

**Treball Final de Carrera**

*Projectes de modelatge i simulació avançada  
de sistemes mitjançant l'entorn ARENA<sup>TM</sup>.  
(Volum I)*

Josep Aranda Arjona

**Enginyeria en Organització Industrial**

Director: Juli Ordeix Rigo

Vic, Gener 2011

**ÍNDEX :**

	Pàgina
<b>Resum de Treball Final de Carrera.....</b>	<b>7</b>
<b>Summary of Final Work of Career.....</b>	<b>8</b>
<b>1. OBJECTIUS.....</b>	<b>9</b>
<b>2. METODOLOGIA.....</b>	<b>10</b>
<b>3. INTRODUCCIÓ.....</b>	<b>12</b>
 <b><i>Primera Part. Conceptes Teòrics.</i></b>	
<b>4. RESUM DE L'ENTORN ARENA.....</b>	<b>15</b>
<b>4.1. Característiques bàsiques de la simulació amb ARENA.....</b>	<b>17</b>
<b>4.2. Limitacions de la versió per a estudiants.....</b>	<b>18</b>
<b>4.3. Compatibilitat amb Microsoft Office.....</b>	<b>19</b>
<b>4.4. Conceptes bàsics.....</b>	<b>20</b>
<b>4.5. Conceptes avançats.....</b>	<b>24</b>
<b>4.6. Definició de temps dins del sistema ARENA.....</b>	<b>30</b>
<b>4.6.1. Variables i atributs més utilitzats en l'entorn ARENA.....</b>	<b>33</b>
<b>4.6.2. Distribucions estadístiques incloses dins l'entorn ARENA.....</b>	<b>36</b>
<b>4.6.3. Arena Input Analyzer.....</b>	<b>38</b>
<b>4.6.4. Arena Expression Builder.....</b>	<b>46</b>
<b>4.7. Mòduls bàsics del sistema ARENA (Template Basic Process).....</b>	<b>51</b>
<b>4.8. Mòduls avançats del sistema ARENA (Template Advanced Process).....</b>	<b>55</b>
<b>4.9. Tractament de fallides.....</b>	<b>62</b>
<b>5. METODOLOGIA DE SIMULACIÓ.....</b>	<b>65</b>
<b>5.1. Creació d'entitats.....</b>	<b>66</b>
<b>5.2. Utilització dels recursos mitjançant entitats.....</b>	<b>67</b>

<b>5.3.</b>	<b>Distribució del flux de les entitats a través del model.....</b>	<b>68</b>
5.3.1.	Distribució del flux de les entitats a través del model unitàriament.....	68
5.3.2.	Distribució del flux de les entitats a través del model en grup.....	69
5.3.3.	Sincronització del flux de les entitats a través del model.....	71
5.3.4.	Distribució del flux de les entitats a través del model duplicant-les.....	72
5.3.5.	Retenció d'entitats i posterior alliberament mitjançant events.....	73
5.3.6.	Recerca i extracció d'entitats d'una cua mitjançant entitats de control.....	76
5.3.7.	Recollida d'un grup d'entitats d'una cua i transport a una altra part del model.	81
<b>5.4.</b>	<b>Anàlisi dels costos del sistema.....</b>	<b>82</b>
<b>5.5.</b>	<b>Animació de la simulació.....</b>	<b>84</b>
5.5.1.	Capes de presentació. Layers.....	84
5.5.2.	Models de simulació gràfica.....	84
5.5.3.	Animació i graficació mitjançant el Template Basic Process.....	86
5.5.3.1.	Animació d'entitats.....	86
5.5.3.1.1.	Modificació en l'animació de les entitats.....	86
5.5.3.2.	Animació d'un recurs.....	86
5.5.3.3.	Animació de les cues del model.....	86
5.5.3.4.	Animació del valor d'una variable en el model.....	88
5.5.3.5.	Addició de complements i figures al model.....	89
5.5.3.6.	Graficació de rellotges.....	89
5.5.3.7.	Graficació de dates.....	90
5.5.3.8.	Representació d'histogrames.....	91
5.5.3.9.	Gràfics dinàmics mitjançant la opció PLOT.....	91
5.5.4.	Animació mitjançant el Template Advanced Process.....	92
5.5.4.1.	Retenció d'entitats i posterior alliberament. Emmagatzenaments.....	92
5.5.5.	Animació mitjançant el Template Advanced Transfer.....	94

	Pàgina
5.5.5.1. Mòduls de transferència avançada.....	96
5.5.5.2. Funcionament d'un transport fix.....	106
5.5.5.3. Sol·licitud de transport lliure. Mitjançant Conveyors.....	107
5.5.5.4. Animació del camí entre estacions.....	110
5.5.5.5. Animació de l'element transportador.....	112
5.5.5.6. Animació de l'element conveyor.....	113
5.5.5.7. Modelatge i Animació del transport mitjançant vehicles filoguiat. AGV's.....	114
<b>5.6. Generació i anàlisi de reports.....</b>	<b>116</b>
5.6.1. Tractament de Reports.....	118
5.6.2. Recol·lecció d'estadístiques.....	119
5.6.2.1. Estadístiques recol·lectores del conteig d'entitats.....	120
5.6.2.2. Recol·lecció d'estadístiques de l'entitat.....	122
5.6.2.3. Estadístiques de l'interval de temps entre dos punts del model.....	124
5.6.2.4. Estadístiques del temps entre arribades.....	126
5.6.3. Estadístiques dins dels models de transport.....	126
5.6.3.1. Exemple de model de transport amb estadístiques.....	128
5.6.4. Emmagatzament d'estadístiques dins d'un arxiu extern.....	133
5.6.4.1. Emmagatzament mitjançant el mòdul ReadWrite.....	133
5.6.4.2. Emmagatzament mitjançant el mòdul STATISTIC.....	135
<b>5.7. Altres eines d'anàlisi i processos avançats.....</b>	<b>137</b>
5.7.1. ARENA Process Analyzer.....	139
5.7.2. ARENA Output Analyzyer.....	154
5.7.2.1. Generació de l'histograma.....	156
5.7.2.2. Interval de confiança per a la mitjana .....	159
5.7.2.3. Promig acumulat .....	160
5.7.2.4. Comparació de mitjanes .....	162

	Pàgina
5.7.2.5. Anàlisi de variància per a un sol factor (Anova).....	164
5.7.3. Interfície ARENA-EXCEL.....	166
5.7.4. ARENA Visual Basic for Applications (VBA).....	175

## ÍNDEX resumit de la segona part:

### *Segona Part. Exercicis Pràctics.*

<b>6. EXERCICIS PRÀCTICS.....</b>	<b>11</b>
<b>6.1. PRÀCTIQUES AMB CONCEPTES BÀSICS.....</b>	<b>11</b>
6.1.1. Entitats.....	12
6.1.2. Recursos.....	15
6.1.3. Variables.....	23
6.1.4. Atributs.....	27
6.1.5. Cues.....	29
<b>6.2. PRÀCTIQUES AMB CONCEPTES AVANÇATS.....</b>	<b>32</b>
6.2.1. Conjunts/Agrupaments (Sets).....	32
6.2.2. Submodels.....	34
6.2.3. Vistes.....	36
6.2.4. Setup.....	37
<b>6.3. PRÀCTIQUES AMB DEFINICIÓ DE TEMPS.....</b>	<b>41</b>
6.3.1. Constants.....	41
6.3.2. Distribucions Estadístiques.....	43
6.3.3. Expressions Matemàtiques.....	44
<b>6.4. PRÀCTIQUES AMB ELS MÒDULS BÀSICS DEL SISTEMA ARENA..</b>	<b>46</b>
6.4.1. Mòduls Lògics.....	46
6.4.2. Mòduls de Dades.....	51

