2+b^2+m^2+ab-ac-bc}{18}$$

On utilise cette loi lorsqu'on dispose d'une estimation du minimum du maximum et de la valeur la plus probable [1].

✓ **Exemple d'utilisation :**

La durée de service dans un processus de production suit une loi géométrique (2, 4, 6) (heures) ça veut dire que la durée de service au minimum c'est 2 heures, maximum c'est 6 heures et le mode (la valeur la plus probable) c'est 4 heures.

I.2.6.3. Loi exponentielle :

Une variable aléatoire x absolument continue est dite variable aléatoire exponentielle de paramètre α (constante positive) si sa densité de probabilité est :

$$f(x) = \begin{cases} \alpha e^{-\alpha x}, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

Et la distribution :

$$F(x) = \begin{cases} 1 - e^{-\alpha x}, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

$$E(x) = \frac{1}{\alpha}; \quad V(x) = \frac{1}{\alpha^2}$$

La loi exponentielle est la seule loi continue à permettre la prise en compte de phénomène set modélise la durée de vie de ce phénomène sans mémoire, ou sans vieillissement ou sans usure [1].

Cette loi est souvent utilisée en pratique. Par exemple, dans le cas de temps séparant les arrivées de deux « clients » successifs dans l'étude d'un phénomène d'attente, ou dans le cas d'une durée de bon fonctionnement d'un équipement technique.

✓ **Exemple d'utilisation :**

La durée de vie d'une machine $x \geq 3$ ans avec sa paramètre $\lambda = 0,32$

$$\begin{aligned} P(x \geq 3) &= (1) \Rightarrow e^{-0,32 \times 3} - e^{-0,32 \times (+\infty)} \nearrow 0 \\ &= e^{-0,32 \times 3} \simeq 0,38 \end{aligned}$$

0,38 est la probabilité por la duréde vie d'une machine plus de 3ans.

I.2.6.4. Loi normalisé(ou de la place –gauss) : N (m, σ)

Une variable aléatoire x absolument continue est dite normale (ou une suit une loi normale, ou gaussienne) si elle admet pour densité de probabilité la fonction :

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{(x-m)^2}{2\sigma^2} \right], x \in \mathbb{R}$$

Où: m et σ sont respectivement la moyenne et l'écart- type.

Cette loi s'applique dans le cas de processus dont la distribution est symétrique et pour lesquels la moyenne et l'écart –type sont estimés [1].

✓ Exemple d'utilisation :

Par exemple dans les domaines :

- Durée de vie des ampoules.
- Espérances de vie moyenne.

I.2.7. Les méthodes de modélisation :

Plusieurs méthodes de modélisation sont aujourd'hui utilisées. Chacune d'elles étant plus ou moins bien adaptée à des aspects spécifiques d'analyse des performances d'un système donné.

Les classes de méthodes de modélisation se divisent en deux grandes parties [6]:

- les méthodes **descriptives**
- Les méthodes **analytiques**

I.2.7.1. Les méthodes descriptives :

Elles permettent de décrire le comportement logique d'un système généralement sans faire l'aspect temporel [6].

I.2.7.2. Les méthodes analytiques :

Elle procède par décomposition du sujet.

On décompose un ensemble en ses éléments constitutifs, ses éléments essentiels, afin d'en saisir les rapports et de donner un schéma général de l'ensemble [5].

- ✓ **Exemples** : Analyse qualitative / quantitative : décomposer l'ensemble pour déterminer la nature et les proportions des constituants.

I.2.8. Les réseaux de Pétri :

Les Réseaux de Pétri ont été inventés par Carl Adam Pétri, un mathématicien Allemand, au début des années soixante (1960-1962).

I.2.8.1. Définitions d'un RDP :

Un RDP (réseaux de pétri) est un graphe biparti (composé de deux types de nœuds) [7]:

- **Les places** (P_i) qui permettent de décrire les états du système modélisé. L'ensemble de ces places est noté $P = \{P_1, P_2, \dots\}$.

- Les places peuvent contenir des jetons, représentant généralement des ressources disponibles.

- La distribution des jetons dans les places est appelée le marquage de RDP et

Chaque place contient un nombre entier (positif ou nul) de marques ou jetons.

- **Les transitions** (T_i) qui représentent les changements d'états. L'ensemble de ces transitions est noté $T = \{T_1, T_2, \dots\}$.

Les Places et transitions sont reliées par des arcs orientés.

A chaque arc, on attribue un poids (nombre entier). Par défaut ce nombre est égal à 1.

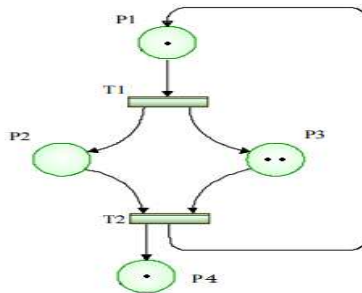


Figure I.2 : Un modèle sur les réseaux de pétri (RDP).

M_0 = Marquage initial (c'est la distribution des jetons dans les places)

La modélisation par le RdP est une méthode descriptive (**Page 9**)

On n'utilise pas la modélisation par le réseau de pétri car cette méthode donne un modèle graphique sans l'intervenir de l'aspect temporel. Nous basons sur les méthodes analytiques (Page 9) puisque cette méthode est intervenir l'aspect temporel.

✓ Exemple d'utilisation d'un RdP :

Model de remplage des bouteilles de l'eau.

I.2.9. Les systèmes d'attente :

La théorie des files d'attente est une technique de la Recherche opérationnelle qui permet de modéliser un système admettant un phénomène d'attente, de calculer ses performances et de déterminer ses caractéristiques pour aider les gestionnaires dans leurs prises de décisions.

I.2.9.1. Modèle de Files d'attente :

Tous les exemples de phénomènes d'attente ont des caractéristiques communes que l'on peut résumer ainsi : Des entités circulent dans un système et utilisent des ressources communes. Le système, les entités ou les ressources peuvent avoir un comportement imprévisible, c'est-à-dire dans le contexte d'une modélisation mathématique, aléatoire. Une file d'attente se compose des éléments suivants :

- La population.
- Les produits.
- Le nombre de serveurs.
- Les tendances quant à l'arrivée et au service.
- L'ordre de traitement des clients.
- Une salle d'attente, c'est-à-dire un lieu où les clients ou les produits attendent quand aucun des serveurs n'est disponible pour les servir.

➤ Fille d'attente infinie avec un serveur(s=1) : M/ M / 1

Considérons un modèle de file d'attente où les arrivées et les départs se produisent comme dans un processus de naissance et de mort où i.e., indépendants du nombre de clients dans le système

$$\left. \begin{array}{l} \lambda_n \equiv \lambda \quad \forall n \\ \mu_n \equiv \mu \quad \forall n \end{array} \right\} :$$

➤ Fille d'attente infinie avec s serveurs: M/ M /s

Considérons un modèle de file d'attente où les arrivées et les départs se produisent comme dans un processus de naissance et de mort où chaque serveur a un taux de service μ , mais le taux de service dépend du nombre de clients dans le système; si le nombre est inférieur à s, un sous ensemble de serveur égal au nombre de clients sont actifs; si le nombre de clients est supérieur ou égal à s, les s serveurs sont actifs.

$$\left. \begin{array}{l} \lambda_n \equiv \lambda \quad \forall n \\ \mu_1 \equiv \mu \\ \mu_2 \equiv 2\mu \\ \mu_3 \equiv 3\mu \\ \vdots \\ \mu_{s-1} \equiv (s-1)\mu \\ \mu_n \equiv s\mu \quad \forall n \geq s \end{array} \right\}$$

➤ Fille d'attente finie avec 1 serveur(s=1) : M/ M /1/K

Considérons la situation où le système a une capacité finie ; i.e., si le nombre de clients dans le système est K, alors il ne peut entrer dans le système et il est perdu. Nous avons donc un modèle de fille d'attente où les arrivées et les départs se produisent comme dans un processus de naissance et de mort où [8] :

$$\lambda_n = \begin{cases} \lambda & \text{si } n \leq K-1 \\ 0 & \text{si } n \geq K \end{cases}$$

$$\mu_n = \begin{cases} \mu & \text{si } n \leq K \\ 0 & \text{si } n \geq K+1 \end{cases}$$

✓ La méthode de marche :

- Les clients ou les produits arrivent selon une loi de poisson de taux λ .
- Le taux de service suit une loi exponentielle de paramètre μ (la durée minimum/ la durée maximum d'un service).
- Un serveur traite les clients ou les produits un à un selon le mode **P.A.P.S** (Premier qui Arrivée, Premier qui Sort)

Exemple :

Les machines qui assemblés les pièces

M/M/1

Une seule machine qui assemble les pièces :

$\lambda = 5$ pièces / 15 minutes.

$\alpha = 3$ minutes / 4 minutes.

M/M/S

Plusieurs machines qui assemblés les pièces :

$\alpha(1) = 3$ minutes / 4 minutes.

$\alpha(2) = 3$ minutes / 4 minutes.

$\alpha(n) = 3$ minutes / 4 minutes.

M/M/1/K

Une seule machine qui assemble les pièces :

$\lambda = 5$ pièces / 15 minutes.

$\alpha = 3$ minutes / 4 minutes.

$K =$ de 1 à K pièces au max dans la file d'attente

M/M/S/K

Plusieurs machines qui assemblés les pièces avec K pièces au max dans la file d'attente

$\lambda = 5$ pièces / 15 minutes.

$\alpha(1) = 3$ minutes / 4 minutes.

$\alpha(2) = 3$ minutes / 4 minutes.

$\alpha(n) = 3$ minutes / 4 minutes.

K= de 1 à K pièces au max dans la file d'attente

I.3. La simulation :

I.3.1. Notion de simulation :

Simuler le fonctionnement d'un système c'est imiter son fonctionnement au cours de temps en manipulant un modèle.

C'est donc une méthodologie essentiellement pratique qui permet de modéliser aussi bien des systèmes bien conceptuels que des systèmes existant déjà.

Elle peut être utilisée pour :

- décrire et analyser la dynamique d'un système.
- Répondre à la question de type "what if ? (qu'est-ce qui se passerait si) " sur le système réel.
- Aider à la conception d'un système réel [1].

I.3.2. Définition de la simulation :

La simulation est une technique, permettant d'étudier le comportement d'un système dynamique en construisant un modèle logiciel de celui-ci [1].

La simulation est un processus qui consiste à:

Observer le système réel à partir duquel on va concevoir un modèle à étudié, et sur ce dernier on va effectuer des études théoriques.

Après on va implémenter ce modèle ($\Rightarrow$ modèle logiciel sous arena) pour faire des expérimentations sur lui-même (et non pas des calculs), et en fin on va interpréter les observations fournies par le déroulement du modèle et formuler des décisions relatives au système.

L'objectif de la simulation peut être de comprendre le comportement dynamique du système, de comparer des configurations, d'évaluer différentes stratégies de pilotage, d'évaluer et d'optimiser des performances. Et l'élimination des comportements indésirables avant la phase d'application.

I.3.3. L'utilité de la simulation :

La simulation est très utile dans les cas suivants :

- **Le système n'est pas décomposable** en sous-systèmes plus simples et indépendants les uns des autres.
- **Le système n'existe pas** encore et une étude préliminaire (phase de conception) est en cours.
- **Les expériences sur le système réel coûtent cher**, ou sont impossibles à réaliser pendant son fonctionnement.
- **Le système est inaccessible** : le système circulatoire en est un excellent exemple.
- **Les temps d'observation nécessaires** sur le système réel sont incompatibles avec les besoins : l'évolution du système est très rapide, de sorte que, à l'échelle humaine,

les changements d'état du système semblent continus, ou cette évolution est au contraire très lente, de sorte que le système n'évolue pas à l'échelle de temps dont on dispose pour mener l'étude (c'est généralement le cas d'un système écologique, qui évolue très lentement à l'échelle humaine).

• **Faire des expériences sur le système réel est dangereux** ou peut avoir des conséquences graves : pensons au système circulatoire, ou au danger que peut représenter la mise en œuvre d'une expérimentation dans une raffinerie de pétrole...

I.3.4. Simulations Informatiques :

Processus de création d'un modèle « informatisé », dans le but de mener des expériences numériques pour comprendre le comportement du système réel, sous des conditions bien définies [9].

- Reposent sur des modèles mathématiques et logiques.

Populaire car:

- Coût (relativement) faible
- Facilité apparente de mise en œuvre
- Utilisation aussi pour communiquer et convaincre
 - Le coût de ces simulations va en diminuant grâce à la baisse du coût du matériel, la réutilisabilité des logiciels et des Framework de développement, les améliorations en architecture logicielle.
 - Liens avec la visualisation 3D (améliorations simultanées des algorithmes et des matériels), la réalité virtuelle.
 - Progrès considérables en ergonomie (« dessiner » son modèle).

I.3.5. Les domaines d'application :

Les domaines d'application sont divers. Sont listés ci-dessous quelques classes d'applications et quelques exemples de problèmes typiques rattachés à ces classes [1]:

- **Systèmes de flux de production**

- équilibrage de lignes d'assemblage.
- conception de systèmes de transfert entre des postes.
- dimensionnement des stocks d'un atelier.
- comparaison de pilotage.
- évaluation de la charge prévisionnelle.
- étude de la synchronisation entre les réceptions des pièces et l'assemblage.

- **Flux logistiques et systèmes de transport**

- conception et dimensionnement d'entrepôts.
- dimensionnement d'une flotte de camions.
- étude de procédures de contrôle des flux de véhicules en circulation.

- **Production des services**

- étude de transactions bancaires.
- gestion de restaurants.

➤ Systèmes informatiques et télécommunications

- étude de la file d'attente mémoire d'un serveur.
- étude des comportements des utilisateurs.

➤ Autres classes d'applications

- domaine militaire (coordination des opérations, ...),
- gestion d'hôpitaux (personnel, lits, service d'urgence, ...),
- la météo, les jeux, ...

I.3.6. Les limites de la simulation :

Certaines limites sont dues à la technique elle-même, d'autres sont dues aux produits actuellement disponibles sur le marché (notons qu'une formation sur les logiciels utilisés est souvent nécessaire) [1]:

- La programmation demande un certain niveau d'expertise. La qualité des résultats fournis lors de l'analyse des solutions est liée à la qualité de la modélisation et au savoir-faire du «modélisateur» (la modélisation est un métier).
- La simulation n'est pas une technique d'optimisation au sens propre. Elle ne peut qu'établir les performances d'une solution conçue et imaginée par l'utilisateur. C'est une technique entièrement itérative qui ne propose pas de solution finale mais qui permet seulement à l'utilisateur d'envisager des choix possibles. En tout état de cause, c'est lui qui devra décider de ce qui répond le mieux aux problèmes posés.

I.3.7. Méthodologie générale :

Nous distinguons classiquement quatre phases distinctes:

La modélisation (représenter le comportement du système), la programmation, l'expérimentation et l'interprétation des résultats (accompagnée d'actions) [8].

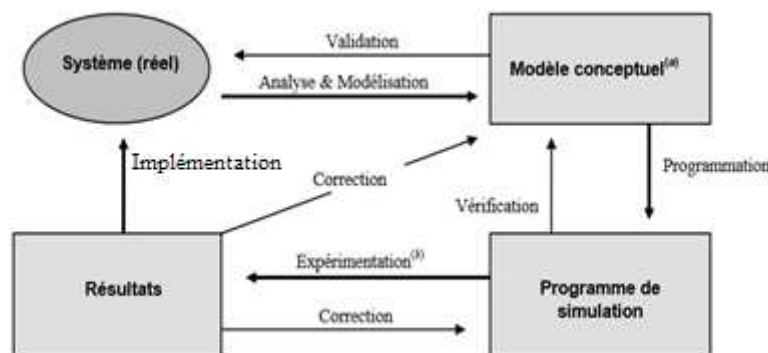


Figure I.3 : Méthodologie d'une simulation.

Système (réel) :

- ✓ Analyse & Modélisation : L'analyse, première étape du développement d'une application (programme de simulation), consiste à définir et à modéliser les problèmes d'un système

Modèle conceptuel: Le modèle n'est qu'une approximation du système, il est conditionné Par l'objectif de l'étude.

- ✓ Le passage du système au modèle conceptuel est une étape essentielle pour la simulation.

On utilise une modélisation conceptuelle par réseaux de Pétri.

Programme de simulation :

- ✓ Programmation : Obtenir une structure du modèle (logiciel) proche de celle du système (réel) à simuler
- ✓ Expérimentation : Il s'agit de construire des théories, ou hypothèses, qui prennent en Compte le comportement observé.

Résultats :

- ✓ Implémentation : application des changements sur le système réel.

I.3.8. Création d'un modèle :

➤ **Démarche :**

La réalisation d'un modèle de simulation (ceci est nécessaire si l'observation du système est impossible ou bien nécessite trop de temps ou entraîne des coûts prohibitifs) implique de répondre, avant toute chose, aux questions suivantes [10]:

- Quel est le but de l'étude ?
- Quelles sont les hypothèses simplificatrices qu'il est raisonnable de faire ?
- Combien de temps faut-il observer le modèle afin de tirer des conclusions valables pour le système ?
- Que doit-on évaluer avec le modèle, et sous quelle forme doit-on rendre les résultats ?
- Comment peut-on s'assurer que le modèle obtenu est correct, c'est-à-dire qu'il fournit des estimations correctes pour les critères retenus.

I.3.9. Les différents types de modèle :

Il existe différents types de modèles:

- ✓ **modèle de système réel**:(expérimentation directe) comme modèle animal, revêtement de route, signalisation, management magasin
 - ✓ **modèle physique** : prototype c'est le premier exemplaire d'un modèle (d'une machine, d'un véhicule).
 - ✓ **modèle mathématique/logique**: simulateur de vol,...
- Chaque type de modèle possède ses avantages et ses inconvénients [11]:

Tableau 1 : les différents types de modèle.

Modèle	Avantages	Inconvénients
système réel	- Pas de problème d'approximation.	- Difficultés de contrôle. - Très coûteux
modèle physique	- Bon niveau de contrôle. - Très réaliste	- Approximation. - coûteux.
modèle mathématique/logique	- peu coûteux. - Observation faciles	- Approximation : « on y met que ce qu'on veut bien y mettre »

I.3.10. Classification des modèles de simulation :

Nous pouvons classer les modèles de simulation en deux catégories :

- ✓ les modèles de simulation **déterministes**.
- ✓ les modèles de simulation **stochastiques**.

Nous désignons par modèle de simulation **déterministe** un modèle ne faisant pas intervenir le hasard. Au contraire, un modèle de simulation stochastique nécessitera la connaissance de lois de probabilité pour représenter le système et une simulation du hasard pour décrire son fonctionnement.

La plupart des systèmes sont **stochastiques** et l'on approche souvent leur comportement en faisant intervenir des lois de probabilité [10].

Dans les modèles de simulation **stochastiques** on a : les modèles statiques et les modèles dynamiques.

- **les modèles statiques**, pour lesquels le temps n'intervient pas.

Exemple : modèle comptable permettant de calculer le bénéfice en fin d'année à l'aide d'un tableur.

- **les modèles dynamiques**, pour lesquels le comportement est une fonction du temps.
Exemple : système de maintenance dans une usine. À l'intérieur des modèles dynamiques, on distingue :

- **les modèles à événements discrets** (ou discontinus) dans lesquels les changements d'état ne surviennent que lors d'événements tels le début ou la fin d'une opération, la mise en attente d'une pièce dans un stock, la libération d'une ressource, ... Dans une simulation à événements discrets, les flux essentiels que l'on examine sont composés d'éléments isolables que l'on peut dénombrer et identifier individuellement.

- **les modèles continus**, plus adaptés aux flux continus, qui utilisent des équations mathématiques pour prendre en compte les changements d'état qui s'effectuent de façon

continue au cours du temps. Les valeurs des variables d'état sont recalculées régulièrement selon un pas d'horloge d'après ces équations.

Exemple : un réacteur chimique.

- **les modèles combinés** (ou mixtes), qui intègrent les deux aspects [13].

I.3.11. La méthode de Monte-Carlo :

La méthode de Monte-Carlo est une méthode d'approximation, au sens statistique du terme. Il n'y a pas de définition précise de ce qu'est une technique de type Monte-Carlo, mais la description la plus habituelle consiste à dire que les méthodes de ce type se caractérisent par l'utilisation du hasard pour résoudre des problèmes centrés sur le calcul d'une valeur numérique.

➤ Le principe général des méthodes de Monte-Carlo :

C'est d'utiliser le hasard simulé pour évaluer des quantités qui apparaissent comme l'espérance d'une variable aléatoire, en calculant la moyenne d'un grand nombre de réalisations indépendantes de cette variable aléatoire. On peut appliquer ces méthodes, soit à des modèles stochastiques complexes (par exemple en économie, physique) pour lesquels il n'est pas possible de mener des calculs explicites (ou même approchés), soit à des problèmes d'analyse numérique n'ayant a priori rien à voir avec l'aléatoire (système linéaires, équations aux dérivées partielles, intégrales) mais pour lesquels les méthodes de Monte-Carlo sont plus efficaces et/ou plus faciles [11].

✓ Exemple de la méthode de Monte-Carlo :

La détermination du nombre π , via l'estimation de la surface du cercle de rayon 1. On considère un grand nombre (N) de paires de nombres (r1, r2) Tirés avec une distribution uniforme. L'ensemble de ces points sont donc dans le carré de côté $[0, 1] * [0, 1]$. La proportion p de points qui sont dans le cercle unité va tendre vers π quand le nombre de points N tend vers $+\infty$ (Le point (r1, r2) est dans le cercle unité si $r_1^2 + r_2^2 \leq 1$)[18].

I.3.12. Les avantages et les inconvénients de la simulation :

✓ Les avantages :

- Développement des politique d'exploitation / gestion de ressources pour améliorer les performances du système.
- Obtenir des nouvelles informations sans perturber le fonctionnement réel.
- Expérimentation dans des délais précis [11].

✓ les inconvénients :

-Ne pas fournir des réponses faciles aux problèmes complexes : les résultats de simulation sont suivant complexes à interpréter.

- Un système peut être si complexe avec tellement d'intersections mutuelles entre ses sous-systèmes ou composantes, qu'il devient impossibles(ou extrêmement difficile) de le décrire en terme de relation mathématiques.

-Les données imprécises → résultat erroné [11].

I.4.Conclusion :

Dans ce chapitre nous présentons la modélisation et la simulation d'un système de production car la simulation sur ordinateur est un outil très utile pour bien comprendre le principe de fonctionnement d'un système. Elle aide à observer et à comprendre le fonctionnement en temps réel et à analyser plusieurs scénarios du fonctionnement et leurs avantages et leurs conventions et aussi les modèles et leur déferant type et définir quelque notion de réseau pétri.

Dans le chapitre suivant, nous avons représenté les systèmes de production comme SED et leurs mesures de performance et aussi de ce système de production.

II. Les systèmes de production :

II.1. Introduction :

Avant d'aborder la simulation informatique, il est important de présenter et de faire une Analyse du système que nous cherchons à modéliser et simuler : le SdP (système de Production) de l'entreprise. Nous verrons de manière assez succincte, ses composantes, ses Modes d'organisation, ainsi que les paramètres auxquels il est soumis.

II.2. C'est quoi un système de production ?

Le système de production est composé de l'ensemble de ressources qui permettent cette transformation. Dans cet ensemble, on distingue essentiellement quatre types de ressources: des équipements (machines, outils, moyens de transport, moyens informatiques, ...), des moyens humains qui permettent le bon déroulement du processus de transformation, des produits à différentes étapes de fabrication (matières premières, produits semi-finis, produits finis, ...), des entrepôts de matières ou des aires de stockage.

L'objectif d'un système de production est de fournir aux clients des produits de bonne qualité, à un prix compétitif et en respectant les délais. L'entreprise doit chercher en permanence à améliorer ses produits et ses délais de livraison, elle doit en outre diminuer constamment le coût de production [17].

II.2.1 La simulation d'un système de production :

La simulation d'un système industriel a pour but de reproduire sur un modèle certains comportements dynamiques du système afin d'analyser les flux dans le système et la disponibilité des ressources. Le modèle permet de décrire les particularités physiques du système avec certains traits de gestion de production. Il nécessite l'utilisation d'éléments physiques représentatifs des entités tangibles dans la situation réelle étudiée et l'emploi d'éléments logiques caractérisant les aspects conceptuels du modèle.

II.3. Les systèmes à événements discrets (SED):

Les systèmes de **production** sont basés sur les systèmes à **événements discrets** car :

Les systèmes à événements discrets (SED) sont des systèmes dynamiques dont l'espace des états est discret et dont l'évolution est conforme à l'occurrence d'événements physiques à des intervalles de temps éventuellement irréguliers ou inconnus. La trajectoire des états est constante par morceaux. Ces systèmes sont une abstraction utile pour décrire et modéliser de nombreux systèmes conçus par l'homme tels que:

II.3.1. Les systèmes de production et ateliers flexibles:

Les méthodes de production manufacturière modernes se sont orientées vers des ateliers organisés en réseaux de Machines multitâches à la place des anciennes "lignes de transfert" rigides de Machines mono-tâche. Cette organisation, qui permet la production

simultanée de petites ou moyennes séries en respectant des ratios de production, conduit à des systèmes plus souples mais beaucoup plus difficiles à gérer.

II.3.2. Les systèmes informatiques :

Nous visons là tout type de système, des plus microscopiques comme une “puce” spécialisée en traitement de signal, jusqu’aux plus macroscopiques comme un réseau d’ordinateurs communiquant par satellites.

II.3.3. Réseaux de communication et de transport :

Téléphone ou réseau ferroviaire sont des exemples de situation où des messages ou des trains doivent emprunter les mêmes voies sans se télescoper ou entrer en collision.

II.3.4. Planification de tâches :

La gestion d’un chantier de construction ou d’un projet quelconque nécessite l’articulation de tâches dont certaines peuvent se dérouler en parallèle, c’est-à-dire sans ordre préétabli, alors que d’autres supportent des contraintes de précédence. Ces tâches peuvent faire appel à des ressources communes (hommes, machines) [10].

II.4. Les différents systèmes de production :

II.4.1. La production à l’unité ou en série :

Certains biens sont produits à l’unité. Il s’agit généralement de produits spécifiques répondant à des besoins particuliers qui font l’objet d’un cahier des charges détaillé. Il peut s’agir de produits très volumineux ou très coûteux, pour lesquels on parle de production unitaire par projet (ex : viaduc de Millau, usine clé en main, ...) ou de produits très personnalisés, pour lesquels on parle de production unitaire à la tâche (ex : conception d’un logiciel sur-mesure pour une entreprise donnée, conception d’un meuble sur-mesure, ...).

D’autres biens sont produits en série. Il s’agit généralement de produits standardisés, fabriqués en quantité, pour réaliser des économies d’échelles (réduction des coûts de production du fait des quantités). La taille des séries est fonction de la demande des clients et des coûts. La taille optimale sera celle qui permet de répondre à la demande des clients pour un coût minimum. Lorsque la série porte sur de grandes quantités (plus de 100 000 unités), on parle alors de production de masse (ex : la production de DVD vierges, ...) [12].

II.4.2. La production continue ou discontinue :

Certains produits sont fabriqués en continu au cours d’un processus unique (ex : un verre, du ciment, ...). Les outils de production, fortement automatisés, sont implantés en lignes et permettent une circulation linéaire des matières premières. Les opérateurs (salariés) sont peu nombreux et sont chargés de la surveillance et de l’entretien des équipements. La production s’effectue en continue (souvent 24h sur 24 et 7 jours sur 7).

D'autres produits sont fabriqués en discontinu. Les quantités sont généralement restreintes et les produits variés. Les outils de production sont moins spécialisés et sont regroupés par nature (ex : atelier découpe, atelier ponçage, ...). Les opérateurs sont généralement plus spécialisés [6].

II.5. La production de service :

II.5.1. Les spécificités de la production de service :

Les services ont un processus de production différent des biens. Immatériels, les services ne sont pas stockables (ex : une place dans un avion). Ils sont consommés dès leur production (ex : une coupe de cheveux). Le client peut participer à la production du service (ex : choix des plats dans un restaurant) et même réaliser certains actes de la production (ex : retrait de l'argent à un distributeur d'argent) [12].

II.5.2. Le front office et le back office :

La production de service se décompose en deux types d'opérations :

- le front office, qui est la partie directement en contact avec le client (ex : guichet, comptoir, ...)
- le back office, qui la partie où le service est préparé et où les salariés travaillent en dehors de la présence du client (ex : cuisine, atelier, ...) [12].

II.6. Le choix d'un système de production :

II.6.1. Les critères de choix :

L'organisation doit opter pour le système de production qui lui permettra d'optimiser les délais de fabrication et la qualité du produit, tout en minimisant les coûts. Elle doit prendre en compte les contraintes qui lui sont propres (qualification de la main-œuvre, nature des équipements, capacité financière, ...) et celles qui sont liées à son environnement (coût des matières premières, délais d'approvisionnement, évolution technologique, ...).

Le choix d'un système de production n'est pas immuable. Il peut être remis en cause [12].

II.7. L'évolution des systèmes de production :

L'évolution des systèmes de production est étroitement liée au développement des technologies informatiques.

La robotisation de la production a permis de remplacer l'homme pour les tâches les plus répétitives, les plus pénibles et les plus dangereuses, que ce soient pour les produits ou les services. La robotisation permet d'obtenir une plus grande précision (ex : micro-informatique) et une plus grande constance du niveau de qualité. Elle a augmenté le niveau de qualification de certains emplois (pilotage des robots, ...).

L'informatisation de la production s'est poursuivie avec le transfert d'informations entre les robots pour organiser, gérer et suivre la production en cours.

La GPAO (gestion de la production assistée par ordinateur) permet de rationaliser et d'optimiser les ressources et l'ensemble des tâches nécessaires à la production. De la conception du produit à sa réalisation, les informations nécessaires sont gérées par un ordinateur [12].

II.8. Classification des éléments :

Afin de simplifier la description des systèmes, on nomme, par la suite [10]:

Client : l'objet élémentaire se déplaçant dans le système. Il peut s'agir de pièces, de personnes physiques, de paquets d'informations...

Serveur : une unité de traitement où des clients subissent une opération de traitement. Il s'agit d'une machine, d'un bureau, d'un cabinet de consultation, d'un ordinateur...

Ressource : un composant consommable ou non, nécessaire au déroulement correct du service : forêt, papier, palette, mémoire disponible...

Unité de stockage : une unité dans laquelle des clients sont stockés et sont extraits conformément à une politique de gestion. Dans le cas du bureau de poste, l'unité de stockage est une file dont la politique de gestion est premier arrivé, premier servi. Dans un stock de pièces à usiner on peut, au contraire utiliser une pile, basée sur la politique dernier arrivé, premier servi.

Station : l'ensemble constitué d'un ou plusieurs serveurs d'une ou plusieurs unités de stockage et dont le fonctionnement est autonome sans nécessairement être indépendant des autres unités. Par exemple, l'ensemble caissière-caisse-tapis roulant peut être une station de l'ensemble de traitement qui permet aux clients d'un supermarché de régler leurs achats.

Unité de transport : elle permet le transport d'un lieu à l'autre. En fait, les unités de transport relient une station à une autre, ou un serveur à une unité de stockage, ou inversement. Une unité de transport peut être un tapis roulant, ou bien un robot prenant des pièces dans un stock et les déposant sur une machine à commande numérique.

II.9. Les Mesures de performance :

Pour évaluer la performance d'une entreprise, il est nécessaire d'effectuer des mesures à tous les niveaux : la structure (les composants d'une entreprise), économique, organisationnel...

II.9.1. La performance structurelle :

- ✓ **Les entités** : sont les composants d'un système industriel (la main d'œuvre, les produits, les machines,...).
- ✓ **Les arrivés** des produits dans le système sont des produits premières ou des produits préfabriqués. Le temps de passage peut prendre une loi de répartition aléatoire ou constante.

Exemple : l'arrivée de 20 produits dans une minute (1min/20 produits), ou l'arrivée des produits entre 5 et 20 dans une minute.

- ✓ **Les processus** : qui font des opérations sur des produits (composés, réparés,...) ou donnent des services aux clients On parle de la minimisation du temps d'écoulement dans chaque processus, augmentation de la possibilité de gérer un grand nombre de produits.

II.9.2. Performance économique :

Il s'agit de mesurer les composantes de la compétitivité de l'entreprise : la compétitivité-prix et la compétitivité-hors prix.

- **La compétitivité-prix** : désigne la capacité d'un produit à attirer des clients au détriment des produits concurrents du fait de son prix. Sa mesure permet de situer la place de l'entreprise sur le marché par rapport à ses concurrents.
- **La compétitivité hors-prix** : désigne la capacité d'un produit à attirer des clients au détriment des produits concurrents du fait des éléments indépendants du prix. Elle est obtenue grâce à des éléments comme la qualité des produits.

II.10. Les travaux sur la simulation des systèmes de production(Histoire) :

Les premiers outils de simulation sur ordinateur destinés à l'ingénierie de production (appelée industrial engineering par les anglophones) sont apparus dans les années 1950. L'un des pionniers est Jay W. Forrester qui est à l'origine de la dynamique des systèmes (System Dynamics) en 1956 et de nombreuses applications sur ordinateur.

C'est dans les années 1960 que sont apparus les premiers langages informatiques destinés à la simulation à événements discrets : GPSS (General Purpose Simulation System), CSL (Continuous Simulation Language), SimScript, Simula, etc.

Les années 1970 constituent une période d'expansion pour la simulation grâce aux progrès de l'informatique. Alors qu'auparavant, la modélisation était réalisée par programmation et la simulation proprement dite obtenue uniquement par calculs, la simulation interactive et visuelle (Visual Interactive Simulation) présentée par Hurrion s'est développée et est à l'origine du premier outil « moderne », See-Why. Ces outils ont permis de s'affranchir en grande partie de la connaissance d'un langage de modélisation et ont apporté une aide supplémentaire à l'analyse des flux grâce à des animations graphiques du processus. C'est aussi dans cette décennie qu'est apparue le langage DEVS (Discrete Event System Specification) et les premières applications de simulation distribuée.

Dans les années 1980, la production industrielle a subi une profonde mutation pour s'adapter aux évolutions des marchés. De nouveaux concepts sont apparus tels que le Juste-à-Temps, les ateliers flexibles, etc. l'informatique a aussi connu une profonde évolution avec l'apparition des ordinateurs personnels, ce qui a permis un essor considérable des outils de simulation pour l'ingénierie. De nombreux logiciels ont vu le jour et la plupart existe encore aujourd'hui, sous une version plus évoluée : Witness, Hocus, Genetik, Siman, Promodel, etc.

A partir des années 1990 et jusqu'à nos jours, les outils de simulation n'ont pas apporté de grandes nouveautés mais des améliorations notables sont apparues concernant la facilité de modélisation (Visual Interactive Modelling Systems – VIMS), les possibilités d'animation et la représentation en 3D, l'ouverture ou la compatibilité avec d'autres applications informatiques (feuilles de calcul, bases de données...), la distribution sur internet, le couplage avec des algorithmes d'optimisation, etc. [13].

II.11. L'approche orientée objets :

L'approche orientée objets est de plus en plus répandue dans le développement des systèmes

Le système est décrit par un ensemble d'objets qui communiquent par envoi de messages. La modélisation orientée objets se fait en deux étapes : la première concerne la définition de la structure du système par la spécification des objets (attributs) et la seconde concerne la définition du comportement dynamique du système en précisant les relations entre objets ainsi que leur fonctionnement propre (méthodes). Cependant, à un système orienté objet, il manque un mécanisme de haut niveau permettant le contrôle des envois de messages dans le temps (comportement dynamique). Au sein du LLP/CESALP, des recherches ont été menées dans ce sens, pour plus de détails sur ce sujet nous orientons le lecteur sur les travaux de Guetari [19]. L'un des avantages de l'approche orientée objets est la combinaison des avantages des langages généraux (flexibilité et champ d'application larges) avec les avantages des outils spécialisés (objets paramétrables et construction simple des modèles).

Plus de 100 logiciels de simulation sont disponibles sur le marché. Certains auteurs classent ces logiciels en deux catégories : les logiciels pour la simulation à événements discrets et les logiciels pour la simulation à variables continues. Si le second type est applicable pour un flux continu de produit et d'information, le premier est beaucoup plus applicable dans les systèmes manufacturiers (composants, assemblages des pièces électronique,...). [19].

II.12. Les différents logiciels de simulation :

Il existe des divers outils de simulation performant pour gérer des systèmes de production et d'autre ... , Parmi les logiciels de simulation existant on a écrit sur les plus utilisée au monde :

II.12.1. SIMUL8 :

SIMUL8 est un progiciel pour la simulation à événements discrets.

Il permet de créer un modèle visuel du système à l'étude en dessinant des objets directement sur l'écran. Les objets typiques peuvent être les files d'attente ou des points de service. Les caractéristiques des objets peuvent être définies en termes de, par exemple, la capacité ou la vitesse. Une fois que le système a été modélisé une simulation peut être entreprise. Le flux d'éléments de travail à travers le système est représenté par l'animation sur l'écran afin que la pertinence du modèle puisse être évaluée [14].



II.12.2. AUTOMOD :

Le logiciel de simulation AutoMod™ répond aussi bien aux besoins d'un utilisateur occasionnel qu'à ceux d'un expert. Vous pouvez simuler facilement et avec précision des systèmes de toute taille ou de niveaux de détail, allant des opérations manuelles aux installations entièrement automatisées. Votre réussite et votre productivité sont accrues lorsque vous utilisez les capacités uniques d'appliqué AutoMod, telles que :

Animation 3-D en réalité virtuelle,
Modélisation interactive,
Modèles de systèmes automatisés,
Langage de modélisation propriétaire « English-like » [15].



II.12.3. Enterprise Dynamics :

Enterprise Dynamics est une plateforme de simulation orientée objet et événement. Vous pouvez créer un modèle en sélectionnant des objets de simulation standards dans la bibliothèque et en les glissant dans l'espace du modèle. Ces objets, appelés « atomes », simulent des comportements réels dans toute leur complexité grâce à la représentation 3D parfaitement réaliste.

Enterprise Dynamics intègre un environnement de développement intelligent :

- Créez des objets en toute simplicité.
- Modifiez les paramètres des objets pendant la conception et l'exécution du modèle.
- Réutilisez des objets dans des modèles différents.

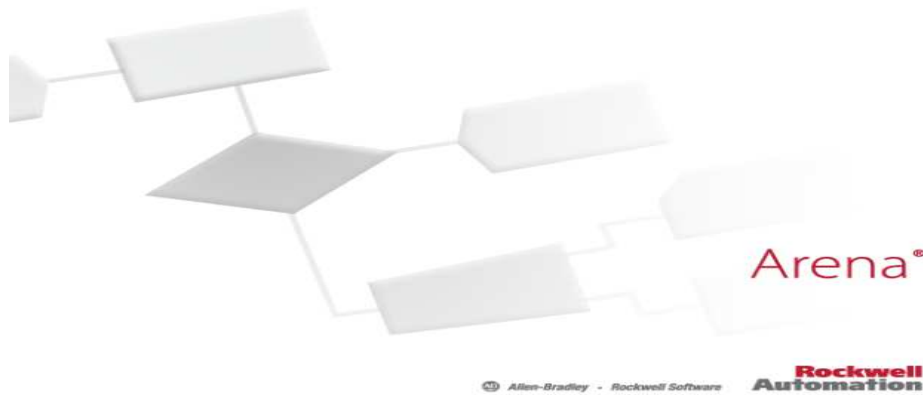
L'architecture logicielle est suffisamment puissante et flexible pour ajouter de nouvelles fonctionnalités, de nouveaux protocoles et un nombre illimité d'objets de simulation [16].



II.12.4. ARENA :

SIMAN-ARENA - conçu en 1982 par C.D. Pedgen de System Modeling Corporation est un langage de simulation du type interaction de processus, ARENA représentant la version « graphique » de SIMAN. La description du modèle (logiciel) du système simulé se fait à l'aide d'un assemblage constitué de mise en série, en parallèle ou en feedback de différents blocs fonctionnels, issus de bibliothèques (templates) d'ARENA. Une telle approche de

modélisation permet d'obtenir une structure du modèle (logiciel) proche de celle du système (réel) à simuler [2].



La simulation donnera des résultats permettant de déterminer les erreurs d'architecture de notre system et augmenter la production et minimisé le temps d'attente d'un produit ou d'un client, tout ça il gère avec des outils informatiques (logiciels) pour celas on a utilisé le simulateur **ARENA** sous **Windows** [21].

II.13. Conclusion :

Dans ce chapitre, nous avons cherché à positionner notre thématique de recherche à travers une synthèse couvrant : d'abord, les systèmes de production et les systèmes à évènement et l'évaluation de leur performance et les déférents logiciels de simulation et ensuite, les différents systèmes de production et leur état de l'art actuel dans la simulation des systèmes de production.

Ce positionnement thématique centré sur les systèmes de production est basé sur les systèmes à évènement discrets. Dans le chapitre suivant, nous allons implémenter notre travail.

III. Simulation d'Assemblage des pièces électroniques:

III.1. Introduction :

Après avoir présenté dans le chapitre II la présentation des systèmes de production basée sur les systèmes à événements discrets, nous donnons dans ce chapitre la modélisation de notre modèle et quelques détails sur la réalisation de l'application et les outils utilisés.

Nous expliquons le principe de fonctionnement de l'application via des différents extraits d'interfaces implémentées.

III.2. Pour quoi on a choisi Arena-simulation ? :

C'est un outil facile à maîtriser et l'utiliser.

- Plus de 5000 objets d'animation complexes sont incluses dans la bibliothèque de l'animation Arena.
- Animation peut également être créée par l'utilisateur. Clip Art, des bitmaps, des dessins AutoCad et beaucoup d'autres.
- La compatibilité des données : données d'importation et d'exportation de / vers un fichier de Les types de fichiers suivants: Excel, Access, XML, texte,....
- Script Visual Basic : Contrairement à d'autres outils qui utilisent des langages propriétaires de script, Arena utilise un éditeur standard VBA (inclus) et le modèle objet Arena pour construire des interfaces utilisateur personnalisées et sur mesure interfaces de données à des modèles Arena.
- Visual Basic Automation : toutes les fonctions Arena peuvent être automatisées avec la programmation Visual Basic.

III.3. Utilisation d'ARENA :

L'installation d'**ARENA** est une installation simple sauf que la version complète de ce logiciel est payante.

Dans notre projet on a utilisé la version **STUDENT** est une version gratuite destiné à un usage scolaire seulement. Cette version gratuite d'Arena :

- Offre une fonctionnalité complète.
- Le modèle limité de la taille et peut être copié sur un autre PC.
- Il n'y a aucune limite de temps.

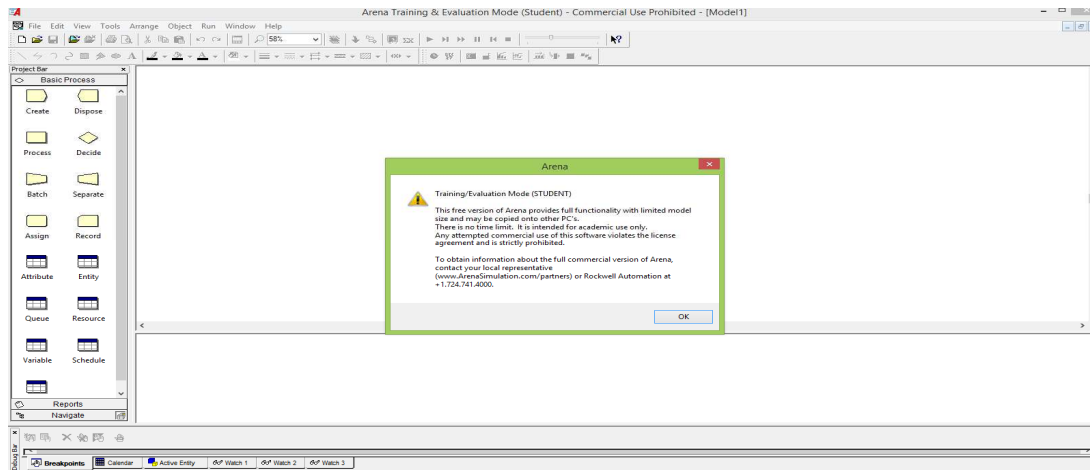


Figure 4: Image interface d'ARENA.