	Pàgina
<b>6.5. PRÀCTIQUES AMB ELS MÒDULS AVANÇATS.....</b>	<b>53</b>
6.5.1. Mòduls Lògics.....	53
6.5.2. Mòduls de Dades.....	60
<b>6.6. PRÀCTIQUES DEL TRACTAMENT DE FALLIDES D'UN RECURS....</b>	<b>64</b>
6.6.1. Mòdul Lògic FAILURE (Advanced Process).....	64
<b>6.7. PRÀCTIQUES SOBRE LA METODOLOGIA DE SIMULACIÓ.....</b>	<b>66</b>
6.7.1. Creació d'Entitats.....	66
6.7.2. Utilització dels recursos mitjançant les entitats.....	66
6.7.3. Distribució de les entitats a través del model.....	69
6.7.4. Anàlisi dels costos del sistema.....	74
6.7.5. Animació de la simulació.....	76
6.7.6. Generació i anàlisi de reports.....	103
6.7.7. Altres eines i processos avançats.....	104
6.7.8. <b>PRACTIQUES D'INTERFÍCIE ENTRE ARENA I EXCEL.....</b>	<b>105</b>
6.7.9. <b>PRACTIQUES D'INTERFÍCIE ENTRE ARENA I ACCESS.....</b>	<b>108</b>
6.7.10. <b>PRACTIQUES D'INTERFÍCIE ENTRE ARENA I XML.....</b>	<b>110</b>
6.7.11. <b>PRACTIQUES AMB CONTROLS ActiveX.....</b>	<b>111</b>
6.7.12. <b>PRACTIQUES AMB ARENA VBA (Visual Basic For Applications).....</b>	<b>113</b>
<b>7. EXEMPLE DE PROCESSOS COMPLERTS.....</b>	<b>118</b>
<b>8. CONCLUSIONS.....</b>	<b>122</b>
<b>9. BIBLIOGRAFIA.....</b>	<b>125</b>
<b>10. ANNEXES.....</b>	<b>128</b>
10.1. Annex I. CD amb les pràctiques utilitzades.....	128
10.2. Annex II. Statistics Collection Variables.....	128

## Resum de Treball Final de Carrera

### Enginyeria en Organització Industrial

**Títol:** Implementació avançada de modelatge i simulació de sistemes mitjançant l'entorn ARENA <sup>TM</sup>.

**Paraules clau:** Modelatge, Simulació, Arena ; **Autor:** Josep Aranda Arjona

**Direcció:** Juli Ordeix Rigo ; **Data:** 19 de Gener de 2011

## Resum

En l'exercici de les funcions típiques de qualsevol activitat humana, l'home ha de prendre decisions d'un o altre tipus permanentment i aquesta situació porta afegit, per lo general risc i incertesa, el que compromet la qualitat i l'èxit de la decisió. Per ajudar i contrarestar aquesta situació, l'home ha desenvolupat a través del temps una diversitat d'eines que li permeten minimitzar el risc i la incertesa en la presa de decisions.

La simulació és una d'aquestes eines. Mitjançant la seva aplicació no només s'aconsegueix la comesa anterior, sinó que també es minimitzen els costos involucrats en la decisió mitjançant un millor ús dels recursos, la disminució del temps emprat per a la seva realització i la minimització de les probabilitats de risc.

Aquest projecte representa la continuació d'un primer estudi introductor de la simulació anomenat: *Estudi i guia docent per a modelatge i simulació de sistemes mitjançant l'entorn ARENA* <sup>[9]</sup>. Realitzat per l'alumna d'Enginyeria en Organització Industrial de la Universitat de Vic, Montse Carbonell Crosas, l'any 2008 i sota la codirecció del director d'aquest segon projecte, el professor Juli Ordeix Rigo.

Aquest nou projecte s'inicia amb una primera part teòrica, continguda dins del primer volum, la qual reforça els conceptes teòrics referents a la simulació amb ARENA, ja vistos en l'anterior projecte. Complementant aquells considerats bàsics i els de més utilitat i finalitza introduint nous conceptes avançats.

Els nous capítols de temàtica avançada, junt als primers més bàsics de la primera part són exercitats dins de la segona, formant el segon volum d'aquest projecte. El mateix requereix la participació activa de l'alumne, per tal de realitzar cadascun dels 89 exercicis pràctics que es plantegen i poder consolidar l'aprenentatge teòric d'aquesta eina avançada de simulació fent consultes als apartats teòrics recomanats dins de cada exercici.

La complexitat dels exercicis anirà augmentant gradualment i s'insisteix en seguir la metodologia presentada en el projecte per a realitzar-los tots de forma ordenada i ascendent. L'alumne quan acabi la part pràctica, haurà consolidat tota la part teòrica i serà capaç d'exercir com analista per tal de generar els seus propis projecte de simulació.

## Summary of Final Work of Career

### Engineering in Industrial Organization

**Title:** Implementation of advanced system simulation and modeling on ARENA™

**Key words:** Modelling, Simulation, Arena ; **Author:** Josep Aranda Arjona ;

**Direction:** Juli Ordeix Rigo ; **Date:** 1<sup>st</sup> January 19, 2010

## Summary

In the exercise of typical functions of any human activity decisions have to be made and that usually brings in risk and uncertainty compromising quality and success. This in turn, along with inexperience and pressure, might make decision-makers back away from facing the responsibility involved. To overcome this situation tools have been developed to minimize risk and uncertainty in decision-making. Simulation is one of these tools. By its means not only is inexperience offset, but employed resources and time are minimized, and risks mitigated.

This project is a continuation of an introductory simulation study of 2008 entitled: System modelling and simulation study and teachers guide, by Montse Carbonell Crosas, an industrial organization student at Universitat de Vic under the direction of the same supervisor of the current project Juli Ordeix Rigo.

Regarding its structure, this project extends the teaching of ARENA's software laying the knowledge the student is supposed to have acquired after the preceding project. It begins with the theoretical concepts already covered concerning ARENA, completing the most basic and useful. Afterwards new advanced concepts are presented and illustrated with a series of practical exercises. By conducting these exercises, which gradually increase their difficulty, the student will be able to analyze, generate and evaluate different system scenarios after building, verifying and validating the model.



# 1. OBJECTIUS.

La simulació avançada de projectes de modelatge i simulació de sistemes mitjançant l'entorn **ARENA** és un projecte, que ensenya a l'alumne a aplicar una metodologia i li proporciona també eines per a l'anàlisi de sistemes reals o virtuals mitjançant la captura d'una part de la complexitat i la variabilitat inherents a la realitat de tot sistema, per tal de constituir un suport robust en el moment de la presa de decisions en entorns complexos.

Mitjançant un model de simulació complex es poden fixar els següents objectius:

- **Optimització.** Permet determinar la configuració de certes variables que maximitzen o minimitzen una mesura de l'acompliment d'una o altre fita segons sigui el cas.
- **Predicció.** Permet estimar l'efecte tant de les variacions del sistema com de l'entorn.
- **Coneixement.** Permet:
  - Confirmar relacions causa-efecte de diferents events.
  - Confirmar que totes les variables rellevats son conegudes.
  - Avaluar la sensibilitat del model davant variacions en els paràmetres més importants.
- **Justificació.** Permet:
  - Decidir entre dues alternatives en base a un valor d'una mesura d'acompliment.
  - Generar noves idees si el model és creat mitjançant una bona animació.

Dins d'un sistema també es troben factors controlables e incontrolables que produeixen algun efecte en una determinada variable de resposta a priori complexa de determinar. Des d'aquesta perspectiva, es fixa també com a objectiu de la simulació avançada el fet d'avaluar l'efecte que diferents valors d'aquests factors tindrien per al sistema, tant controlables com incontrolables, sobre la o les variables de resposta que s'escullin.

Aleshores, cada conjunt de valors per als factor abans esmenats rep el nom d'escenari i cal avançar que els factors incontrolables no poden variar segons el criteri de l'analista, però, mitjançant la simulació, sí que es poden estimar els efectes que les variacions d'aquests tindrien sobre el sistema.

## 2. METODOLOGIA.

La ment humana reten moments concrets de passat de la persona i no pot recordar molta quantitat d'informació de cop. Aleshores, quan es tracta d'ensenyar s'ha de tenir en compte aquesta característica i desenvolupar una metodologia que faciliti aquest aprenentatge.

Dins l'àmbit d'aquest projecte es presenten dues metodologies per a l'estudi del mateix.

La primera metodologia té un caire objectiu i es detalla a partir de l'apartat 5 denominat **METODOLOGIA DE SIMULACIÓ** i ajudarà a l'alumne a centrar tots els passos que s'han de fer per a fer un projecte de simulació mitjançant ARENA.

La segona metodologia tracta aspectes més conceptuals i pretén ensenyar una nova guia d'estudi. Que ajudi a interpretar aquest projecte, per tal d'aprendre a simular processos avançats, aplicant la **METODOLOGIA DE SIMULACIÓ** de forma ordenada i correcte.

La primera metodologia és més fàcil d'aplicar i és l'objectiu principal a transmetre a l'alumne, dins de la primera part d'aquest projecte. Donat que tant només s'han de seguir els seus passos per arribar a simular correctament:

1. **Creem entitats.**
2. **Utilitzem recursos del sistema mitjançant les entitats.**
3. **Distribuïm el flux de les entitats a través del model unitàriament o en grups.**
4. **Analitzem els costos del sistema.**
5. **Animem la simulació.**
6. **Generem i analitzem reports.**
7. **Utilitzarem també d'altres eines i processos avançats.**

Però el fet de seguir aquest passos teòrics, no garanteix que l'alumne sigui capaç d'agafar l'eina i posar-se a desenvolupar projectes de simulació amb la solvència necessària.

Mentre feia la primera part del projecte me n'adónava dia a dia que allò que jo estava aprenent era molt extens, doncs a més de l'aprenentatge també tenia que consolidar tots els conceptes teòrics i la memòria teòrica era impossible de seguir punt per punt, si l'alumne no té ben consolidats uns coneixements previs.

L'aprenentatge de sistemes i llenguatge informàtics complexos no es pot fer llegint el manual teòric, aquests tracten temes que moltes vegades es posen seguits dins l'índex i no tenen cap relació entre sí.

D'aquesta manera es presentava per a mi un nou repte, desenvolupar quelcom per a fer que l'alumne pugui aprendre a simular mitjançant l'estudi d'aquest projecte.

Finalment, vaig decidir plantejar dins la segona part, un canvi radical de projecte. Per tal de generar una metodologia d'aprenentatge pràctica, que ajudés a l'estudi de la primera part teòrica.

La nova metodologia que s'aprendrà ha estat batejada com a "**Consolidació Pràctica**" i tracta d'aprendre els conceptes teòrics, seguint un guió pràctic. Això és el que s'ha fet i preparat dins d'una segona part eminentment pràctica, la qual segueix el mateix índex de temes que la primera part teòrica i on l'alumne podrà avançar amb es temes en paral·lel. Però gràcies a l'execució de les pràctiques, a més podrà anar consolidant exercici a exercici tots els coneixements teòrics de la primera part.

**Metodologia "CONSOLIDACIÓ PRACTICA" per al seguiment del projecte:**

**S'ha de començar llegint la part teòrica dins del Volum I, fins l'apartat 4.3 i quan es comenci l'apartat 4.4 s'ha d'agafar el Volum II.**

Encara que de cop es salti a l'apartat 6, es continuarà aprenent com si encara es trobi llegint la part teòrica a partir de l'apartat 4.4 fins a la seva finalització. Però ara l'alumne es trobarà immers dins de 89 exercicis pràctics que l'ajudaran a consolidar els conceptes teòrics avançats, d'una forma més didàctica i entenedora, doncs dins de cada exercici només es fan anotacions dels punts a treballar amb cada pràctica.

Sembla una mica enrevessat, però garanteix que l'alumne no es perdi dins dels obscurs manuals de ARENA i la recopilació de pràctiques que s'han fet asseguraran la consolidació teòrica des d'una vessant merament pràctica. D'aquesta manera s'assegura la participació de l'alumne, interaccionant contínuament amb l'eina, doncs cada nova pràctica el farà aprofundir més dins del seu aprenentatge sense hores de lectura intenses per tal d'aprendre a programar un entorn immens a partir d'una teoria complexa la primera vegada que es llegeix.

Amb tot lo exposat, aquest projecte pretén superar la complexitat de l'aprenentatge de ARENA, la qual be donada en el moment que s'han de saber molts conceptes teòrics per a realitzar el més petit dels exercicis pràctics.

Una advertència, no es pot pretendre agafat exercicis aleatòriament, doncs l'alumne no serà capaç de seguir el que volen transmetre. Aleshores, no es poden saltar pràctiques anant de darrera al davant o al mig, cada pràctica ha estat col·locada en l'ordre correcte.

Per tal que només es presentin els conceptes nous que cada pràctica introdueix, s'assumeixen d'altres ja vistos en pràctiques anteriors. Encara que també i un cop avança l'estudi es troben pràctiques que fan referència explícita a les anteriors, les quals per si soles no mostren res aparentment però consoliden el coneixement teòric proposat, tonant repassar pràctiques anteriors amb una altra ordre concret.

### 3. INTRODUCCIÓ.

La simulació pot ser aplicada a molts camps, però si parlem per exemple dels processos industrials automàtics, veiem que els mateixos venen evolucionant des de fa anys i en moltes ocasions més ràpid del que ho han pogut fer moltes plantes industrials.