III.3.1. Les modules à configurer sous Arena :

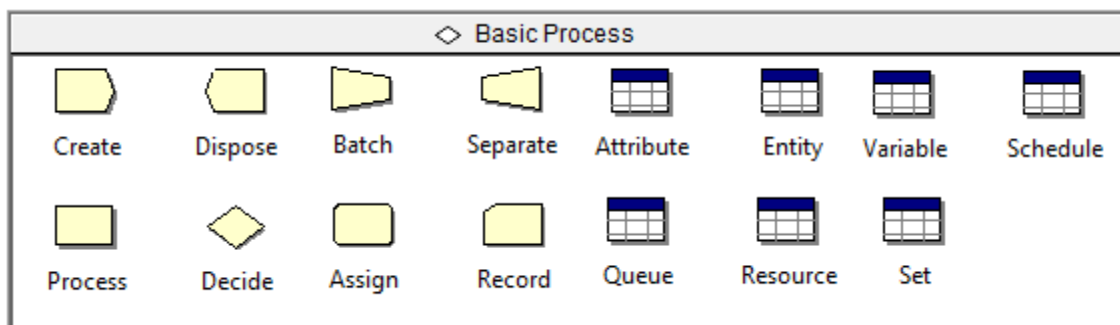


Figure 5: Basic Process d'ARENA.

- **Create:** Ce bloc est considéré comme le module de départ des entités.

Les arrivées dans le système (pièce, machine ...).

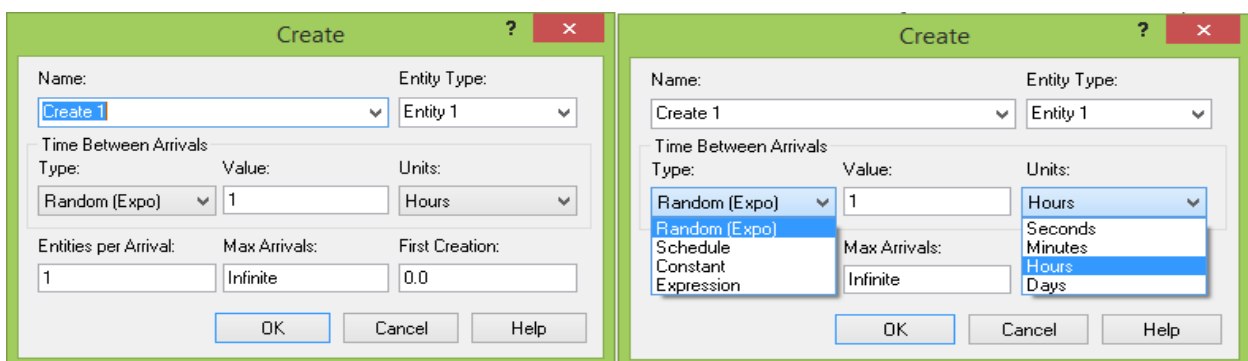


Figure6: Les paramètres de Create.

- ✓ **Type:** il y a plusieurs lois, chacun a une méthodologie différente de travail.
- ✓ **Value:** nombre des arrivées par S/M/H/D.
- ✓ **Units:** unité de temps.
- ✓ **Entities per Arrival:** N arrivées par unité de temps.

- ✓ **Max Arrivals:** maximum des arrivées par unité de temps.
- ✓ **First Creation:** le temps de début des arrivées.
- **Dispose:**

Les sorties du système (soit des pièces, machines,...).

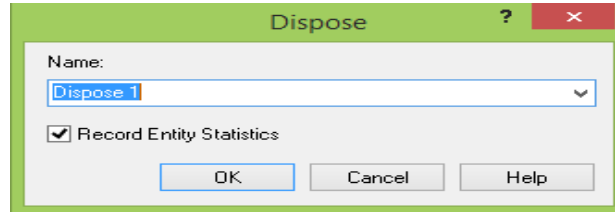


Figure 7: Les paramètres de Dispose.

- **Process:**

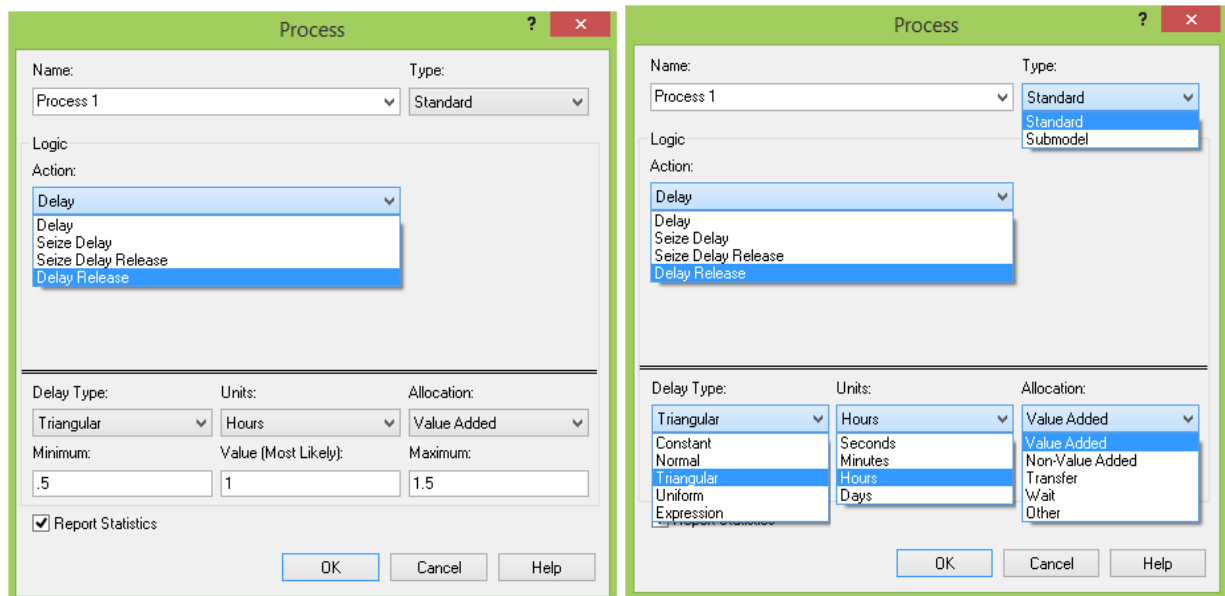


Figure 8 : Les paramètres de Process.

- **Action:**
 - **Delay:** L'entité doit être séjournée dans le bloc process pendant un temps spécifié.
 - **Seize Delay:** File d'attente en amont du bloc Delay, et Nécessité de joindre une ressource au processus
 - **Seize Delay Release:** Comme Seize Delay, mais une entité libère des unités de ressources après Delay (ce que nous voulons dans ce modèle).
 - **Delay Release:** Suppose entité à déjà saisi des unités de ressources à un autre module amont, maintenant retards et de presse unités de ressources.
 - **Delay type:** loi utilisée.
 - **Units:** unité de temps.

➤ Decide:

Décide si l'arrivée est vraie ou fausse.

Figure 9: Les paramètres de Decide.

➤ Type:

- **2-way by Chance:** soit vrai ou faux avec pourcentage de succès.
- **2-way by Condition:** soit vrai ou faux avec condition.
- **N-way by Chance:** plusieurs choix avec pourcentage de succès.
- **N-way by Condition:** plusieurs choix avec condition.

➤ Batch:

Figure 10: Les paramètres de Bach.

- **Batch Size:** divisé l'entité à 2 par exemple.
- **Rule:** attribut ou entité.
- **Save Criterion:** produit.

➤ Separate:

Figure 11: Les paramètres de Separate.

➤ Type:

- **Duplicate Original:** Revient à la forme originale.

- **Split Existing Batch:** faire le contraire d'un Batch existe.
- **Percent Cost to Duplicates (0-100):** le pourcentage pour dupliquer une entité.
- **# of Duplicates:** nombre de duplication.

➤ **Assign & Record:**

Utilisé pour calculer le temps écoulé dans un système (temps d'attente, la durée de service, ...).

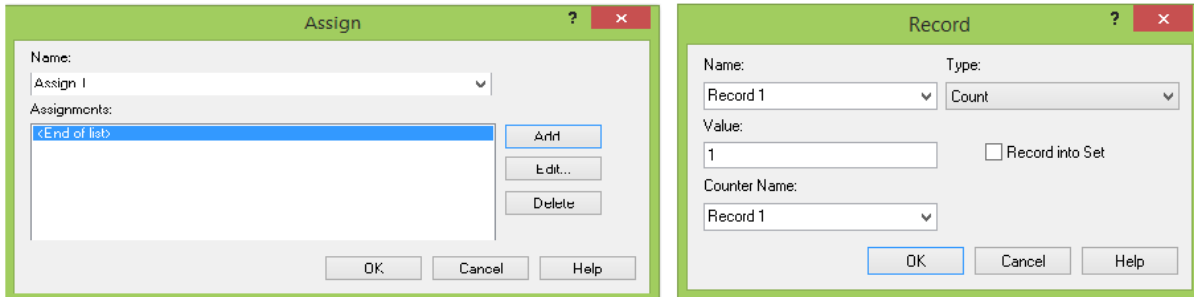


Figure 12 : Les paramètres d'Assign & Record.

III.4. La modélisation et la configuration de notre projet :

Notre cas d'étude : Assemblage électronique

III.4.1. Présentation :

- Nous voulons regrouper quatre pièces A, B, C, D pour construire une plaque électronique
1. Le système de production marche de 8:00 à 17:00, il prend une pose de un hour de 12:00 à 13:00 dans cette période les machines ne marche pas même si il y a des produits dans le système.
 2. les arrivées (les pièces A, B, C et D) n'arrive pas dans la période de une heures et aussi l'arrivée des pièces sa divers avec le temps c'est à dire (le rendement d'arrivée est plus fort dans le matin de 8:00 à 11:00 et diminuer de 11:00 à 12:00 et faible de 13:00 à 17:00.
 3. Les pièces orientes vers deux agents qui faire l'assemblage est la soudure de A avec B et C avec D.
 4. Après l'assemblage de AB et CD ces derniers pilotés vers deux machines marchent indépendamment, ces machines permettant de tester la validité des AB et CD.
 5. Si les plaques AB et CD sont valide alors elle orientes vers l'agent qui faire l'assemblage et la soudure de AB avec CD.
 6. Après l'assemblage on obtient une plaque ABCD, cette plaque piloter vers la machine qui faire le test, si la plaque est valide elle orient vers la machine de regroupement cette machine mettre dans chaque boîte 6 plaques électroniques.
 7. Si les plaques AB et CD sont invalides alors elles sont orientées vers l'agent qui fait la démontage des pièces, il y a deux agents (l'agent qui fait le démontage de AB et l'autre qui démonte la plaque CD).

8. Après la démontage de AB et CD on obtiens quatre pièces A,B,C et D chaque pièce vas orientes vers une machine de test .
9. Si les pièces sont invalides alors on les rejette, sinon on reproduise les pièces valides.
10. Les pièces A et B sont orientées vers l'agent qui fait l'assemblage de AB2.
11. Les pièces C et D sont orientées vers l'agent qui fait l'assemblage CD2.
12. Après l'assemblage de AB et CD on orientant vers les deux machines de test.
13. Si AB2 et CD2 sont valides, elles seront pilotées vers l'agent qui fait l'assemblage.
14. Si les plaques AB2 ou CD2 sont invalides alors on les rejette.
15. Après l'assemblage de ABCD 2, nous orientons les plaques vers la machines de test, si la plaque est valide nous allons mettre dans chaque boite 6 plaques, sinon on rejet la plaque.

III.4.2. Implémentation :

- Nous précédons dans la modélisation de notre cas d'étude par les étapes suivantes:
 1. L'arrivée des pièces suit un ordre : de 8:00 à 12 :00 et de 13:00 à 17:00

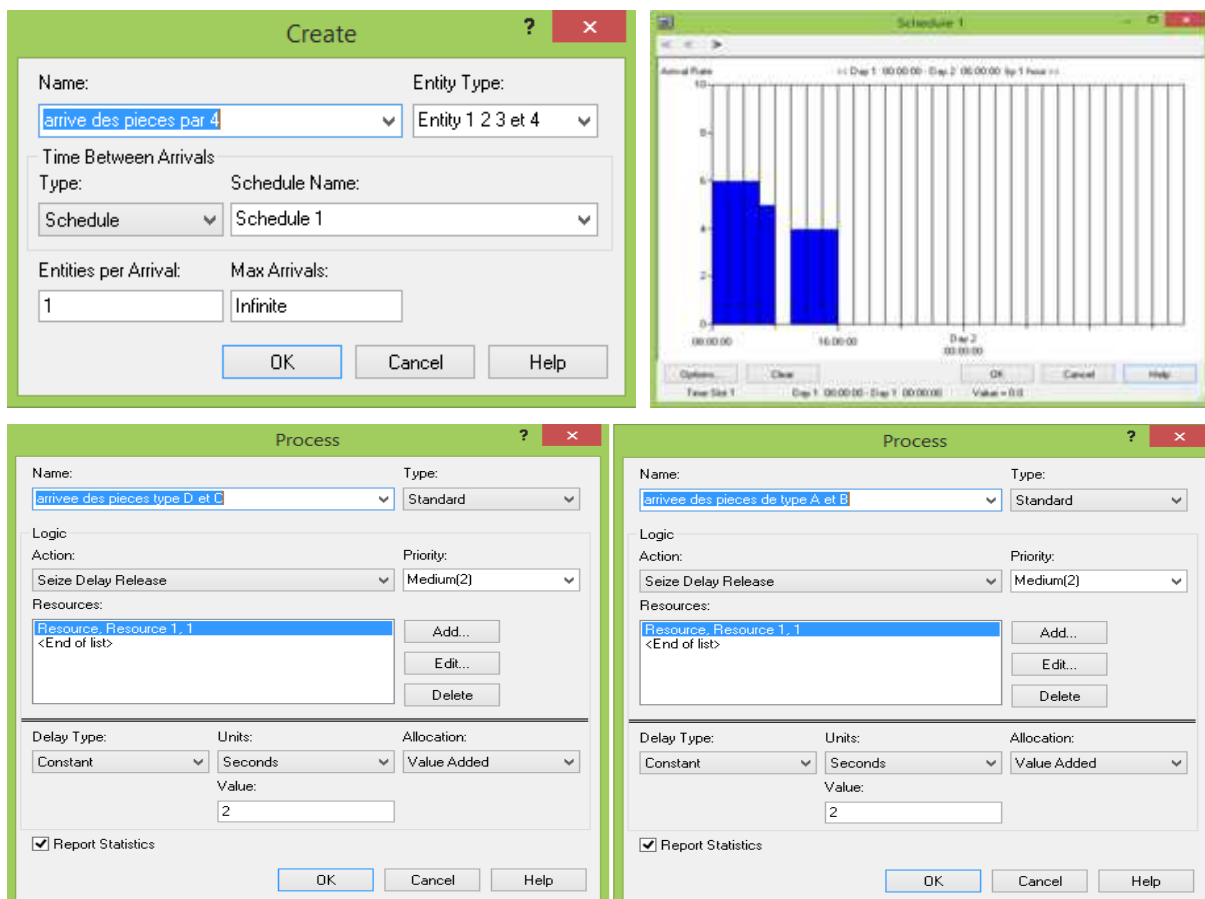


Figure13 : Configuration de Create & Process (Les arrives A, B, C, D).

2. Nous mettons deux Process <Assemblage AB> et < Assemblage CD >
 Assemblage AB permet de souder la pièce A avec B on suit la loi Triangulaire (20, 25,30) par secondes.

L'Assemblage CD permet de souder la pièce C avec D on suit la loi Triangulaire (20, 25,30) par secondes.



Figure 14 : Assemblage de A avec B et C avec D.

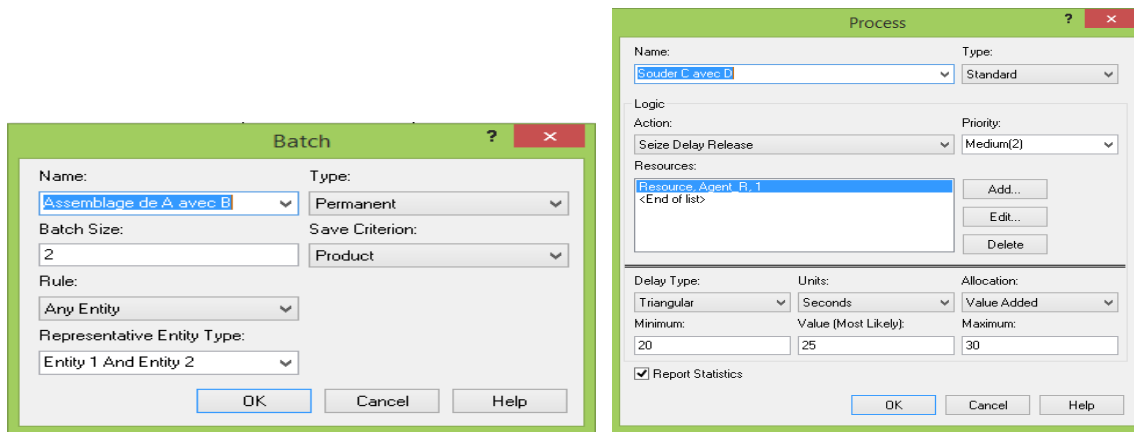


Figure 15 : Configuration de Batch & configuration de process.

- Il y a deux machines de test, ce dernier teste la validité de la plaque AB et CD, chaque machine travaille indépendamment de l'autre, la validité ou l'invalidité d'une plaque suit une method 2-way by chance le pourcentage de la validité d'une plaque électronique AB ou CD égale à 80%, la machines de test prend 5 secondes.

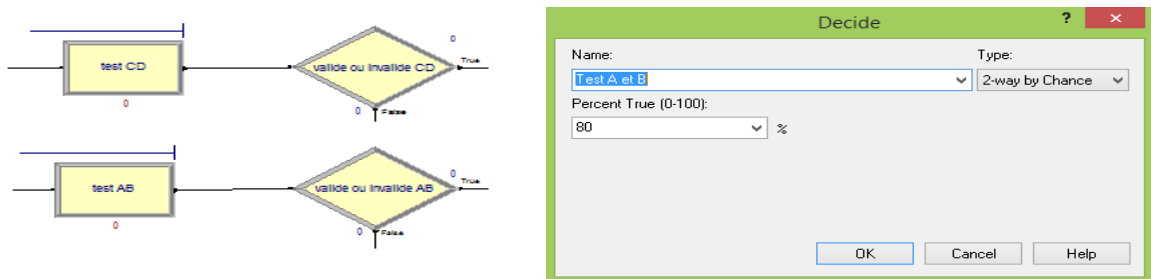


Figure 16 : Machine de test AB et CD & configuration de Decide.

Process

Name: test CD Type: Standard

Logic: Action: Seize Delay Release Priority: Medium(2)

Resources: Set, Machines, 1, Cyclical, Machine CD <End of list>

Delay Type: Constant Units: Seconds Allocation: Value Added

Value: 5

☒ Report Statistics

OK Cancel Help

Figure 17: Configuration de Process.

4. L'assemblage ABCD permet de souder la plaque AB avec CD sauf si les deux plaques sont valides, la durée de l'assemblage suit une loi Triangulaire (20, 25,30) par secondes.

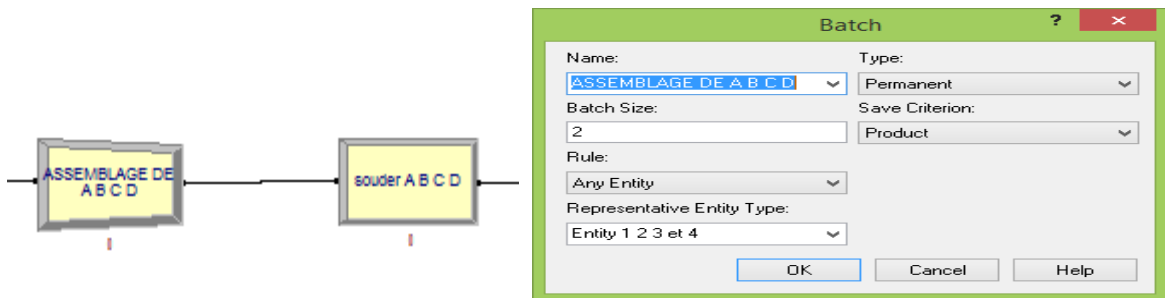


Figure 18 : Assemblage de ABCD & configuration de Batch.

Process

Name: souder ABCD Type: Standard

Logic: Action: Seize Delay Release Priority: Medium(2)

Resources: Resource, Agent, R2, 1 <End of list>

Delay Type: Triangular Units: Seconds Allocation: Value Added

Minimum: 20 Value (Most Likely): 25 Maximum: 30

☒ Report Statistics

OK Cancel Help

Figure 19: Configuration de Process.

- Nous mettons une machine de test, cette machine teste la validité de la plaque ABCD, la validité ou l'invalidité d'une plaque suit une method 2-way by chance le pourcentage de la validité d'une plaque électronique égale à 80%, la machine prendre 5 secondes pour tester si la plaque est valide.

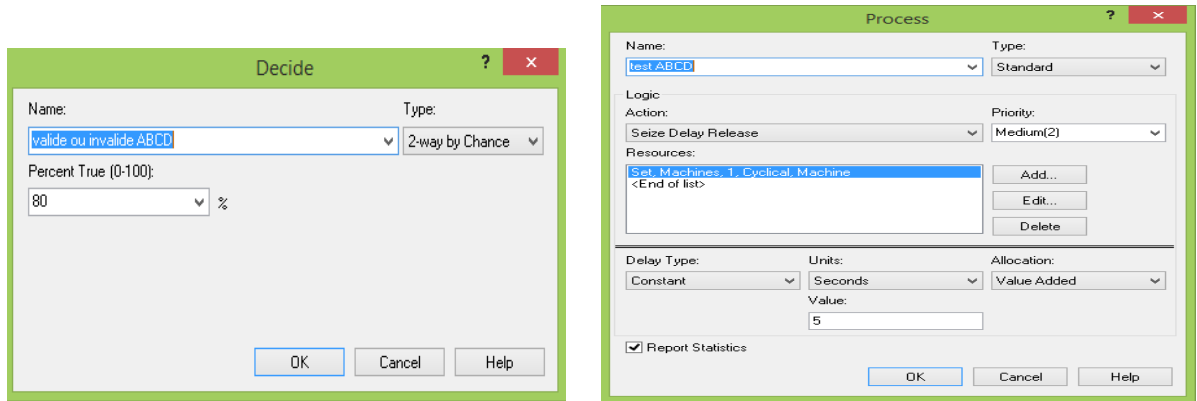


Figure 20: Configuration de Decide & Process.

- Si la plaque est valide, nous l'orientons vers la machine de regroupement (chaque boîte contient 6 plaques électroniques) et on oriente la boîte vers le stock.

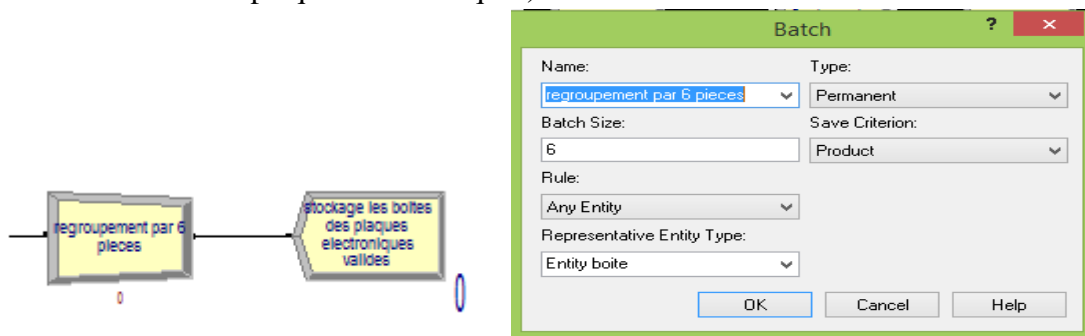


Figure 21 : Regroupement des plaques & Configuration de Batch.

- Si la plaque ABCD est invalide alors nous orientons les plaques invalides vers les agents de séparation. Premièrement, nous séparons la pièce A de la plaque ABCD et l'envoyons vers la machine de test (test que les pièces A), la validité ou l'invalidité d'une pièce suit une method 2-way by chance, le pourcentage de la validité de cette pièce égale à 80%, la machine de test prend 5 seconde, si la pièce est valide on passe vers la deuxième étape sinon on rejette.



Figure 22 : séparer la pièce A de la plaque ABCD. & test de la pièce A.

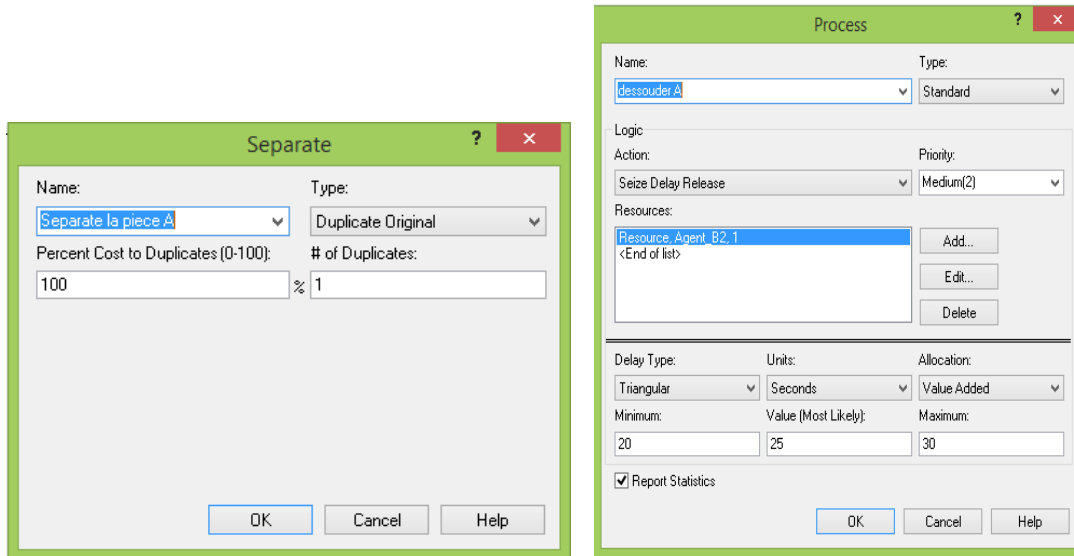


Figure 23 : configuration de Process & configuration de Separate.

8. Deuxièmement, on sépare la pièce B de la plaque BCD et on l'oriente vers la machine de test (test que les pièces B), On suit les mêmes étapes de la pièce A.



Figure 24: dessoûder B.

9. Troisièmement, on oriente la plaque CD vers l'agent responsable du processus de séparation.

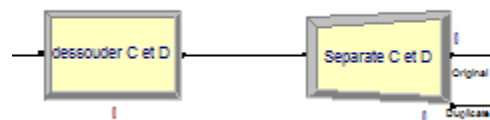


Figure 25 : dessoûder C et D.

10. Après la séparation on oriente la pièce C et la pièce D vers deux machines de test différentes, chaque machine teste la validité ou l'invalidité de pièce, la pièce valide passe vers la deuxième étape et la pièce invalide on la rejette.



Figure 26 : Test si les pièces sont valides.

11. La deuxième étape permet de reproduire les pièces valides A, B, C, D

On oriente les pièces vers deux agents différents pour obtenir AB2 et CD2

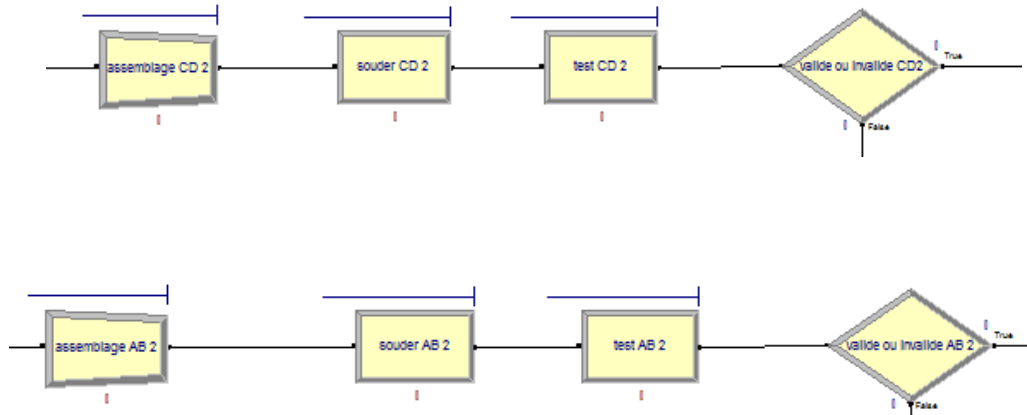


Figure 27: Assemblage & test.

Si les AB2 et CD2 sont valides alors on va piloter vers l'agent qui fait l'assemblage AB2 avec CD2, sinon on rejette.

12. Après l'obtention des plaques ABCD2 nous les orientons vers la machine de test, si la plaque est valide on regroupe par 6 plaques dans chaque boîte, sinon nous rejetons les plaques invalides.

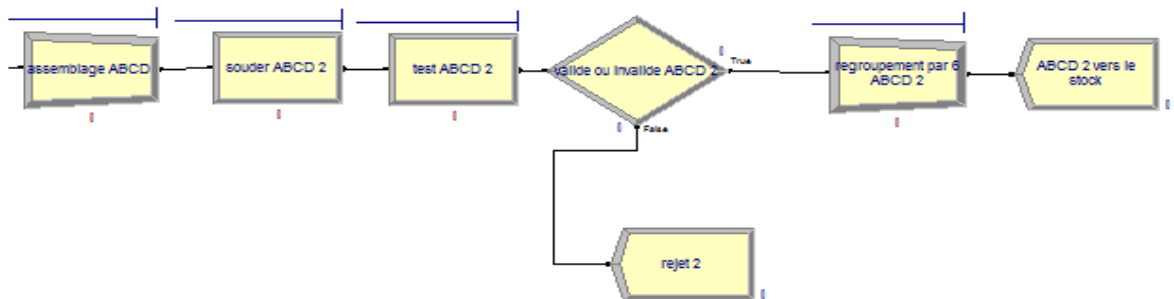


Figure 28: Assemblage & test & regroupement.

III.4.3. Le Model Générale ARENA:

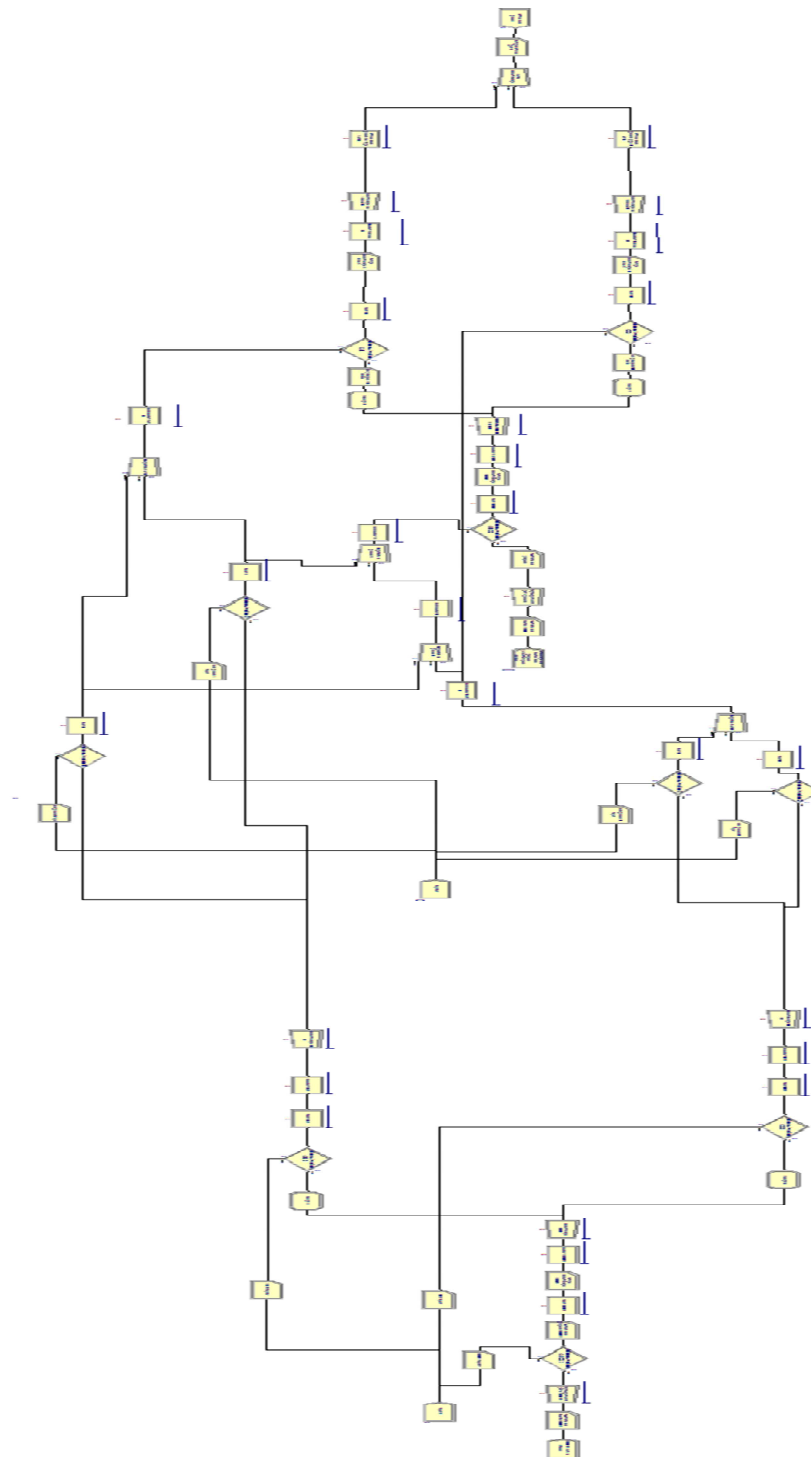


Figure 29: Le modèle général.

III.5. Animation 2D:

III.5.1.Les Agents:

Nous avons utilisé les agents pour l'animation de l'assemblage et le démontage des pièces.

➤ Les Agents rouges: Assemblage des pièces

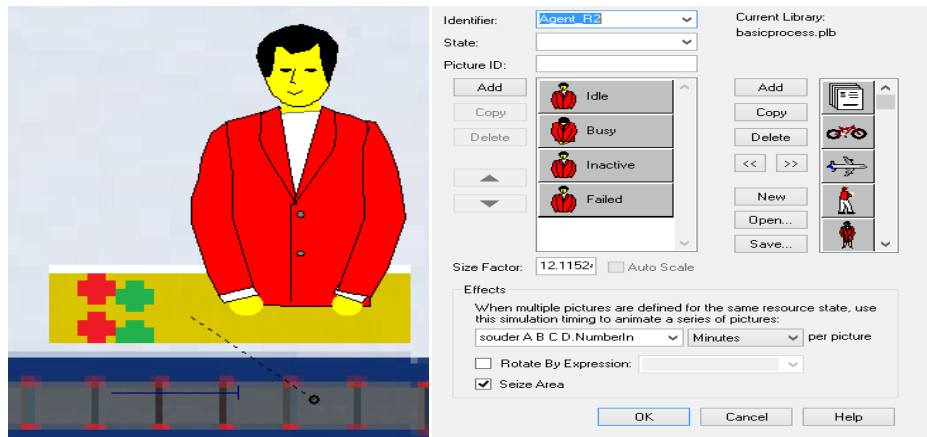


Figure 30: Agent Rouge.

Il représente le : Batch plus Process (c'est-à-dire l'assemblage et la soudure des pièces).

➤ Les Agents bleus: démontage des pièces

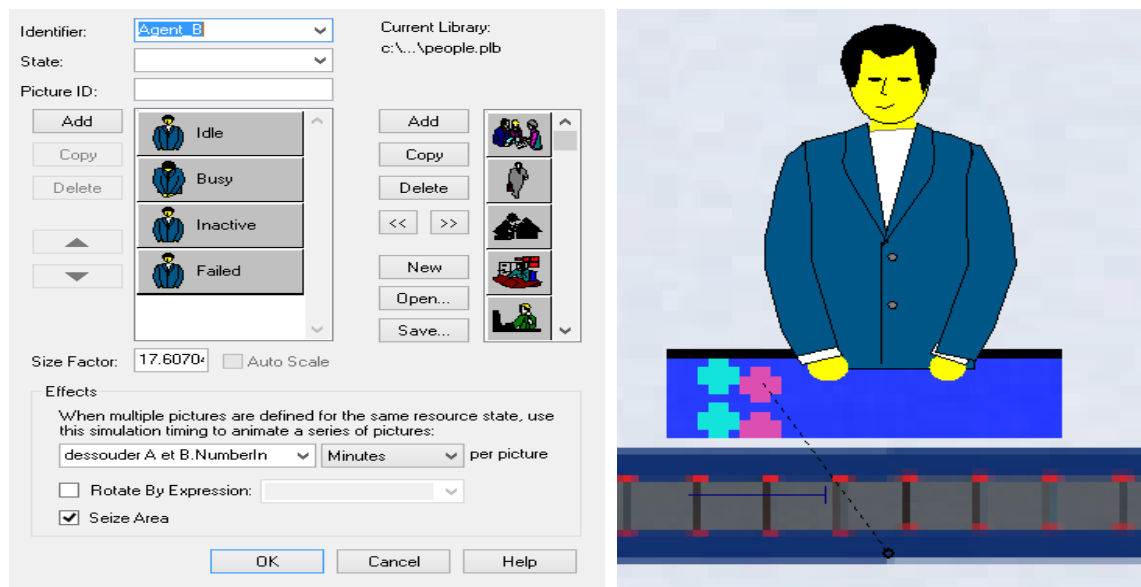


Figure 31: Agent bleu.

Il représente le : Separate plus Process (c'est-à-dire dessouder et séparer les pièces).

III.5.2. Les machines de test : test la validité des pièces et des plaques

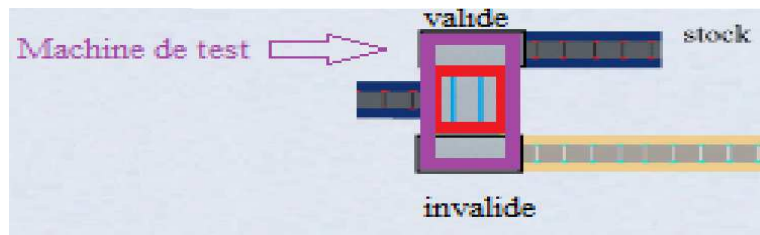


Figure 32: Machine de test.

Elle représente le Test.

III.5.3. Les agents verts : assemblage AB2 et CD2 et ABCD2

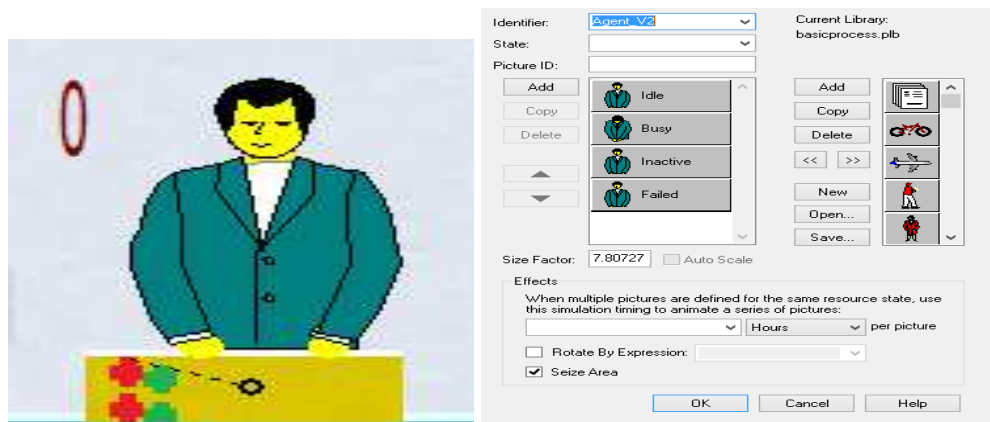


Figure 33 : Agent vert.

III.5.4. Schéma de l'atelier d'assemblage électronique :

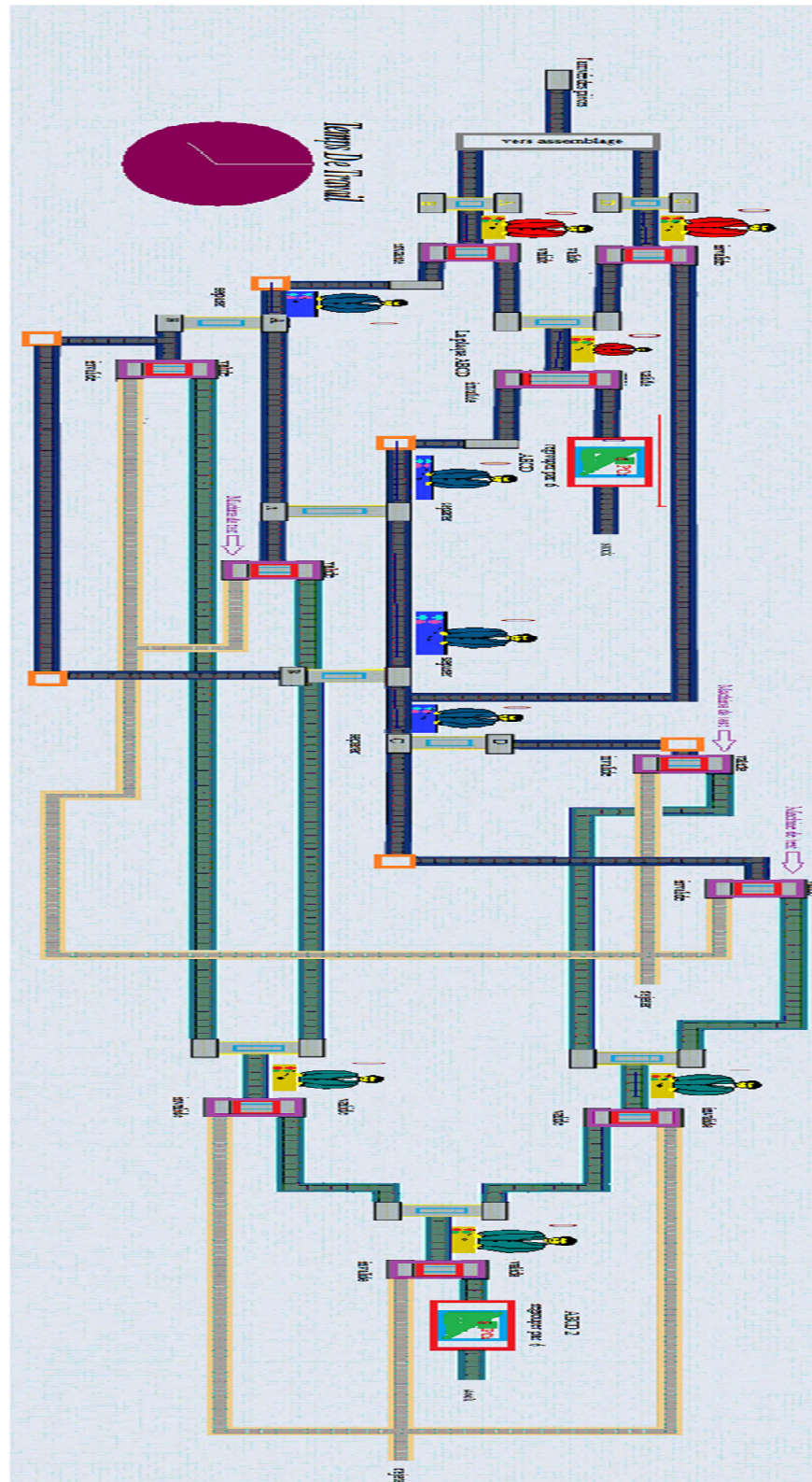


Figure 34 : Schéma général de l'atelier.

III.6. Les résultats de simulation :

Pour avoir des résultats sur la simulation, nous lisons le rapport de simulation qui se génère automatiquement à la fin de la simulation.

Chaque rapport représente les données d'un seul scénario de simulation.

Il y a des données qui sont par défaut dans le rapport comme par exemple le temps d'attente moyen dans une file d'attente d'un processus donné. D'autres informations sont à créer par l'utilisateur comme le temps moyen passé par une entité dans le système.

Record comme moyenne pour : calculer le temps moyen d'assemblage, intervalle de temps (début et fin d'une opération dans les systèmes), nombre des unités passer dans le système....