En aquest context d'evolució i desenvolupament de les automatitzacions, moltes vegades condicionat per l'herència anterior, s'ha de tenir cura en el moment de realitzar l'adaptació a la tecnologia actual i fer-ho d'una manera coherent i previsible. El projecte que es presenta neix per afavorir aquesta adaptació, mitjançant la presentació d'una metodologia de simulació i la seva aplicació pràctica en un entorn real dins del millor software de simulació actual, el programa **ARENA**<sup>TM</sup> de l'empresa nord-americana **ROCKWELL**.

**A través del procés de disseny d'un model del sistema real i dirigint l'experiment cap a ell es pot entendre el funcionament del sistema, la qual cosa permetrà prendre les millors decisions.**

La simulació avançada d'aquest projecte parteix dels conceptes bàsics inicials, teòrics i pràctics, que introdueix el projecte precedent : **Estudi i guia docent per a modelatge i simulació de sistemes mitjançant l'entorn ARENA** <sup>[9]</sup>. Serà imprescindible haver estudiat i realitzat les pràctiques proposades en el mateix, per continuar amb l'estudi avançat de l'entorn **ARENA** exposat en aquest projecte.

Però, l'aplicació de la simulació per a buscar l'essència del sistema implica, per lo general, manegar un volum considerable de dades i l'execució d'un alt nombre de repeticions del procés. Donat que l'objectiu és el d'obtenir una adequada història artificial que permeti prendre decisions amb un alt risc de confiabilitat.

Aleshores, aquest tractament massiu de dades només és factible que es faci mitjançant l'ajuda computacional amb un software especialitzat. D'aquest tipus existeixen al mercat una àmplia gamma i un d'ells es ARENA, estudiat en aquest projecte. L'absència de guies pràctiques en el nostre país per al seu aprenentatge han motivat la seva realització.

Es demostrarà un cop finalitzat l'estudi, com la simulació facilitarà tant la creació de nous projectes d'automatització com els canvis dels existents. Aquests no han de buscar solament la consecució de l'objectiu, emprant les eines de simulació han de tenir com a fita una realització clarivent, estructurada i sistemàtica, que faciliti la modificació en les fases inicials de tot projecte o condioni el menys possible les futures adaptacions que s'hagin d'aplicar, evitant un malbaratament de recursos.

Les característiques principals d'aquest projecte s'engloben dins d'un entorn de simulació eminentment pràctic amb un punt de vista purament teòric, aportant diverses taules resum amb les principals

característiques del software. Les quals podran convertir-se en manual de consulta i referència per als futurs usuaris o per aquells que vulguin aprendre a utilitzar aquest software.

Cada exercici comença a partir del supòsit pràctic redactat, a continuació s'aplica la metodologia de simulació emprant les eines que aconsella el projecte per a cada fase i al final es mostra a l'alumne el model programat dins l'entorn **ARENA**<sup>TM</sup>. Una vegada finalitzat l'exercici li permetrà visualitzar d'una forma gràfica el nou procés i fruit del mateix, també se li mostra com obtenir els informes tècnics.

**La simulació pretén obtenir l'essència d'alguna cosa sense tenir en compte la realitat e implica la modelació dels processos o sistemes, de tal manera que el model emuli la resposta dels sistemes actuals o proposats com esdeveniments que tenen lloc en el temps.**

Durant la realització dels exercicis, un cop vist el funcionament de la simulació, es proposaran diversos canvis en el valor de les variables del sistema. L'alumne podrà així estudiar el procés, variant les restriccions de la simulació en temps d'execució i veurà les relacions causa efecte que cadascuna d'elles introdueix al sistema.

Estructuralment, cada fase de la metodologia que s'ha d'emprar en la simulació, es desenvolupa directament dins de cada exercici pràctic que l'alumne ha de realitzar. No s'exposen els conceptes teòrics de la fase dins l'àmbit de cada pràctica, la seva realització es concentra només en l'aplicació de la metodologia i en la programació avançada de l'entorn **ARENA**<sup>TM</sup> per part de l'alumne, objectiu d'aquest projecte.

Amb la finalitat de poder veure les simulacions de cadascun dels tres supòsits pràctics d'automatització industrial estudiats, els codis en l'entorn **ARENA**<sup>TM</sup> de cadascun també s'incorporaran mitjançant suport informàtic (CD Rom) al projecte.

# *Primera Part*

## *Conceptes Teòrics*

### 3. RESUM DE L'ENTORN ARENA.

ARENA té els seus orígens l'any 1982, quan Dennis Pegden va publicar el primer llenguatge de simulació de propòsit general per a modelar sistemes de fabricació mitjançant un PC.

Aquesta primera aplicació constava de diverses característiques per a modelar processos de fabricació de caràcter especial, que feien que aquest llenguatge fos bastant útil i eficient en el moment de modelar sistemes grans i complexes.

Però va ser a partir de 1993 quan es va introduir el sistema de modelatge ARENA, el qual permetia crear ambients gràfics e interactius per al disseny de models mitjançant l'ús del llenguatge SIMAN.

Amb base a aquest llenguatge es poden crear models que simulin àrees específiques dels processos, com el transport d'elements i la comunicació entre molts d'altres, mitjançant un entorn de treball integrat per a construir models de simulació. ARENA integra totes les funcions necessàries per al desenvolupament d'una simulació d'èxit com son les animacions, l'anàlisi d'entrada i sortida de dades i la verificació del model.

Abans de començar a detallar les característiques de l'entorn ARENA, repassarem els conceptes més genèrics de la simulació:

- **SISTEMA:** Es un conjunt d'elements que es troben interaccionant buscant un fita o fites comuns. Per portar-ho a terme operen sobre una dada o informació, sobre energia, matèria o organismes, amb l'objectiu de produir com a sortida d'informació, energia, matèria o organismes. Un sistema es doncs un conjunt de components interrelacionats que, d'una forma organitzada, reben entrades i les processen, emetent sortides per obtenir una fita comú.

#### Representació d'un sistema.



**Classes de sistemes.**

- Naturals i artificials.
  - Deterministes i probabilístics.
  - Socials, home-màquina i mecànics.
  - Oberts i tancats.
  - Permanents i temporals.
  - Estables i no estables.
  - Adaptatius i no adaptatius.
- **MODEL:** És una representació de la realitat que es desenvolupa amb el propòsit d'estudiar-la. En la majoria dels anàlisi no és necessari considerar tots els detalls, de tal manera que el model no només és un substitut de la realitat, sinó també una simplificació d'aquesta.

Els models es poden classificar en:

- Models icònics.
- Models analògics.
- Models simbòlics, els quals alhora inclouen:
  - Models deterministes.
  - Models estocàstics o probabilístics.
  - Models dinàmics.
  - Models estàtics.
  - Models continus.
  - Models discrets

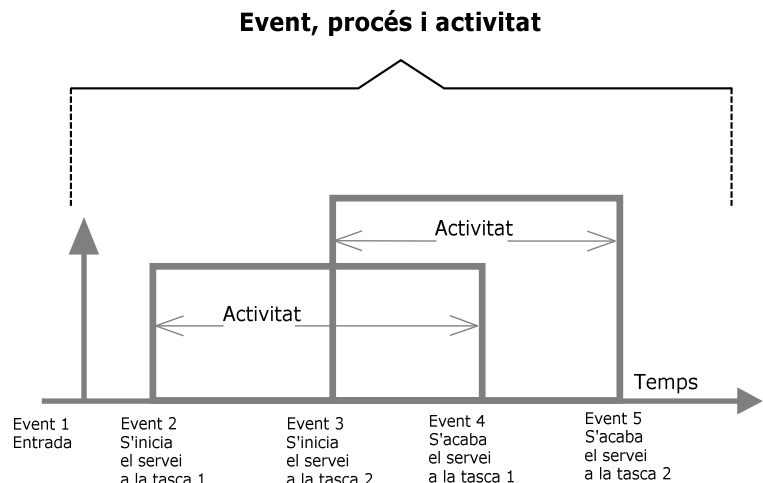
Els models tenen les següents característiques:

- Confiabilitat.
- Simplicitat.
- Baix cost en desenvolupament i operació.
- Usabilitat.
- Fàcil comprensió del model i dels seus resultats.
- La relació cost-benefici ha de ser positiva.



### 3.1. CARACTERÍSTIQUES BÀSIQUES DE LA SIMULACIÓ AMB ARENA.

- **EVENT:** Es el fet que far canviar l'estat del sistema. Hi ha esdeveniments interns i externs. Els mateixos també es coneixen com a endògens i exògens respectivament.
- **PROCÉS:** Un procés és un conjunt d'activitats o events coordinats i organitzats, els quals és realitzen o succeeixen de forma alternativa o de manera simultània, amb una finalitat determinada.
- **ACTIVITAT:** Conjunt de fenòmens que succeeixen dins d'un procés amb una durada determinada, un cop s'ha iniciat l'event que ha originat el seu inici.



El desenvolupament de models de simulació mitjançant el sistema ARENA té moltes avantatges degut a les seves característiques bàsiques:

- És una potent eina de simulació que comprèn un entorn amigable, el qual està especialment dissenyat per a les persones que no tenen coneixements de programació.
- Els utilitaris que ofereix són de fàcil usabilitat i disposen d'una excel·lent capacitat gràfica.
- Ofereix gran versatilitat, doncs es pot modelar des d'una fàbrica de cotxes fins a una sala d'espera d'un hospital.
- És compatible amb els productes MICROSOFT OFFICE.

Però ARENA té també punts febles, a saber:

- El sistema complet és molt car i l'edició per a estudiants té moltes limitacions.
- És molt difícil fer córrer un model creat en ARENA en qualsevol programa de simulació. Això es dona bàsicament perquè és difícil sincronitzar els rellotges amb els que funcionen els programes.
- La documentació i ajuda que aquest sistema ofereix és escassa, a més, no és lo suficientment clara i alguns dels exemples que presenta tenen errors.

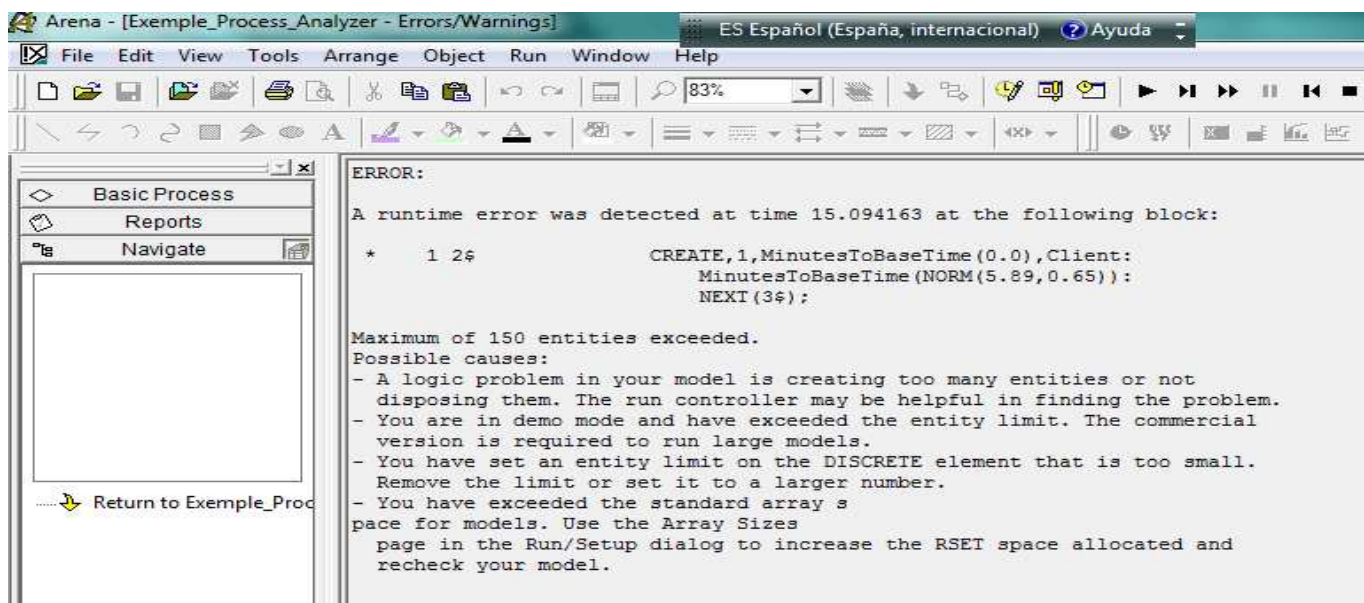
### 3.2. LIMITACIONS DE LA VERSIÓ PER A ESTUDIANTS.

Existeixen diverses restriccions dins de la versió per a estudiants, les quals s'han de tenir en compte si no volem tenir errors en temps d'execució inesperats.

El següent llista ens mostra les limitacions que porta incorporades aquesta versió.

Components del Model	Limitació Versió Acadèmica
Entitats Concurrents	Màxim 150 circulant pel sistema al mateix moment de simulació
Objectes SIMAN	Màxim nombre d'objectes SIMAN dins del model.
Objectes Gràfics	Màxim de 150 objectes d'animació gràfica
Elements de Bloc Inclosos	0

A continuació es mostra un exemple de l'error més crític que haurem de controlar, el nombre màxim d'entitats concurrents pel sistema, el qual s'ha vist que no pot superar les 150.



La bibliografia consultada aconsella establir controls mitjançant blocs DECIDE per evitar el desbordament per excés d'unitats. Es pot aturar el flux d'entitats abans del desbordament, però moltes vegades en segons quins processos simulats, s'ha de superar aquest número i limitarà la fiabilitat de la simulació.

### 3.3. COMPATIBILITAT AMB MICROSOFT OFFICE.

Anteriorment s'ha comentat que la compatibilitat amb els productes MICROSOFT OFFICE és una de les característiques principals del sistema ARENA, això vol dir, entre d'altres aspectes que la barra d'eines, els menús i fins i tots les tecles d'accés directe d'aquest sistema són similars a les de la suite de Microsoft.