- **Calculer le temps moyen :** temps moyenne d'assemblage C avec D, par exemple :

The screenshot shows the 'Record' dialog box with the following configuration:

- Name:** temps assemblage C avec D
- Type:** Time Interval
- Attribute Name:** Temps
- Tally Name:** temps assemblage C avec D
- Record into Set:** (unchecked)

Figure 35 : configuration de Record.

- **Compter le nombre des plaques électroniques :** On utilise **Record**

The screenshot shows the 'Record' dialog box with the following configuration:

- Name:** Nombre des pieces
- Type:** Count
- Value:** 1
- Counter Name:** Nombre des pieces
- Record into Set:** (unchecked)

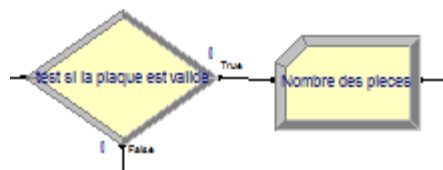
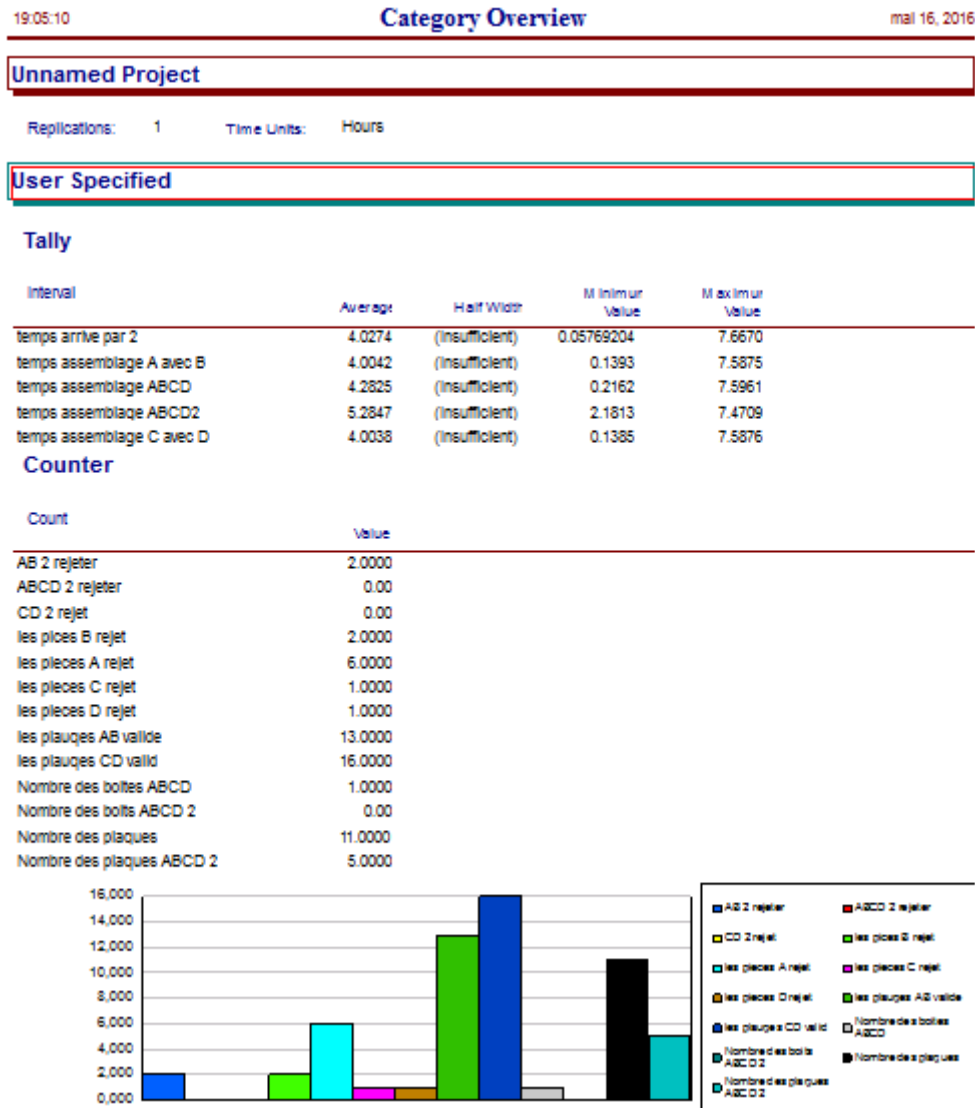


Figure 36 : configuration de Record (count).

III.7. Rapport de simulation :



Après les résultats obtenus, si les dirigeants de l'entreprise sont satisfaits alors le scénario appliqué est le scénario requis.

Sinon, nous allons changer les facteurs d'influences, par exemple :

Nous augmentons le nombre des pièces arrivées dans le temps.

Nous augmentons la qualité des pièces pour diminuer le nombre des pièces à rejeter (nous changeons la possibilité d'avoir les pièces valides vers 90% ou plus).

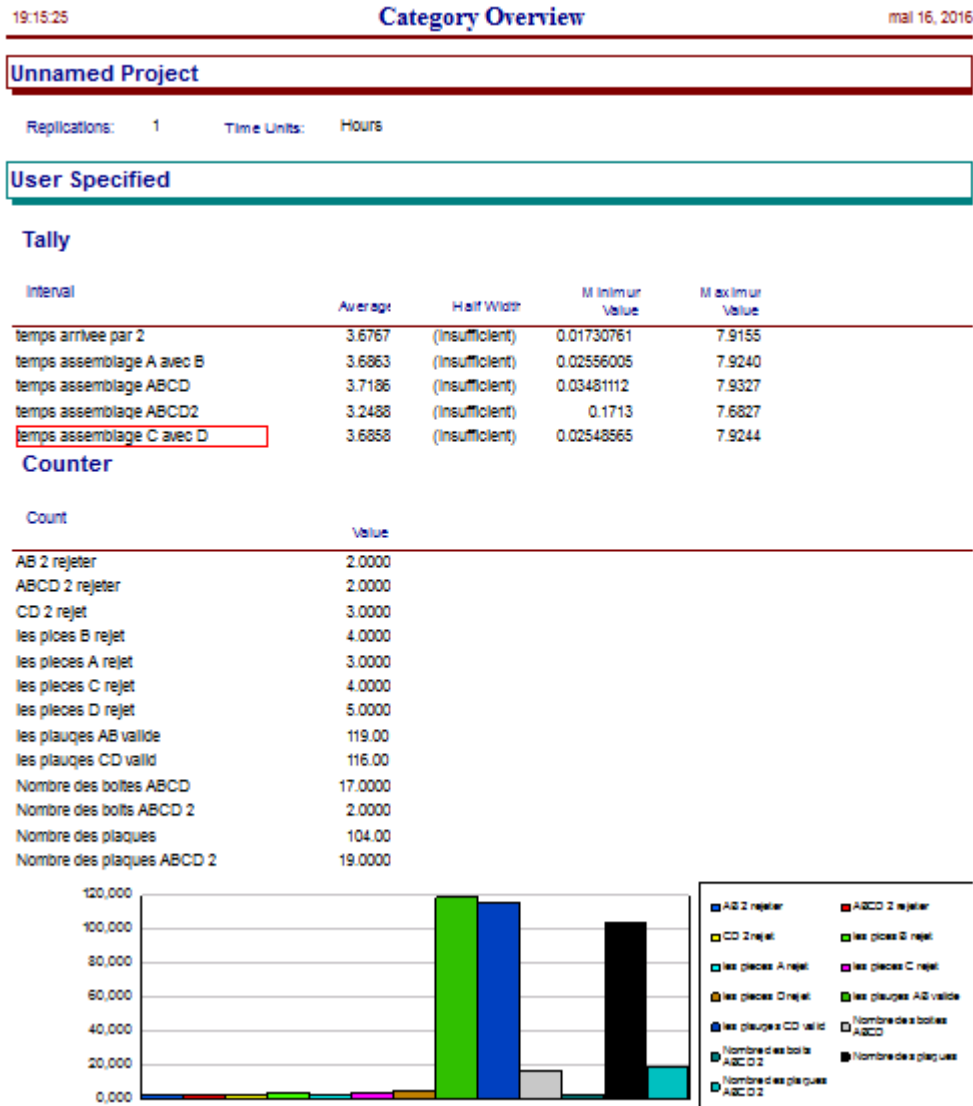
Nous avons diminué le temps d'assemblage des pièces.

III.8. Application du deuxième scénario :

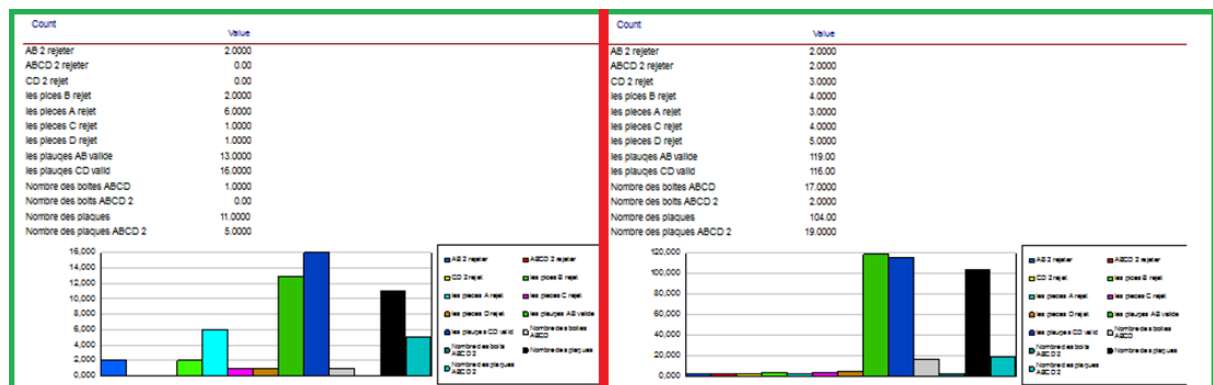
Le deuxième scénario est le même schéma que la première appliquée sauf-que on va changer le :

1. Augmentation du nombre des arrivées par le temps.
2. la possibilité d'avoir les pièces valide vers 90%.

III.8.1. Le résultat de simulation :



Nous comparons les deux scénarios :



Nous trouvons que le deuxième scénario est le mieux adapté avec les besoins d'une entreprise d'assemblage des pièces électroniques.

Après le test et l'évaluation de performance de notre modèle on peut maintenant réaliser notre projet (c'est-à-dire l'implémentation de ce projet dans la vie réelle).

III.9. Interface Visual Basic :

Nous voulons créer une interface qui permet à l'utilisateur de gérer et modifier quelques paramètres de production.

Le langage de programmation qui permet réaliser cette fonction est le Visual Basic.

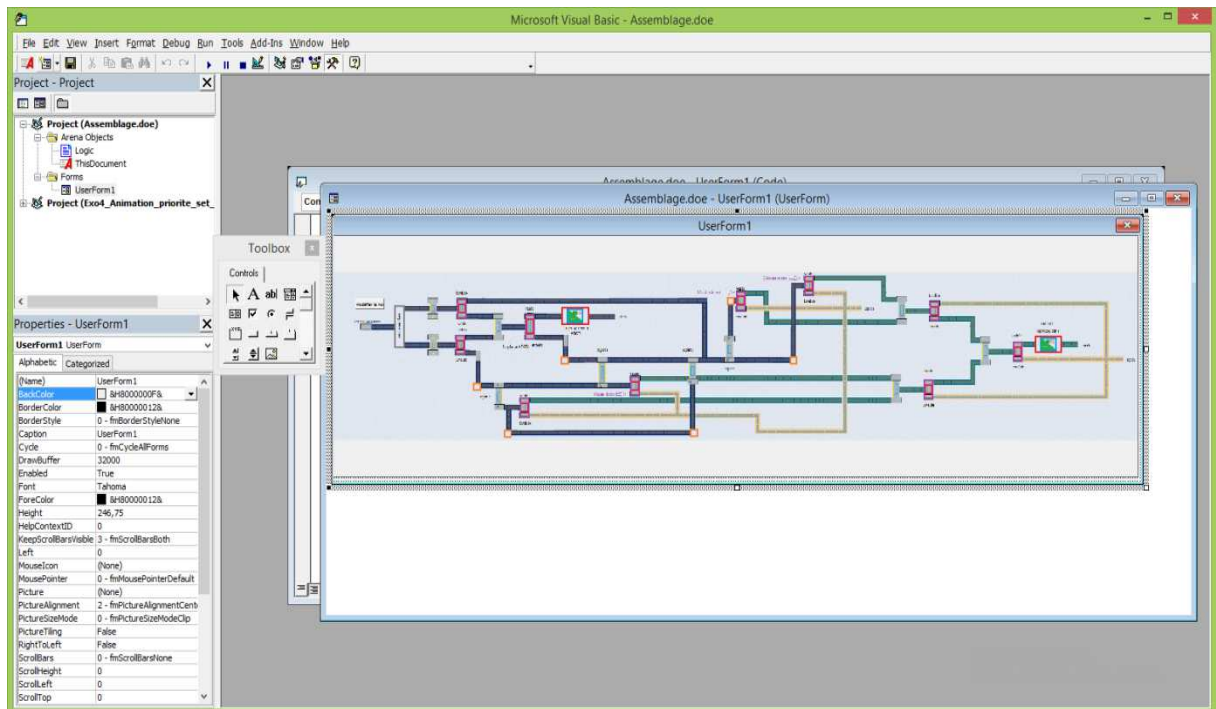


Figure 37: interface avec Visual Basic.

III.10.Conclusion :

Dans ce chapitre, nous avons présenté l'implémentation de notre application où nous avons donné une description détaillée à travers des d'écrans capturées représentant l'interface de l'application.

III. Simulation d'Assemblage des pièces électroniques:

III.1. Introduction :

Après avoir présenté dans le chapitre II la présentation des systèmes de production basée sur les systèmes à événements discrets, nous donnons dans ce chapitre la modélisation de notre modèle et quelques détails sur la réalisation de l'application et les outils utilisés.

Nous expliquons le principe de fonctionnement de l'application via des différents extraits d'interfaces implémentées.

III.2. Pour quoi on a choisi Arena-simulation ? :

C'est un outil facile à maîtriser et l'utiliser [20] :

- Plus de 5000 objets d'animation complexes sont incluses dans la bibliothèque de l'animation Arena.
- Animation peut également être créée par l'utilisateur. Clip Art, des bitmaps, des dessins AutoCad et beaucoup d'autres.
- La compatibilité des données : données d'importation et d'exportation de / vers un fichier de Les types de fichiers suivants: Excel, Access, XML, texte,....
- Script Visual Basic : Contrairement à d'autres outils qui utilisent des langages propriétaires de script, Arena utilise un éditeur standard VBA (inclus) et le modèle objet Arena pour construire des interfaces utilisateur personnalisées et sur mesure interfaces de données à des modèles Arena.
- Visual Basic Automation : toutes les fonctions Arena peuvent être automatisées avec la programmation Visual Basic.

III.3. Utilisation d'ARENA :

L'installation d'**ARENA** est une installation simple sauf que la version complète de ce logiciel est payante.

Dans notre projet on a utilisé la version **STUDENT** est une version gratuite destiné à un usage scolaire seulement. Cette version gratuite d'Arena :

- Offre une fonctionnalité complète.
- Le modèle limité de la taille et peut être copié sur un autre PC.
- Il n'y a aucune limite de temps.

III. Simulation d'Assemblage des pièces électroniques

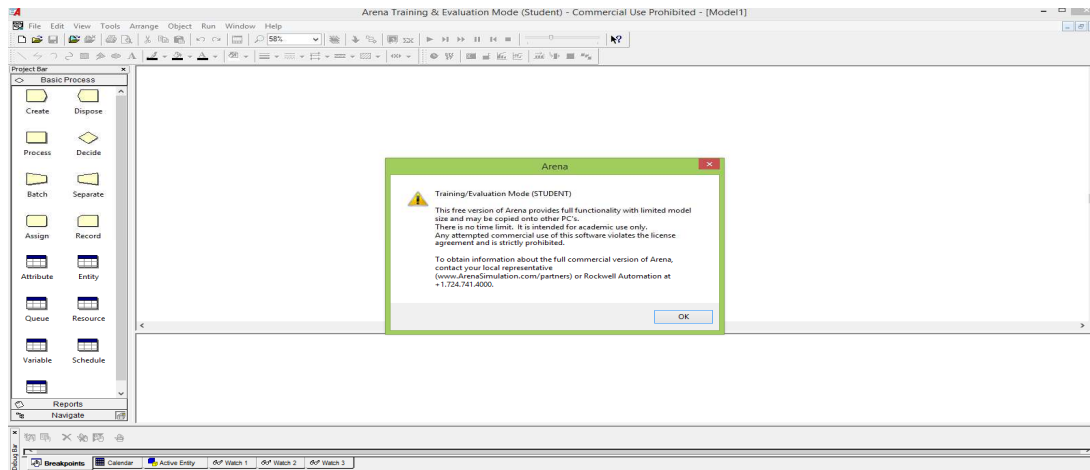


Figure III.1 : Image interface d'ARENA.

III.3.1. Les modules à configurer sous Arena :

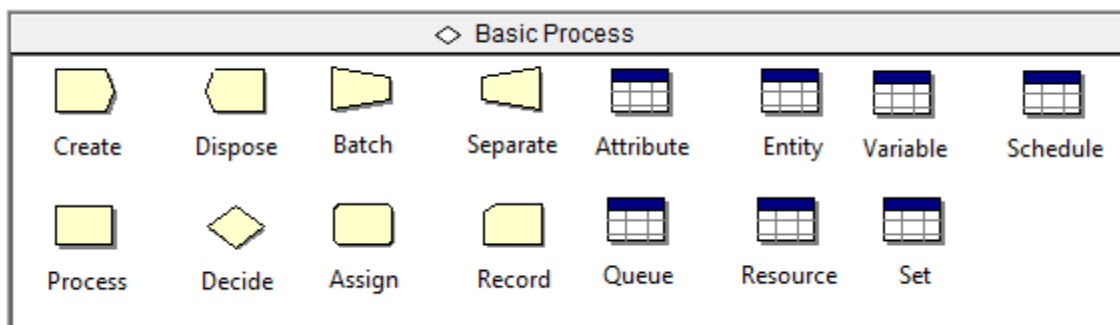


Figure III.2: Basic Process d'ARENA.

➤ **Create:** Ce bloc est considéré comme le module de départ des entités.

Les arrivées dans le système (pièce, machine ...).

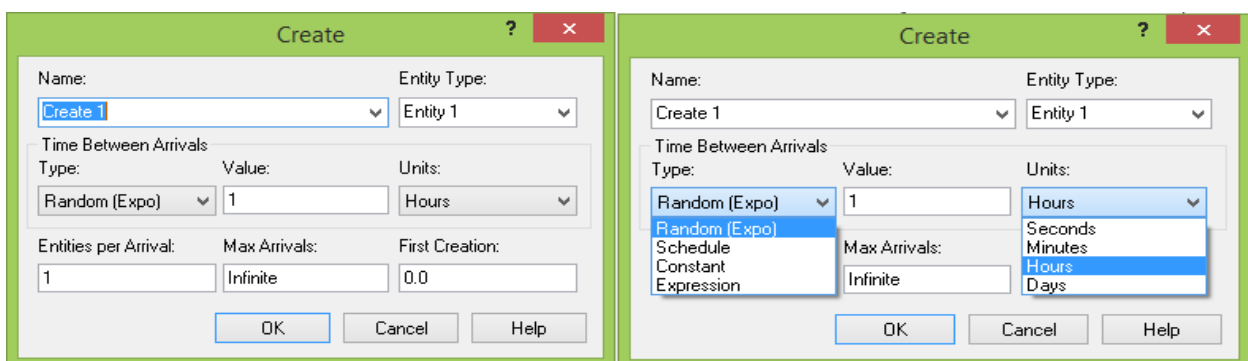


Figure III.3: Les paramètres de Create.

- ✓ **Type:** il y a plusieurs lois, chacun a une méthodologie différente de travail.
- ✓ **Value:** nombre des arrivées par S/M/H/D.
- ✓ **Units:** unité de temps.
- ✓ **Entities per Arrival:** N arrivées par unité de temps.

- ✓ **Max Arrivals:** maximum des arrivées par unité de temps.
- ✓ **First Creation:** le temps de début des arrivées.

➤ **Dispose:**

Les sorties du système (soit des pièces, machines,...).

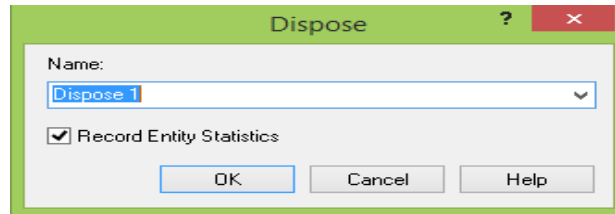


Figure III.4 : Les paramètres de Dispose.

➤ **Process:**

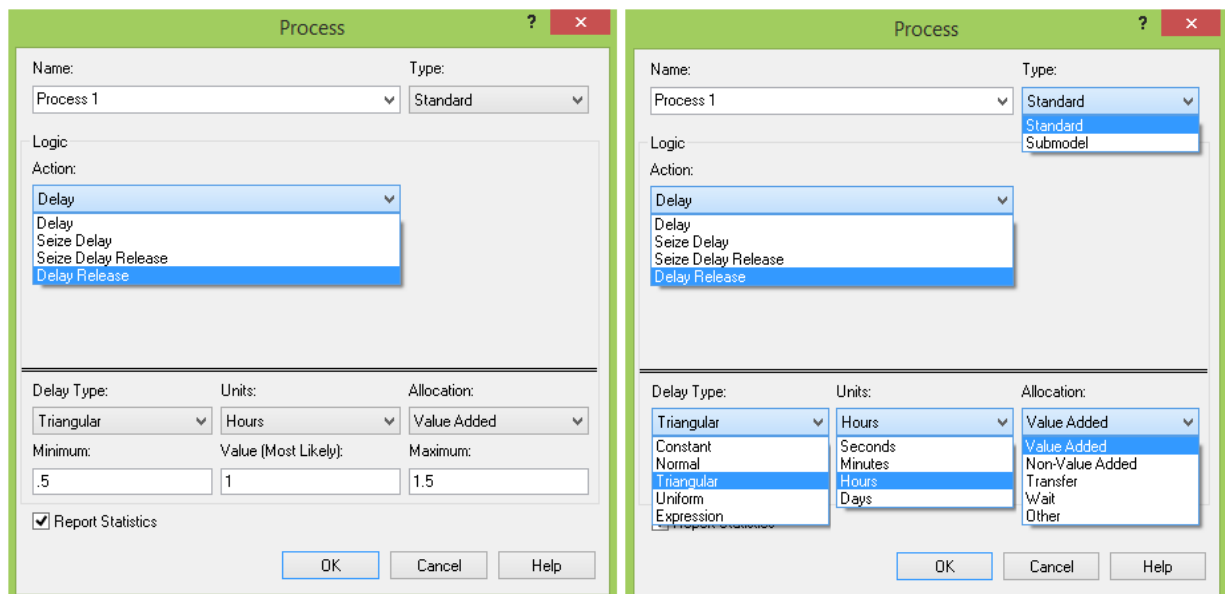


Figure III.5 : Les paramètres de Process.