Entre els aspectes comuns als dos sistemes en troben:

- **MENUS ESTÀNDARDS:** Els menús dins ARENA segueixen els mateixos alineaments en la organització dels menús que les aplicacions OFFICE.
- **BARRA D'EINES I TOOL TIPS:** Es poden afegir o eliminar fàcilment elements a la barra d'eines i es pot conèixer la funció de cada botó al posicionar-se sobre cadascun d'ells. El Tool Tip es desplega sobre la pantalla mostrant el nom del botó i també es pot reorganitzar la barra d'eines mitjançant un arrossegament i enganxat, principal característica dels productes OFFICE.
- **ACCEBILITAT EXTERNA:** La integració amb OFFICE permet, a més, que s'accedeixi externament a totes les funcions que incorpora ARENA des de qualsevol programa de la família Microsoft. Els resultats que s'obtenen de la simulació es poden portar de forma immediata i directa a una fulla de càlcul, una base de dades o a un processador de text.
- **VISUAL BASIC FOR APPLICATIONS:** L'entorn ARENA permet que es puguin crear, dissenyar, executar, modificar i examinar models des de OFFICE en qualsevol moment mitjançant l'entorn Visual Basic.

ARENA integra una versió completa de l'assistent denominat Visual Basic for Applications, amb el seu editor i el llenguatge de programació Visual Basic. Permetent dissenyar i desenvolupar fàcilment programes en Visual Basic, mitjançant els qual podem interaccionar amb l'entorn de simulació en temps real i inclús crear nous blocs per al modelatge, que complementin als que el sistema ARENA ja incorpora per defecte.

El terme "for applications" fa referència al fet que el llenguatge de programació i les eines de desenvolupament es troben integrats dins ARENA igual que en totes les aplicacions del OFFICE.

La comunicació externa que permet obrir i crear un model en ARENA per a controlar la seva execució es realitza mitjançant objectes DAO que utilitzen funcions de transferència entre mòduls i aquest tema es desenvoluparà dins l'àmbit d'aquest projecte d'implementació avançada per al modelatge i simulació de sistemes mitjançant l'entorn ARENA.

### 3.4. CONCEPTES BÀSICS.

Es tracta de fer un repàs dels termes que s'han de tenir clars un cop s'hagi realitzat el projecte precedent (*Estudi i guia docent per a modelatge i simulació de sistemes mitjançant l'entorn ARENA*<sup>[9]</sup>) i no es pretén tornar a fer un anàlisi exhaustiu d'aquests. Es dona per fet que els mateixos ja s'han assimilats un cop estudiat aquest primer projecte i se'ls cita per tal de refrescar-los i remarcar la seva importància, abans de començar l'estudi d'aquest projecte amb els termes més avançats.

Els elements bàsics que interactuen dins del projecte de simulació estan format per: entitats, recursos, variables, atributs i cues. A continuació s'exposa una breu descripció de cadascun d'ells:

- **ENTITATS:** Representen a qualsevol objecte que es mou a través del sistema, tant real com abstracte i que provoca canvis en les variables de resposta.

Cada entitat té característiques pròpies les quals seran definides i es denominen atributs. Les entitats es mouen per tots els mòduls del model fins que troben una espera, una cua o són finalment retirades del sistema, un cop han complert la funció per a les quals varen ser creades.

- **RECURSOS:** Element estacionari que pot ser ocupat per una entitat amb capacitat finita, la qual pot variar amb el temps.

Hi ha dos tipus de recursos: **Estàndard i Submodels** en l'interior del quals hi haurà una altra lògica d'operació.

És molt important diferenciar el procés del recurs, doncs ARENA considera al segon com la unitat de mesura de capacitat del primer i han de tenir noms diferents. Per exemple podem crear un procés anomenat "**tornejat**", però dins del mateix s'ha d'especificar les unitats del recurs que el procés farà servir, el qual es pot anomenar per exemple "**torn**".

Els estats pel que passen els recursos durant la simulació són: **OCUPAT, OCIÓS, INACTIU o AVARIAT**.

La capacitat d'un recurs es refereix al número d'entitats que poden reservar-lo al mateix temps, la qual pot variar en funció del temps mitjançant programació (Schedule) quan es defineix el recurs, però quan parlem de quantitat d'un recurs ens referim al nombre de recursos que s'utilitzen.

Un altre manera d'expressar-ho es dir que capacitat defineix el processament en paral·lel i quantitat representa el procés en sèrie. Per exemple, un recurs "**torn**" té com a capacitat "2" el que vol dir que pot produir dues peces alhora i si la quantitat també es "2" vol dir que hi haurà dos recursos o màquines torn i que la mateixa peça serà processada primer en una màquina i després en l'altre.

Temps de descans (Downtimes) o falles (Failures) poden també ser associats com a recursos per a simular activitats programades o no, com pauses o manteniment de màquines.

Es important destacar que un recurs amb capacitat superior a "1" quan és representat dins del model per un únic recurs, per a finalitats estadístiques les dades seran representades per a tot el conjunt i no per a cada part per separat. Si fossin necessàries estadístiques per a cada part, han de ser definides com a recursos independents i reunides com a conjunt o agrupament per al seu procés (**Set**).

Els recursos sempre seran representats com a figures estàtiques, les quals poden variar en funció de l'estat dels mateixos (en espera, ocupat, etc.).

Mitjançant la programació d'un calendari horari per recurs (**SCHEDULE**), **ARENA** ofereix la possibilitat d'establir un patró de canvi en la capacitat del recursos. Aquesta part es troba molt detallada dins del projecte anterior<sup>[9]</sup> si es fa la seva pràctica 6, anomenada assemblatge i prova de peces electròniques (pàgina 79).

Amb base en el temps, aquests canvis de capacitat poden tenir una durada constant o aleatòria i es repetiran periòdicament en el model. També s'ofereixen opcions de com s'ha de fer servir els canvis de capacitat un cop aquest succeeixen (apropiatu, espera o ignorància).

Sols es permet un horari per recurs, per exemple, capacitat per a processar 5 unitats en 6 hores, 3 unitats en 2 hores, 1 unitat en 1 hora, etc.

- **VARIABLES:** Representen característiques del sistema i son de caràcter global amb un sol valor per a qualsevol part del sistema. Poden ser predeterminades pel programa o definides per l'usuari i es defineixen mitjançant un nom i amb un valor numèric que representi l'estat del sistema.

Dins l'apartat 1.3.1 es mostra una llista amb les variables predefinides dels components del model més importants.

- **ATRIBUTS:** És una característica pròpia de cada entitat i serveixen com a etiquetes o codis de barres, les quals estan inserides sobre cada entitat i poden ser utilitzades per prendre decisions dins del flux del procés.

Es poden definir tants atributs com l'usuari vulgui per tal de modelar el sistema, però, cada entitat individual té el seu propi valor d'atribut. Això implica que per determinar aquest valor, a diferència de les variables, s'ha d'examinar primer l'entitat que el porta.

Els atributs es defineixen per un nom, pes, color, etc., i han de tenir un valor numèric que indiqui qualsevol cosa per l'usuari. Per a qualsevol alteració d'un valor global que afecti a tots els elements de la simulació hem d'utilitzar les variables i no els atributs.

Com exemple, tenim atributs que ja neixen com a entitats, com pot ser el número de sèrie d'una entitat (Entity.SerialNumber) i d'altres que poden ser definits automàticament en temps d'execució.

Els atributs s'assignaran amb valor zero tant punt es creï la nova entitat, encara que siguin entitats diferents i la referència a l'assignació de l'atribut es faci posteriorment. Per exemple, en un mòdul Assign podem tenir diferents casos d'assignació d'atributs:

- **Cas 1:** L'atribut color pot tenir valor de 1, 2 o 3 quan els colors son groc, blau i vermell respectivament.
  - **Cas 2:** Es poden representar atributs el valor dels quals sigui únic per a cada entitat del sistema, així si per exemple parlem de números de factures, l'atribut pot ser 0001 per a l'entitat 1 i per cada nova entitat que entri al sistema es pot incrementar en una unitat i així l'atribut tindrà valors únics per a múltiples entitats.
  - **Cas 3:** Els atributs es poden assignar segons la distribució. Per exemple si la mitja d'un procés dona un pes mig que segueix una distribució normal de 110.47 gr i una desviació estàndard de 7.3 gr. En aquest cas el programa assignarà per a cada nova entitat i sota un atribut que podem denominar pes, un valor numèric que seguirà aquesta distribució NORM(110.47,7.3).
- **CUES:** Aquestes serveixen com a espai d'estada per aquelles entitats que no poden continuar seguint el flux normal del procés per algun motiu, com pot ser el fet que un recurs es trobi ocupat o esperant alguna autorització.

Existeixen bàsicament dos tipus de cues: les individuals i les internes. En el primer cas, la fila té un nom, un mode de selecció, una forma específica d'extreure-hi les entitats (First-In, First-out ; Highest attribute First ; etc) i una capacitat pròpia. Les entitats en files individuals poden ser mostrades a l'animació del model, es poden recol·lectar les seves estadístiques i poden ser buidades directament.

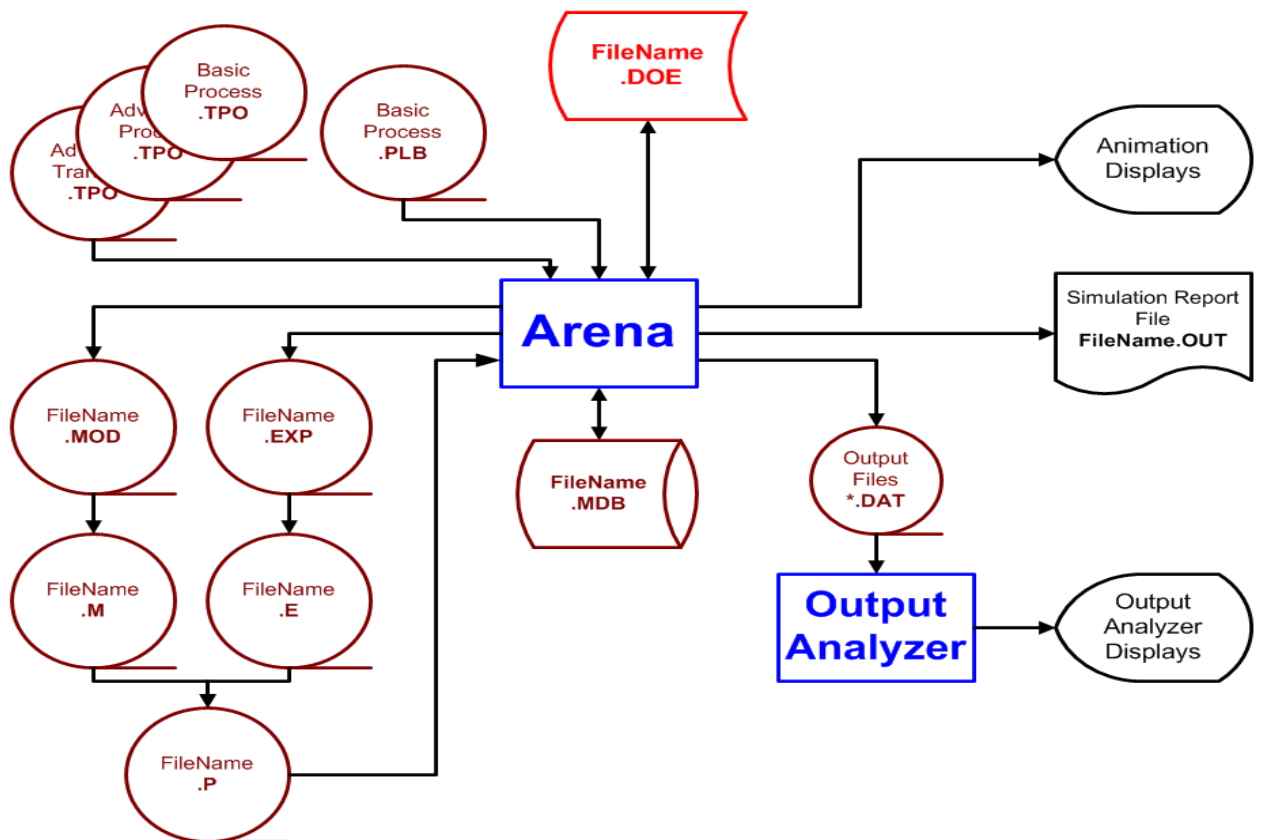
En el cas de les cues internes, funcionen bàsicament extraient entitats "First-in, First-out" i no poden ser animades, no recol·lecten estadístiques i es buiden mitjançant altres mecanismes.

- **FITXERS:** L'entorn ARENA genera al seu voltant una sèrie de fitxers. Segons el seu propòsit les seves extensions poden variar en temps d'execució o es poden mantenir fixes durant tota ella.

Molts usuaris novells de l'entorn ARENA tindran la necessitat durant l'aprenentatge del sistema de consultar aquests fitxers. A continuació s'exposa un esquema gràfic de molta utilitat per tal d'avaluar gràficament la relació entre tots ells.

Com a més destacats tenim:

- L'extensió **.M** i el **.E** son fitxers intermitjos del programa en execució i finalment són s'emmagatzema dins el fitxer **.P**. El mateix és utilitzat per a monitoritzar l'execució i per a rastrejar els possible errors mitjançant el Process Arena Analyzer (PAN).
- Output File **.DAT** s'utilitza per analitzar la sortida de la simulació mitjançant el Arena Output Analyzer.
- Extensió **.DOE** guarda tot el model simulat en format ARENA. Però l' extensió **.MOD** i la extensió **.EXP** s'utilitzen per a emmagatzemar el model lògic i de dades en el llenguatge format original sobre el qual està basat el sistema ARENA, SIMAN.
- La base de dades **.MDB** emmagatzema tots els resultats estadístics de la simulació i es poden exportar o consultar aquestes dades fàcilment cap a, o des d' altres entorns.



### 3.5. CONCEPTES AVANÇATS.

En aquest apartat podem dir que comença la metodologia d'aquest projecte, presentant conceptes avançats d'una forma lo més ordenada possible, per tal que el lector aprengui una metodologia de simulació.

Tanmateix només es profunditzarà en aquells elements que no hagin estat vistos en el projecte precedent (*Estudi i guia docent per a modelatge i simulació de sistemes mitjançant l'entorn ARENA*) i per lo tant siguin nous dins l'estudi del sistema.

Però el lector d'aquest projecte que hagi seguit l'anterior, també se n'adonarà que moltes vegades que es repeteixen el termes ja vistos abans i això bé donat per a la necessitat d'independència per al seguiment d'aquest en el cas que no s'hagi llegit l'anterior, a la continua interrelació entre els conceptes bàsics i els avançats i a la necessitat de completar o reforçar els punts que es consideren claus per a la simulació i han de sortir als dos projectes.