➤ **Action:**

- **Delay:** L'entité doit être séjournée dans le bloc process pendant un temps spécifié.
- **Seize Delay:** File d'attente en amont du bloc Delay, et Nécessité de joindre une ressource au processus
- **Seize Delay Release:** Comme Seize Delay, mais une entité libère des unités de ressources après Delay (ce que nous voulons dans ce modèle).
- **Delay Release:** Suppose entité à déjà saisi des unités de ressources à un autre module amont, maintenant retards et de presse unités de ressources.
- **Delay type:** loi utilisée.
- **Units:** unité de temps.

➤ Decide:

Décide si l'arrivée est vraie ou fausse.

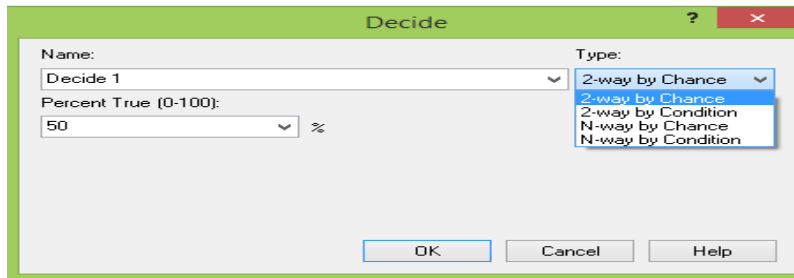


Figure III.6 : Les paramètres de Decide.

➤ Type:

- **2-way by Chance:** soit vrai ou faux avec pourcentage de succès.
- **2-way by Condition:** soit vrai ou faux avec condition.
- **N-way by Chance:** plusieurs choix avec pourcentage de succès.
- **N-way by Condition:** plusieurs choix avec condition.

➤ Batch:

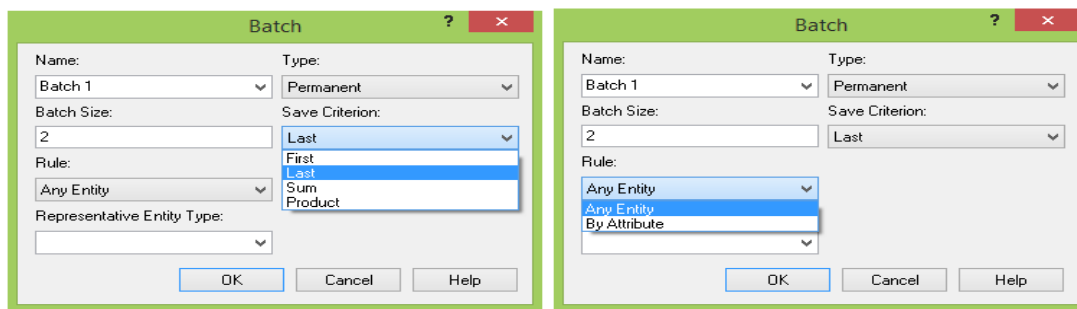


Figure III.7 : Les paramètres de Bach.

- **Batch Size:** divisé l'entité à 2 par exemple.
- **Rule:** attribut ou entité.
- **Save Criterion:** produit.

➤ Separate:

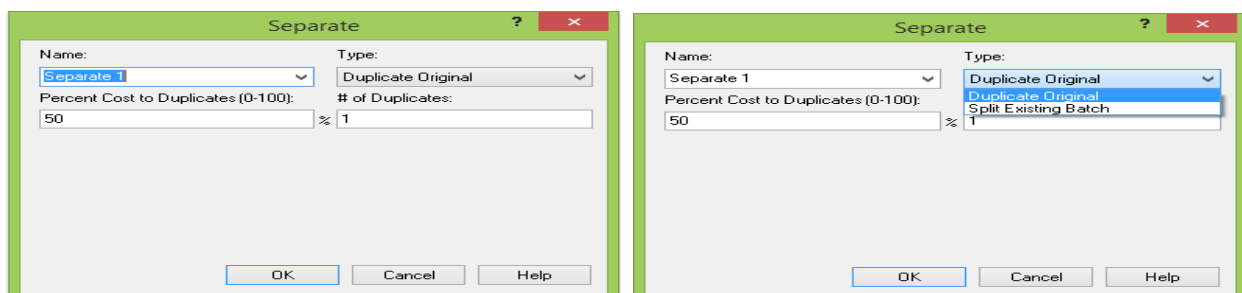


Figure III.8 : Les paramètres de Separate.

➤ **Type:**

- **Duplicate Original:** Revient à la forme originale.
- **Split Existing Batch:** faire le contraire d'un Batch existe.
- **Percent Cost to Duplicates (0-100):** le pourcentage pour dupliquer une entité.
- **# of Duplicates:** nombre de duplication.

➤ **Assign & Record:**

Utilisé pour calculer le temps écoulé dans un système (temps d'attente, la durée de service, ...).

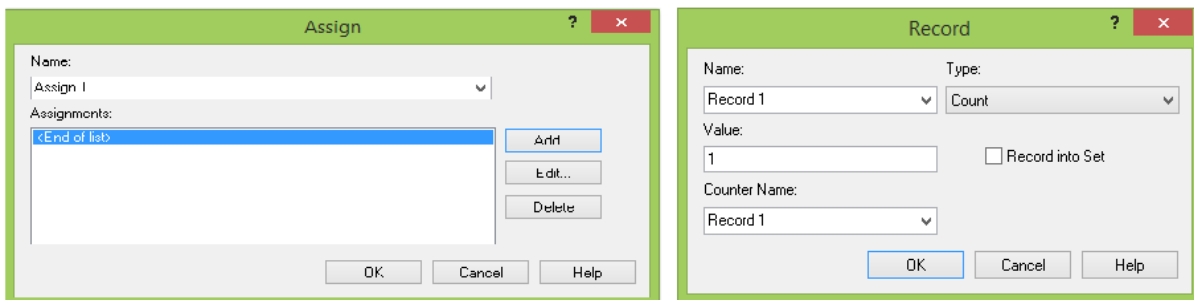


Figure III.9 : Les paramètres d'Assign & Record.

III.4. La modélisation et la configuration de notre projet :

Notre cas d'étude : Assemblage électronique

III.4.1. Présentation :

- Nous voulons regrouper quatre pièces A, B, C, D pour construire une plaque électronique
 1. Le système de production marche de 8:00 à 17:00, il prend une pose de un heure de 12:00 à 13:00 dans cette période les machines ne marche pas même si il y a des produits dans le système.
 2. les arrivées(les pièces A, B, C et D) n'arrive pas dans la période de une heure et aussi l'arrivée des pièces sa divers avec le temps c'est à dire (le rendement d'arrivée est plus fort dans le matin de 8:00 à 11:00 et diminuer de 11:00 à 12:00 et faible de 13:00 à 17:00.
 3. Les pièces orientes vers deux agents qui faire l'assemblage est la soudure de A avec B et C avec D.
 4. Après l'assemblage de AB et CD ces derniers pilotés vers deux machines marchent indépendamment, ces machines permettant de tester la validité des AB et CD.
 5. Si les plaques AB et CD sont valide alors elle orientes vers l'agent qui faire l'assemblage et la soudure de AB avec CD.
 6. Après l'assemblage on obtient une plaque ABCD, cette plaque piloter vers la machine qui faire le test, si la plaque est valide elle orient vers la machine de regroupement cette machine mettre dans chaque boite 6 plaques électroniques.

7. Si les plaques AB et CD sont invalides alors elles sont orientées vers l'agent qui fait le démontage des pièces, il y a deux agents (l'agent qui fait le démontage de AB et l'autre qui démonte la plaque CD).
8. Après la démontage de AB et CD on obtiens quatre pièces A,B,C et D chaque pièce vas orientes vers une machine de test .
9. Si les pièces sont invalides alors on les rejette, sinon on reproduise les pièces valides.
10. Les pièces A et B sont orientées vers l'agent qui fait l'assemblage de AB2.
11. Les pièces C et D sont orientées vers l'agent qui fait l'assemblage CD2.
12. Après l'assemblage de AB et CD on orientant vers les deux machines de test.
13. Si AB2 et CD2 sont valides, elles seront pilotées vers l'agent qui fait l'assemblage.
14. Si les plaques AB2 ou CD2 sont invalides alors on les rejette.
15. Après l'assemblage de ABCD 2, nous orientons les plaques vers la machines de test, si la plaque est valide nous allons mettre dans chaque boite 6 plaques, sinon on rejet la plaque.

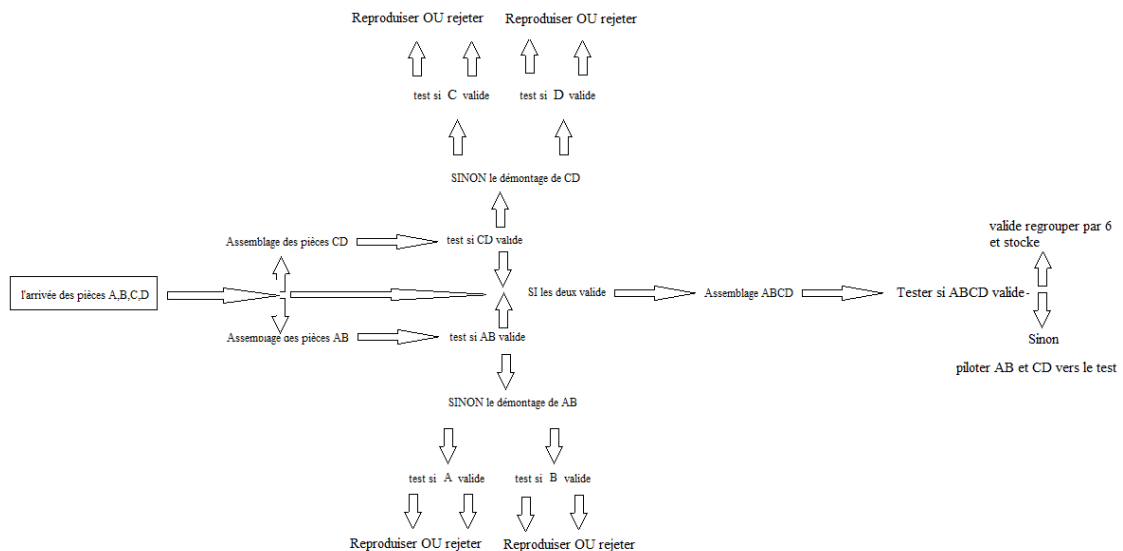


Figure III.10 : Assemblage de ABCD.

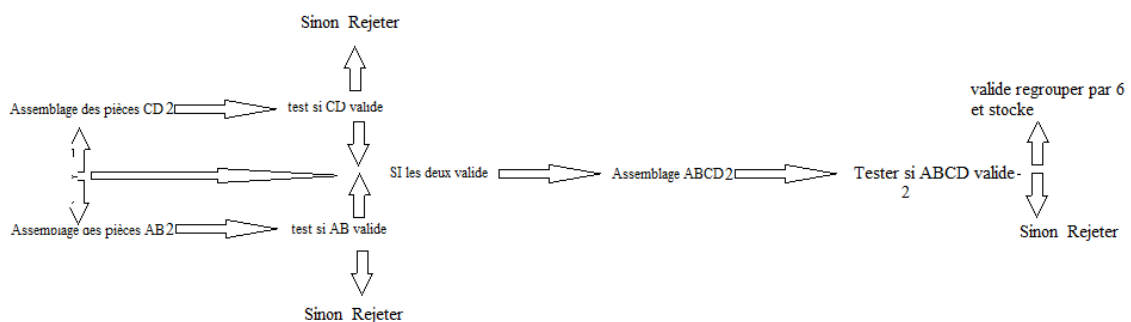


Figure III.11 : Assemblage ABCD2.

III.4.2. Implémentation :

- Nous précédon dans la modélisation de notre cas d'étude par les étapes suivantes:
 - L'arrivée des pièces suit un ordre : de 8:00 à 12 :00 et de 13:00 à 17:00

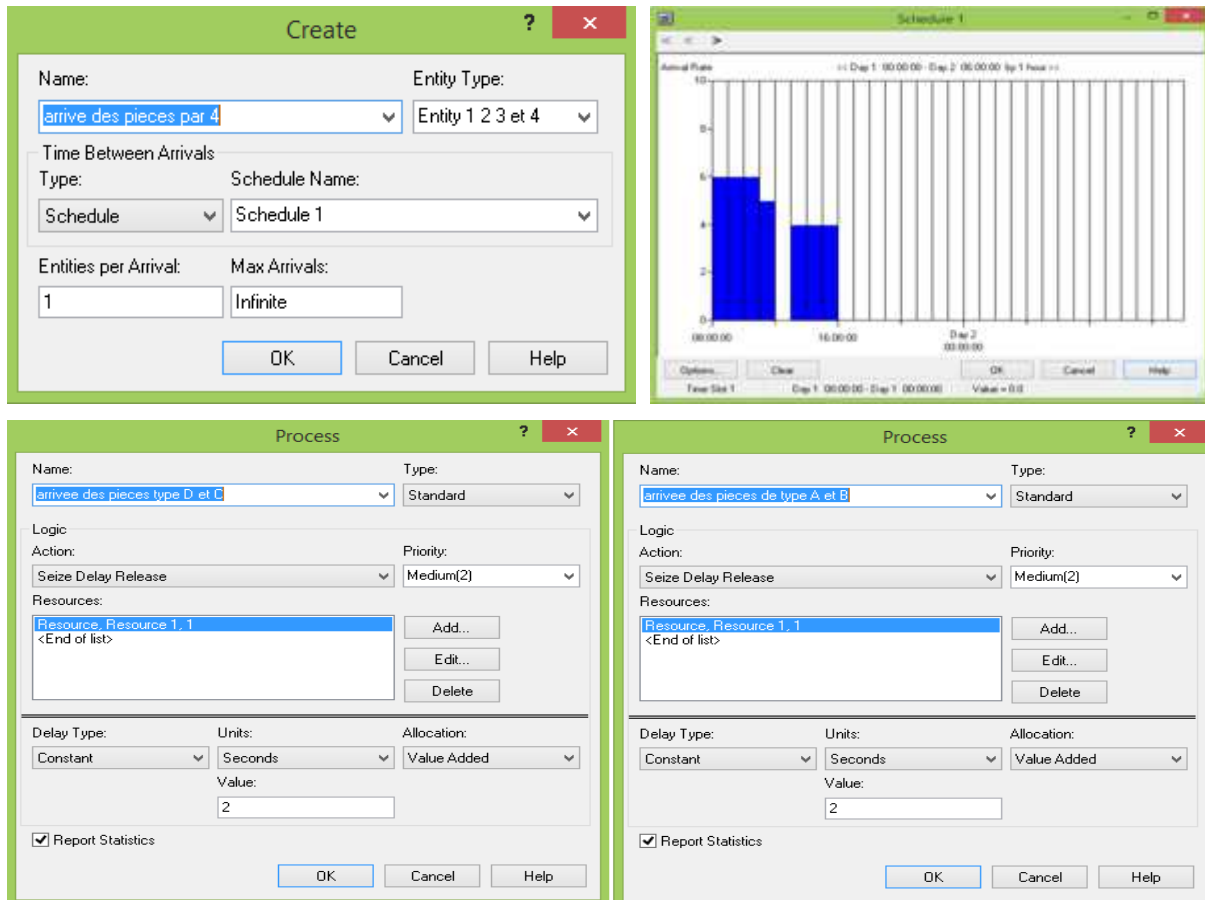


Figure III.12 : Configuration de Create & Process (Les arrives A, B, C, D).

- Nous mettons deux Process <Assemblage AB> et < Assemblage CD >
 Assemblage AB permet de souder la pièce A avec B on suit la loi Triangulaire (20, 25,30) par secondes.
 L'Assemblage CD permet de souder la pièce C avec D on suit la loi Triangulaire (20, 25,30) par secondes.



Figure III.13 : Assemblage de A avec B et C avec D.

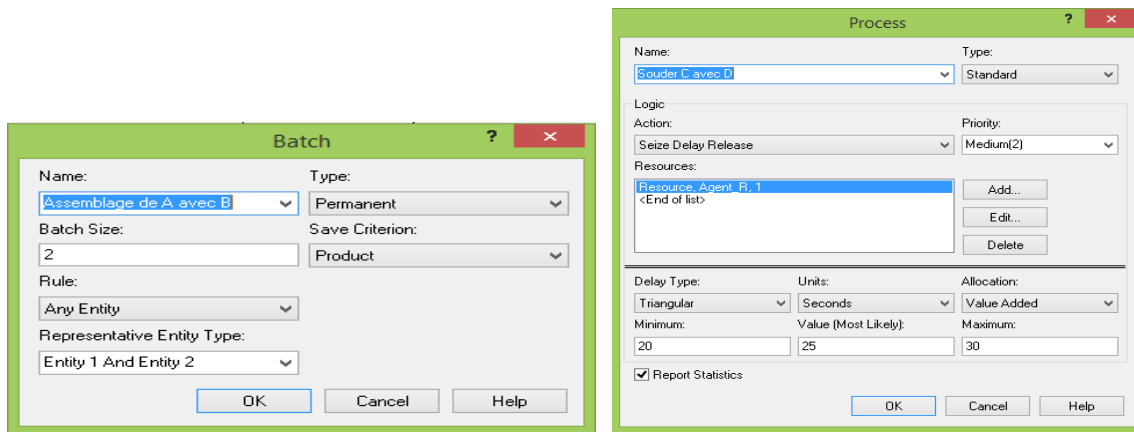


Figure III.14 : Configuration de Batch & Configuration de Process.

3. Il y a deux machines de test, ce dernier teste la validité de la plaque AB et CD, chaque machine travaille indépendamment de l'autre, la validité ou l'invalidité d'une plaque suit une method 2-way by chance le pourcentage de la validité d'une plaque électronique AB ou CD égale à 80%, la machines de test prend 5 secondes.

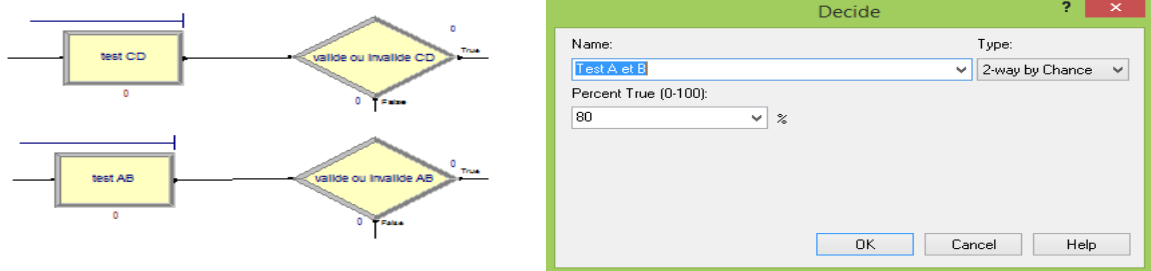


Figure III.15 : Machine de test AB et CD & Configuration de Decide.

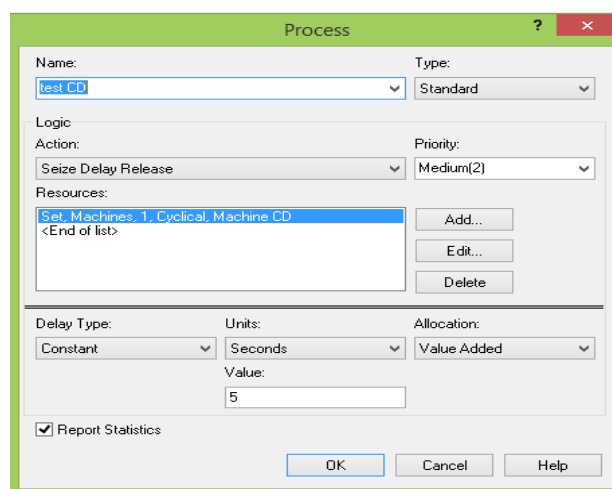


Figure III.16 : Configuration de Process.

4. L'assemblage ABCD permet de souder la plaque AB avec CD sauf si les deux plaques sont valides, la durée de l'assemblage suit une loi Triangulaire (20, 25,30) par secondes.

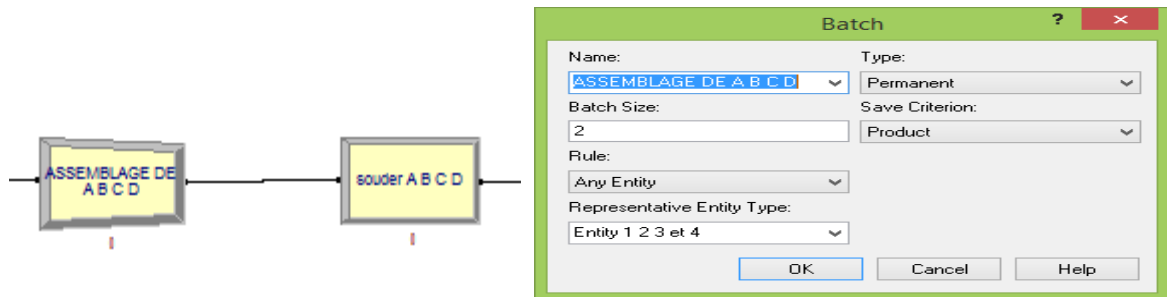


Figure III.17 : Assemblage de ABCD & configuration de Batch.

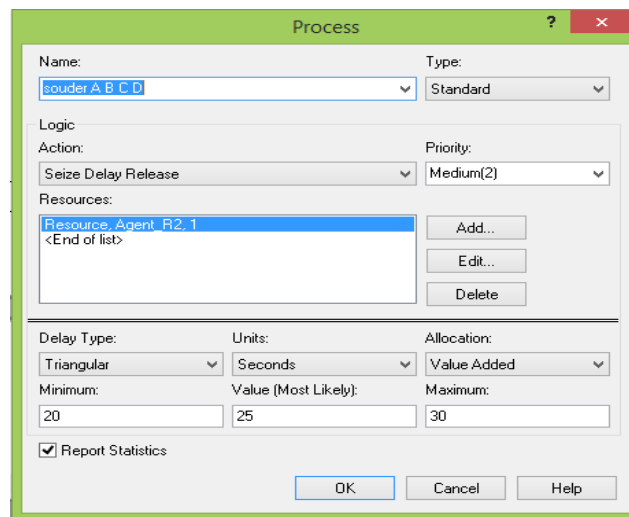


Figure III.18 : Configuration de Process.

5. Nous mettons une machine de test, cette machine teste la validité de la plaque ABCD, la validité ou l'invalidité d'une plaque suit une method 2-way by chance le pourcentage de la validité d'une plaque électronique égale à 80%, la machine prendre 5 secondes pour tester si la plaque est valide.

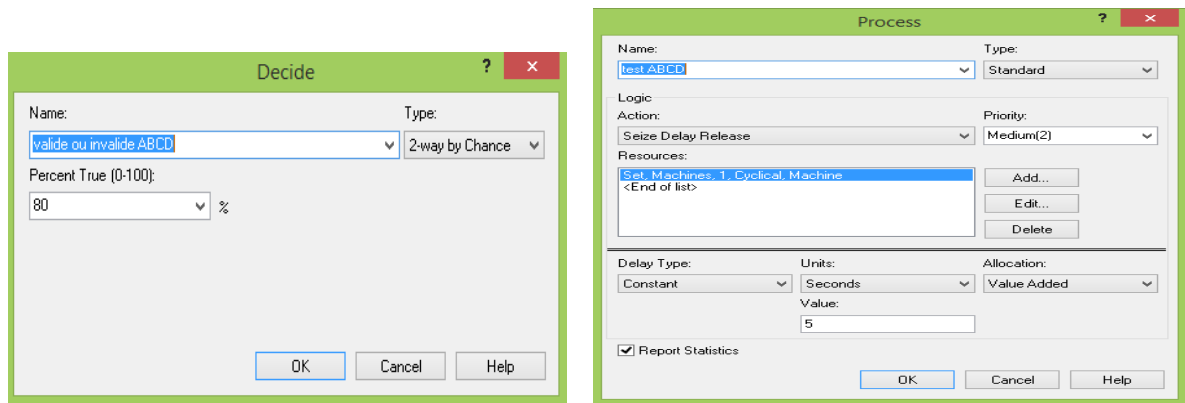


Figure III.19 : Configuration de Decide & Process.

6. Si la plaque est valide, nous l'orientons vers la machine de regroupement (chaque boîte contient 6 plaques électroniques) et on oriente la boîte vers le stock.

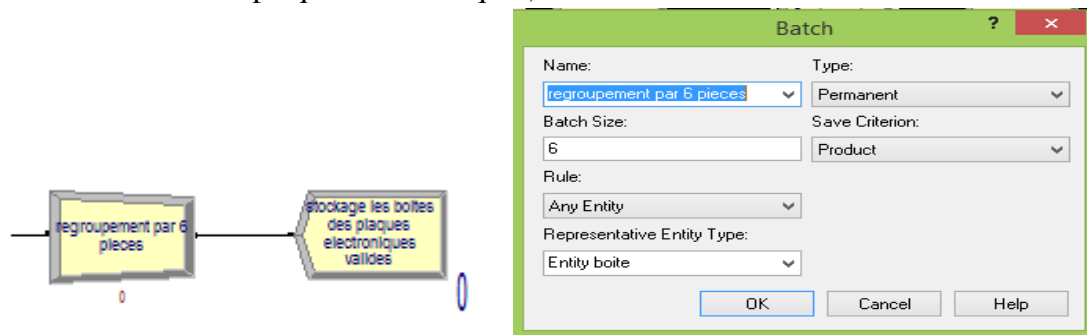


Figure III.20 : Regroupement des plaques & Configuration de Batch.

7. Si la plaque ABCD est invalide alors nous orientons les plaques invalides vers les agents de séparation. Premièrement, nous séparons la pièce A de la plaque ABCD et l'envoyons vers la machine de test (test que les pièces A), la validité ou l'invalidité d'une pièce suit une method 2-way by chance, le pourcentage de la validité de cette pièce égale à 80%, la machine de test prend 5 seconde, si la pièce est valide on passe vers la deuxième étape sinon on rejette.



Figure III.21 : Séparer la pièce A de la plaque ABCD. & test de la pièce A.

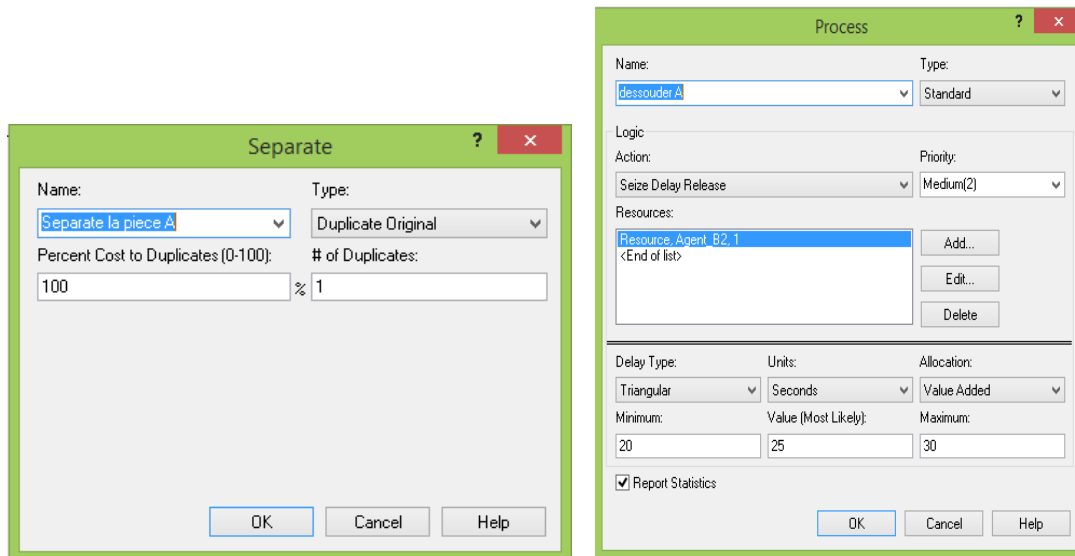


Figure III.22 : Configuration de Process & Configuration de Separate.

8. Deuxièmement, on sépare la pièce B de la plaque BCD et on l'oriente vers la machine de test (test que les pièces B), On suit les mêmes étapes de la pièce A.

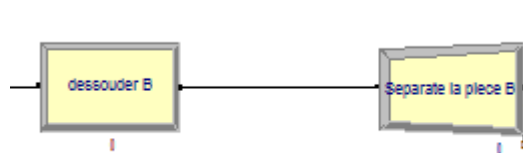


Figure III.23 : Dessouder B.

9. Troisièmement, on oriente la plaque CD vers l'agent responsable du processus de séparation.



Figure III.24 : Dessouder C et D.

10. Après la séparation on oriente la pièce C et la pièce D vers deux machines de test différentes, chaque machine teste la validité ou l'invalidité de pièce, la pièce valide passe vers la deuxième étape et la pièce invalide on la rejette.



Figure III.25 : Test si les pièces sont valides.

11. La deuxième étape permet de reproduire les pièces valides A, B, C, D

On oriente les pièces vers deux agents différents pour obtenir AB2 et CD2

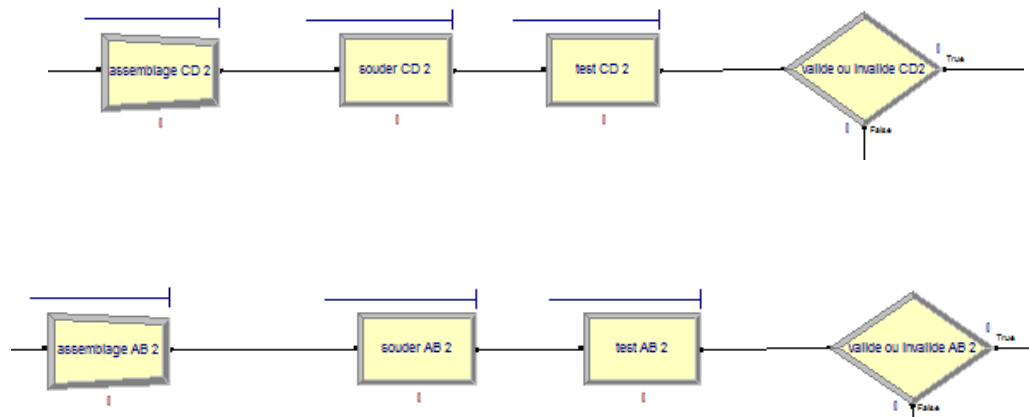


Figure III.26 : Assemblage & test.

Si les AB2 et CD2 sont valides alors on va piloter vers l'agent qui fait l'assemblage AB2 avec CD2, sinon on rejette.

12. Après l'obtention des plaques ABCD2 nous les orientons vers la machine de test, si la plaque est valide on regroupe par 6 plaques dans chaque boîte, sinon nous rejetons les plaques invalides.

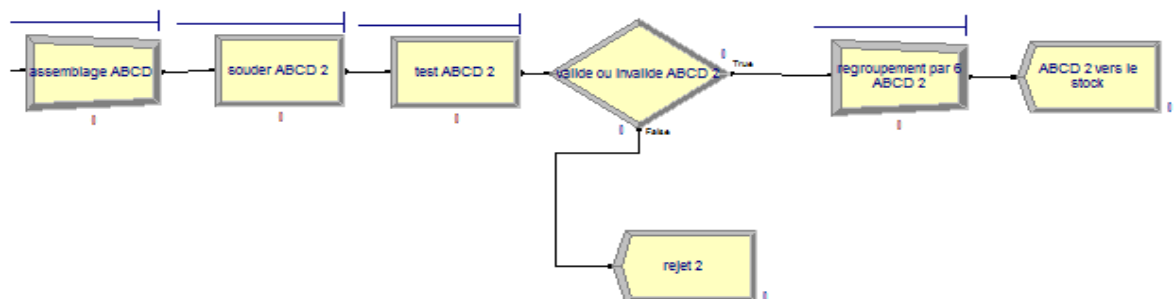


Figure III.27 : Assemblage & test & regroupement.

III.4.3. Le Model Générale ARENA:

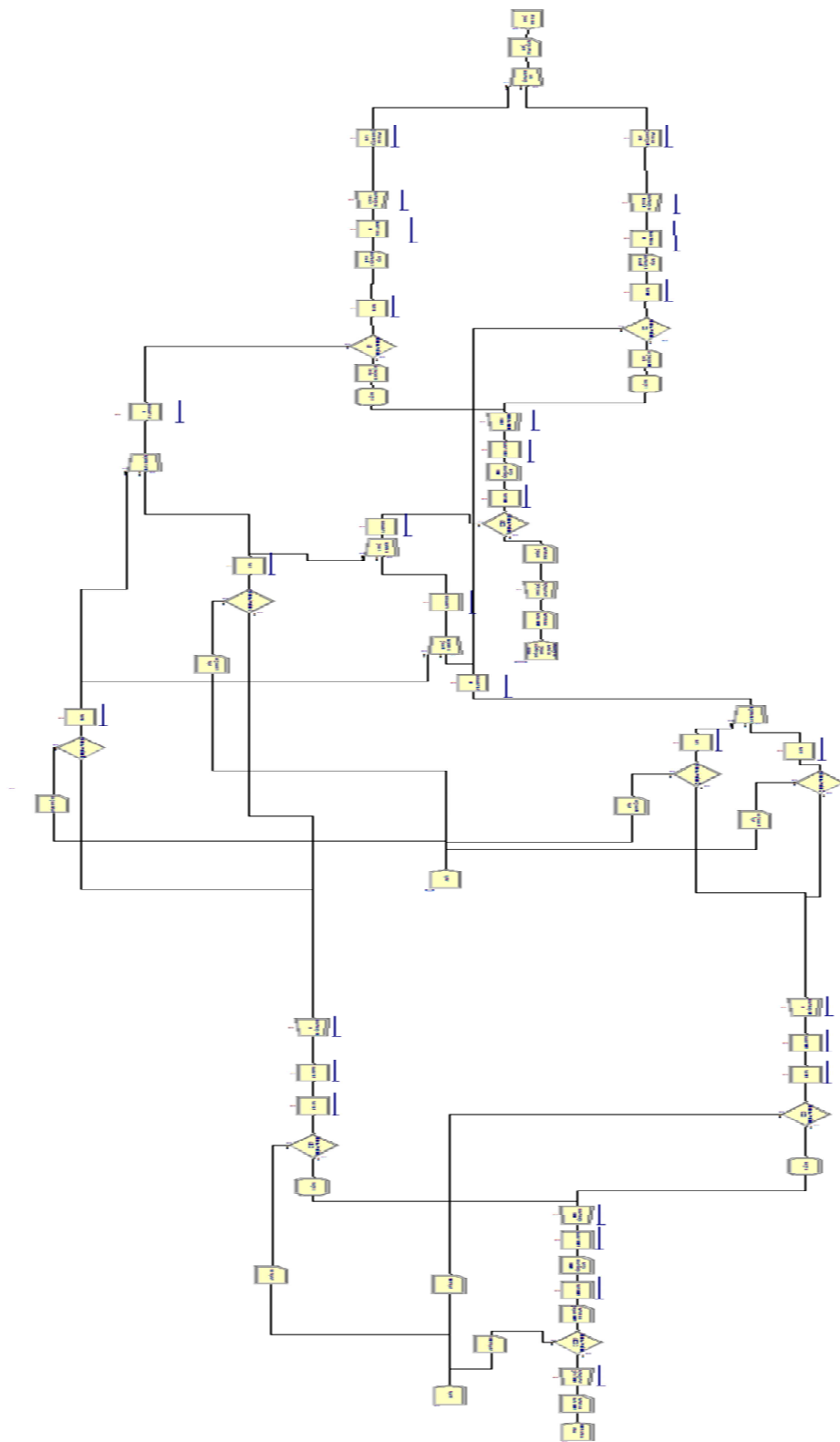


Figure III.28 : Le modèle général.

III.5. Animation 2D:

III.5.1.Les Agents:

Nous avons utilisé les agents pour l'animation de l'assemblage et le démontage des pièces.

➤ Les Agents rouges: Assemblage des pièces

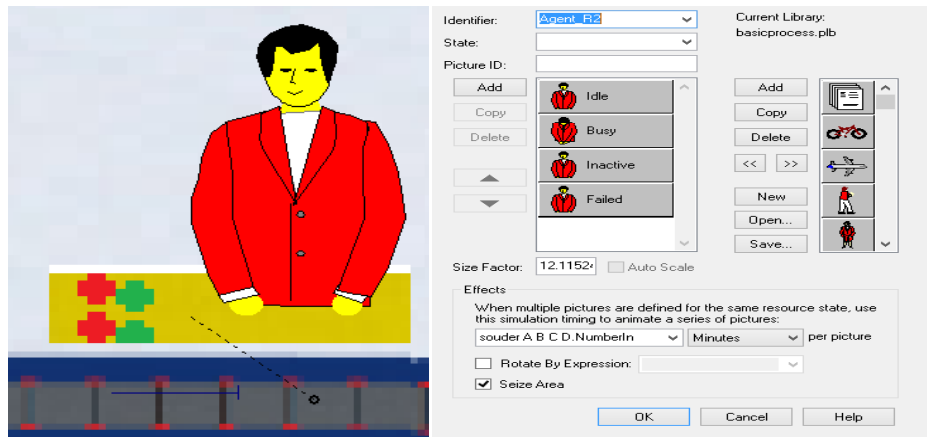


Figure III.29 : Agent Rouge.

Il représente le : Batch plus Process (c'est-à-dire l'assemblage et la soudure des pièces).

➤ Les Agents bleus: démontage des pièces

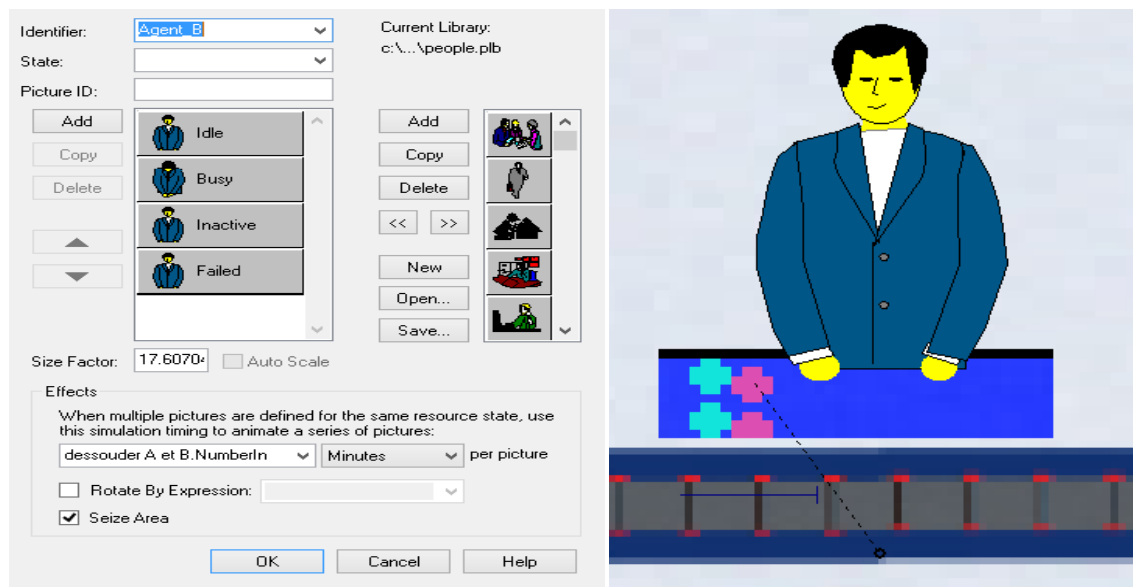


Figure III.30 : Agent bleu.

Il représente le : Separate plus Process (c'est-à-dire dessouder et séparer les pièces).

III.5.2. Les machines de test : test la validité des pièces et des plaques

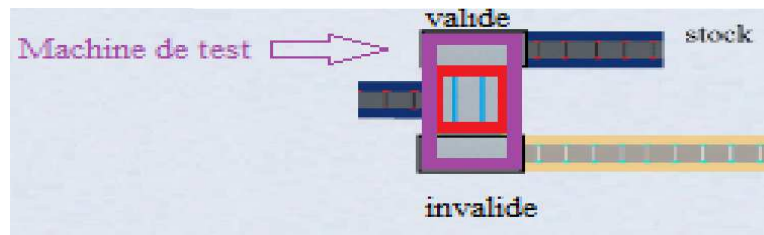


Figure III.31 : Machine de test.

Elle représente le Test.

III.5.3. Les agents verts : assemblage AB2 et CD2 et ABCD2

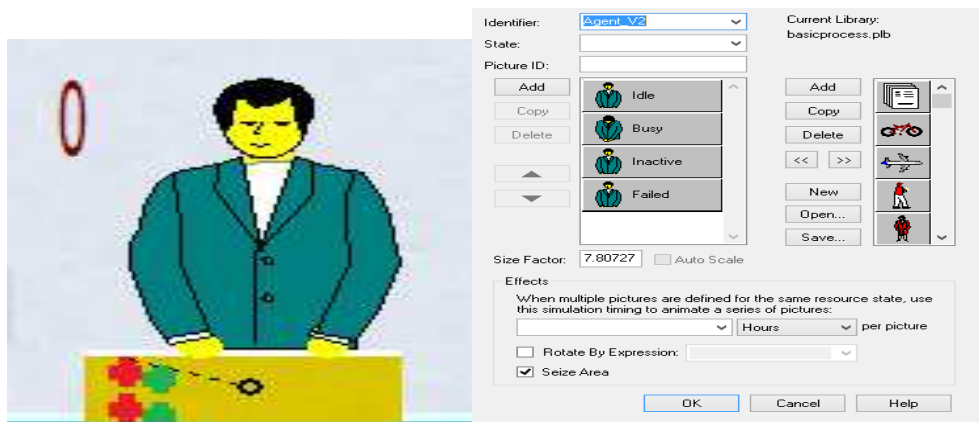


Figure III.32 : Agent vert.

Il représente le : Batch plus Process (c'est-à-dire l'assemblage et la soudure des pièces).

III.5.4. Schéma de l'atelier d'assemblage électronique :

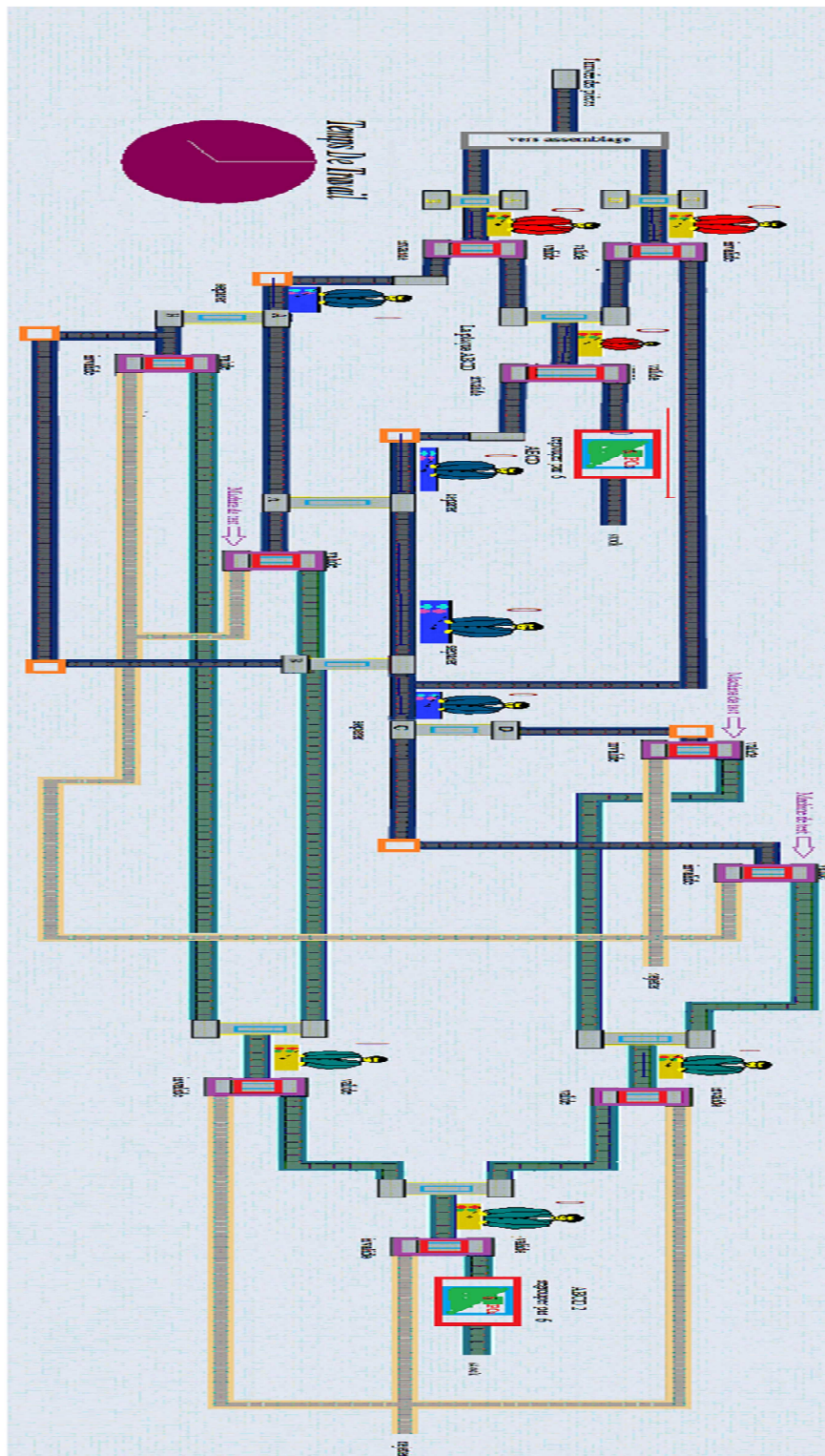


Figure III.33 : Schéma général de l'atelier.

III.6. Les résultats de simulation :

Pour avoir des résultats sur la simulation, nous lisons le rapport de simulation qui se génère automatiquement à la fin de la simulation.

Chaque rapport représente les données d'un seul scénario de simulation.

Il y a des données qui sont par défaut dans le rapport comme par exemple le temps d'attente moyen dans une file d'attente d'un process donné. D'autres informations sont à créer par l'utilisateur comme le temps moyen passé par une entité dans le système.

Record comme moyenne pour : calculer le temps moyen d'assemblage, intervalle de temps (début et fin d'une opération dans les systèmes), nombre des unités passer dans le système....

- **Calculer le temps moyen :** temps moyenne d'assemblage C avec D, par exemple :

The screenshot shows the 'Record' dialog box with the following configuration: Name: temps assemblage C avec D, Type: Time Interval, Attribute Name: Temps, Tally Name: temps assemblage C avec D, and the 'Record into Set' checkbox is unchecked. Buttons for OK, Cancel, and Help are at the bottom.

Figure III.34 : Configuration de Record.

- **Compter le nombre des plaques électroniques :** On utilise **Record**

The screenshot shows the 'Record' dialog box with the following configuration: Name: Nombre des pieces, Type: Count, Value: 1, Counter Name: Nombre des pieces, and the 'Record into Set' checkbox is unchecked. Buttons for OK, Cancel, and Help are at the bottom.

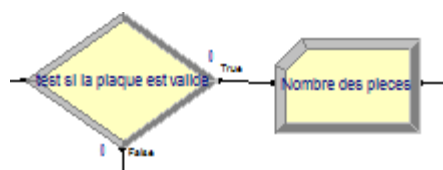
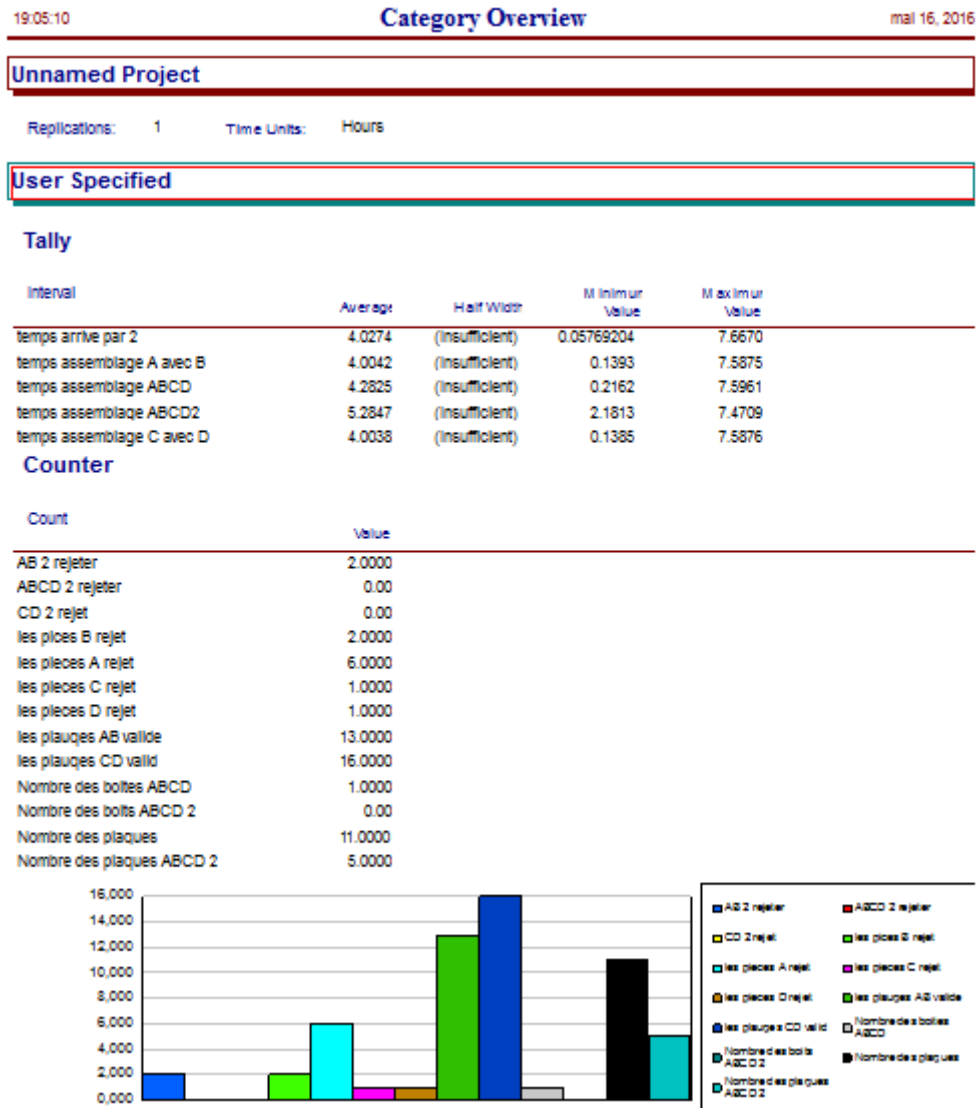


Figure III.35 : Configuration de Record (count).

III.7. Rapport de simulation :



Après les résultats obtenus, si les dirigeants de l'entreprise sont satisfaits alors le scénario appliqué est le scénario requis.

Sinon, nous allons changer les facteurs d'influences, par exemple :

Nous augmentons le nombre des pièces arrivées dans le temps.

Nous augmentons la qualité des pièces pour diminuer le nombre des pièces à rejeter (nous changeons la possibilité d'avoir les pièces valides vers 90% ou plus).

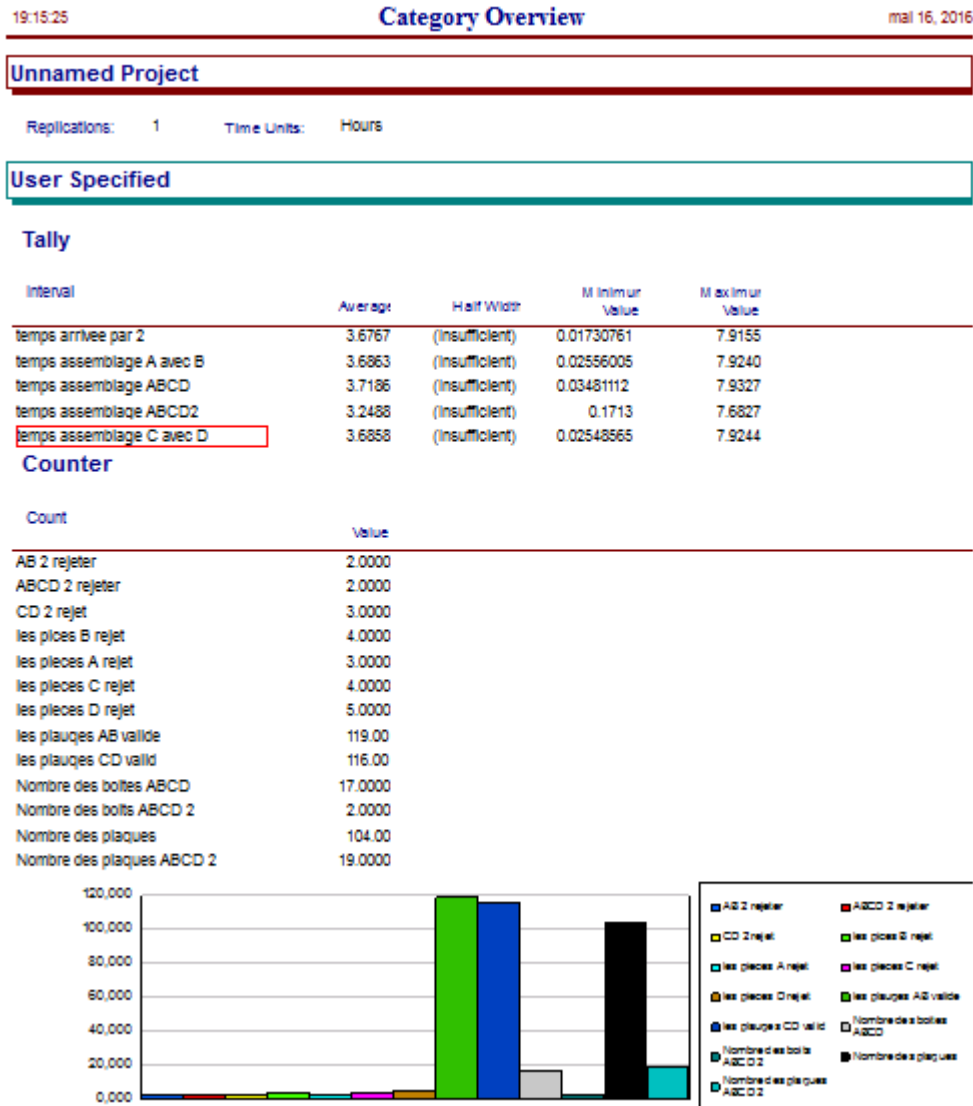
Nous avons diminué le temps d'assemblage des pièces.

III.8. Application du deuxième scénario :

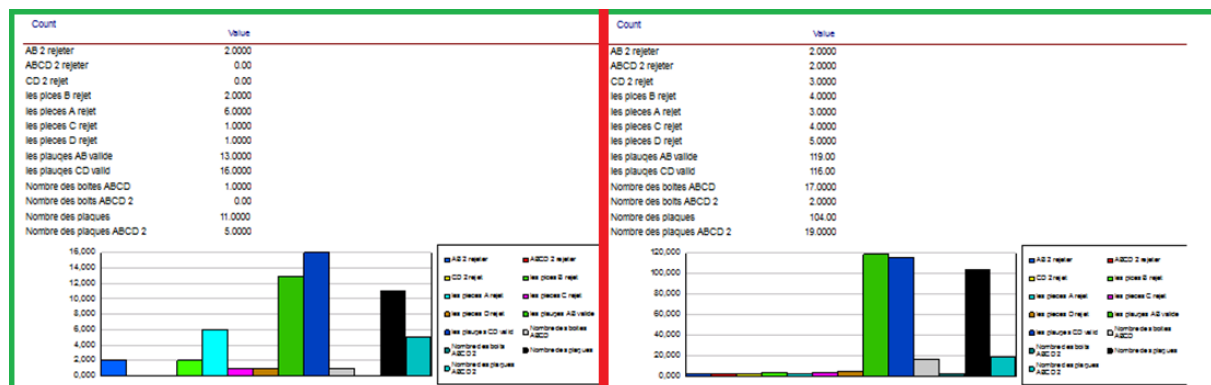
Le deuxième scénario est le même schéma que la première appliquée sauf-que on va changer le :

1. Augmentation du nombre des arrivées par le temps.
2. la possibilité d'avoir les pièces valide vers 90%.

III.8.1. Le résultat de simulation :



Nous comparons les deux scénarios :



Nous trouvons que le deuxième scénario est le mieux adapté avec les besoins d'une entreprise d'assemblage des pièces électroniques.

Après le test et l'évaluation de performance de notre modèle on peut maintenant réaliser notre projet (c'est-à-dire l'implémentation de ce projet dans la vie réelle).

III.9. Interface Visual Basic :

Nous voulons créer une interface qui permet à l'utilisateur de gérer et modifier quelques paramètres de production.

Le langage de programmation qui permet réaliser cette fonction est le Visual Basic.

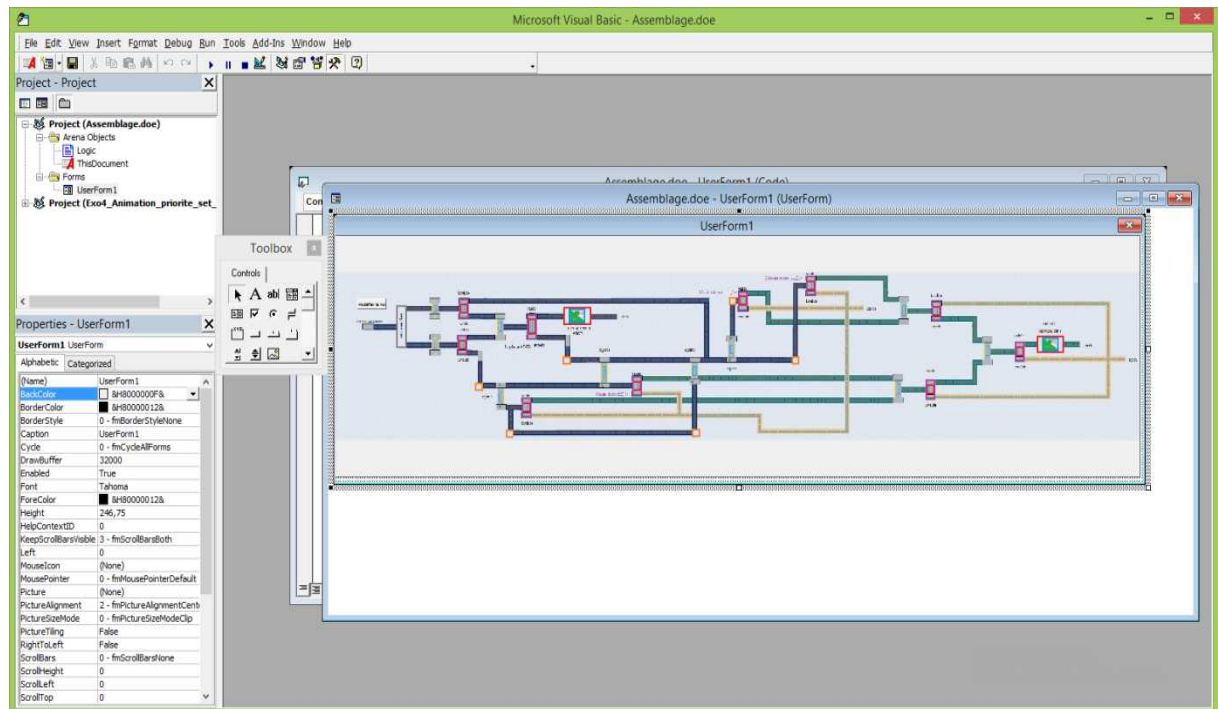


Figure III.36 : Intégration de Visual Basic.

III.10.Conclusion :

Dans ce chapitre, nous avons présenté l'implémentation de notre application où nous avons donné une description détaillée à travers des d'écrans capturées représentant l'interface de l'application.

À la fin de ce chapitre nous avons obtenu deux résultats de deux scénarios et nous avons fait la comparaison entre les deux. Nous avons choisi donc le meilleur scénario par rapport à son rendement.

Conclusion générale

La simulation sur ordinateur est un outil très utile pour bien comprendre le principe de fonctionnement d'un système. Elle aide à observer et à comprendre le fonctionnement en temps réel et à analyser plusieurs scénarios de fonctionnement.

Cependant, la simulation figurait parmi les techniques les plus utilisées à travers le monde. Cependant plusieurs obstacles n'ont pas favorisé une plus large utilisation de la simulation des systèmes de production.

Après l'évaluation de plusieurs scénarios chaque scénario a des conditions moins ou plus d'un autre comme le temps d'assemblage, la quantité des arrivées par le temps, ce dernier est stabiliser sur un modèle d'assemblage des pièces électroniques compatible avec notre demande, la comparaisons entre les scénarios fait à partir de la comparaison de le rendement de production, soit la production avec un haut rendement .

Dans notre projet nous avons déduit après la comparaison entre les scénarios proposés que le deuxième scénario est le mieux adapté avec nos besoins.

ARENA-SIMULATION est un bon moyen pour la simulation d'un système à des évènements discrets, il nous aide à construire notre projet, et désirer notre objectif de simuler un système d'assemblage des pièces électroniques et l'évaluation de sa performance, modifier, tester et gérer le système. En plus, l'animation 2D permet d'avoir une meilleure vue sur le système.

Bibliographie :

- [1] - Professeur Laris : « SIMULATION DES SYSTEMES DE PRODUCTION », thèse de doctorat.
- [2] - Professeur Jean-Louis Boimond : « simulation systèmes de production réseaux de petri siman –arena », thèse de doctorat, http://laris.univ-angers.fr/_attachments/jean-louis-article-2/Cours%2520de%2520simulation.pdf?download=true.
- [3] – G.bontempi : « Projet de la modélisation et la simulation », assistant M.L.Lerman, Département d’informatique, septembre 17/2015.
- [4]- Helene. Guérin : « les variables aléatoires », Université rennes1.
- [5]- math u bordeaux : « mémoire de tables loi normale », Université René Descartes, Licence de Psychologie 3, chapitre 4, sdv 2012.
- [6] - Mathieu .Guidère : « les méthodes d’analyses», l’université de fribourg, http://www.unifr.ch/socsem/cours/compte_rendu/Les%20m%E9thodes%20d'analyse.Pdf.
- [7]- « Les réseaux de petri », université de caen/ basse- Normandie (U.F.R : sciences caen), master 1A mention EEA parcours AEII, 20 juin 2006.
- [8]- Professeur Jacques A. Ferland : « Modèles stochastiques, Modèle de file d’attente », recherche opérationnelle, Stanford University, 2000.
- [9]- « cours de la simulation », université de UCO – Campus virtuel.
- [10] - Gérard Dreyfus : « collection algorithmes dirigée par Gérard Dreyfus », simulation à événement discret, groupes Eyrolles (Gérard Dreyfus -Philippe lacomme -Alain Tanguy), 2007, http://www.eyrolles.com/Chapitres9782212119244/Chap1_Fleury.pdf .
- [11]- Jean Bérard : « Introduction aux probabilités et à la statistique », université de Lyon, <http://math.univ-lyon1.fr/~jberard/cours-www.pdf>.
- [12]- Stéphane goes : « Livre de Stéphane goes », management des organisations.
- [13]- blog agilea : « Historique et intérêts de la simulation de flux », inducteur de progrès expert en supply chain management, 20/03/2014.
- [14]- logiciel de simulation intuitif et rapide simul8 –logiciel , simcore expert en performance industrielle et logistique , Simcore est revendeur des solutions Simul8 en France.[en ligne]. <http://www.simcore.fr/simul8-logiciel.asp> **Accessible 08/02/2016.**
- [15]- logiciel de simulation intuitif et rapide automod –logiciel, simcore expert en performance industrielle et logistique, Simcore est revendeur exclusif du logiciel de simulation AutoMod en France et Bénélux. [en ligne]. <http://www.simcore.fr/automod-logiciel.asp> **Accessible le:08/02/2016.**

[16]- logiciel de simulation intuitif et rapide Enterprise Dynamics, simcore expert en performance industrielle et logistique, Simcore est revendeur du logiciel Enterprise Dynamics de Incontrol en France. [en ligne]. <http://www.simcore.fr/Enterprise-Dynamics-logiciel.asp>
Accessible le: 08/02/2016.

[17]- Xiaojun –ye (Ingénieur en Mécanique Industrielle): « La modélisation et Simulation des Systèmes de Production : une Approche Orientée—Objets », Thèse de doctorat, l'institut national des sciences appliquées de Lyon, 07 mai 2013.

[18]- Céline baranger, julien-mathiaud : « mémoire de méthode de Monte-Carlo », licence université Lyon, France, 2012/2013.

[19]- Georges Habchi : « conceptualisation et modélisation pour la simulation des systèmes de production », Document de Synthèse, université de Savoie, 05/12/2001.

[20]- Francisco E. Martinez Martinez: « Application of SIMAN ARENA Discrete Event Simulation Tool in the Operational Planning of a Rail System », Final Report, Department of Civil Engineering and Surveying, University of Puerto Rico Mayagüez Campus, N 841-94-3901, May 2002.

[21]- Brahim BELATTAR : « Modélisation & Simulation sur Ordinateur », Doctorat Nouvelle thèse en informatique (Diplômé de l'Université Claude BERNARD de Lyon I-France), Faculté des Sciences de l'Ingénieur Université de Batna ,2003/2004.