Així doncs, comencem l'estudi avançat per al modelatge i simulació de sistemes mitjançant l'entorn ARENA amb els elements avançats del mateix següents:

- **CONJUNTS/AGRUPAMENTS (Sets):** Permeten simplificar l'ús a recursos similars dins de cada procés. Quan tenim un conjunt d'elements dins de la simulació que tot i tenir característiques individuals específiques, s'hi accedeix de forma similar, poden ser agrupats en blocs mitjançant el bloc de dades bàsic "Set".

Per exemple imaginem que un client entra a una botiga i existeixen diferents recursos denominats

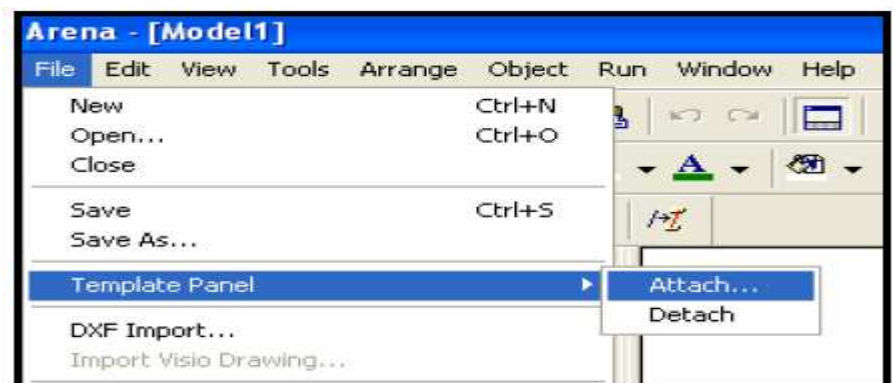
"dependenta<sub>1</sub>",

"dependenta<sub>1</sub>",

"dependenta<sub>n</sub>" per

atendre'l. Ell pot escollir qualsevol d'elles, però encara que qualsevol dependenta el pugui atendre

diferent, doncs tenen diferents atributs personals, les podem agrupar (**Set**) sota el recurs "dependentes". Quan una entitat "client" entri al mòdul de procés "botiga" es reserva per a ell una unitat del recurs "dependentes", el qual pot ser qualsevol de les dependents lliures agrupades anteriorment.





Res impedeix que un recurs pugui formar part de diferents agrupaments al mateix temps. En l'exemple anterior el recurs “**dependental**” també pot formar part d'un grup denominat “**caixeres**” i ser reservat mitjançant el recurs “**dependentes**” o “**caixeres**”.

- **Els “TEMPLATES”:** Per defecte ARENA s'obre només amb els mòduls bàsics, per construir qualsevol model, el primer pas serà el de definir quin conjunt de plantilles o “**Templates**” utilitzarem, els quals ens mostraran tots els blocs disponibles pel programa.

A continuació s'exposa la llista dels diferents fitxer amb extensió .TPO, els mateixos es carreguen mitjançant la opció “ **File -> Template Panel -> Attach...** “

Components	Elements	Applications
Template Panel Modules	Template Files	
Basic Edition	BasicProcess.tpo	Basic modeling modules
Standard Edition	AdvancedProcess.tpo	Advanced modeling modules
Standard Edition	AdvancedTransfer.tpo	Material handling system modeling modules
SIMAN	Blocks.tpo	Original SIMAN logic modules or blocks
SIMAN	Elements.tpo	Original SIMAN data/definition modules or elements
Contact Center Edition	ContactData.tpo	Contact Center Edition Template
Contact Center Edition	Script.tpo	Contact Center Edition Template
Contact Center Edition	CSUtil.tpo	Custom Template building capabilities
Packaging Edition	Packaging.tpo	Arena Packaging Edition Template
Factory Analyzer Edition	Factory.tpo	Factory Analyzer Modules
Factory Analyzer Edition	FactoryBlocks.tpo	Factory Analyzer Blocks Panel
Factory Analyzer Edition	FactoryElements.tpo	Factory Analyzer Elements Panel
Professional Edition	UtilArena.tpo	Custom Template building capabilities

- **SUBMODELS:** La utilització de submodels facilita la visualització global del sistema, permet amagar elements repetitius i obtenir estadístiques de blocs de lògica com a un tot en lloc d'individualment, constituint doncs una manera molt eficaç d'organitzar el nostre procés.

Els submodels poden ser connectats a d'altres mòduls, altres submodels o simplement treballar només. Mitjançant la opció de menú **Object->Submodel->Aggregate** podem introduir la lògica, gràfics i animacions dins del submodel unitàriament o marcar un conjunt de blocs i convertir-los en blocs tots junts. El procés invers seria **Object->Submodel->Unaggregate**, el qual desfà la lògica d'un submodel i l'envia al nivell immediatament superior.

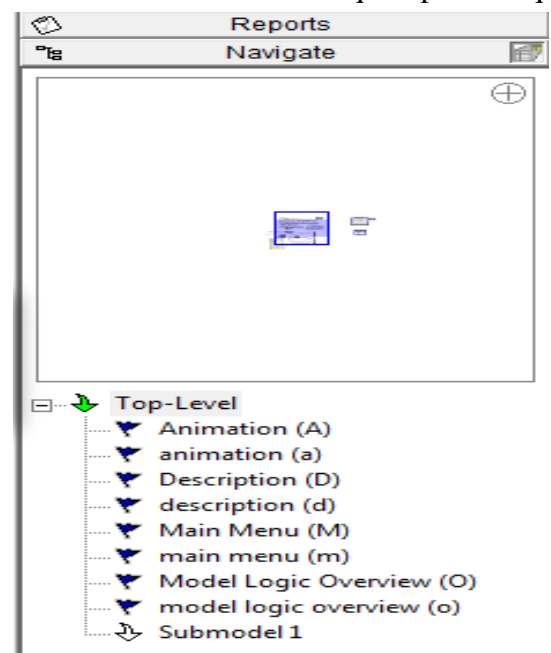
Com a observació afegim que, si utilitzem un mòdul **BATCH** dins d'un submodel, per a realitzar un agrupament d'entitats temporal. L'entitat representativa del conjunt, ha de ser posteriorment desagregada utilitzant un mòdul **SEPARATE**, abans d'abandonar el submodel per tornar a un nivell jeràrquic superior o formar part d'una altra submodel.

- **VISTES:** Existeix un comando dins ARENA anomenat “**Named Views**” el qual permet que s'utilitzin tecles curtes per a fer referència a alguna vista del model o alguna part específica del mateix.

Les mateixes es creen un cop s'hagi creat, amb el zoom corresponent, la vista que volem veure a la pantalla principal. A continuació prement el botó dret sota el menú “**Navegate**” i la seva opció “**Top Level**” podem navegar fins a un nou menú, seleccionant “**View->Named Views->Add**” per tal de crear la vista.

Una vegada creada la nova vista, hi accedirem mitjançant el panell de navegació o prement la tecla abreviada assignada, la qual es mostra entre parèntesi.

A través del panell de navegació també podem accedir als submodels generats.

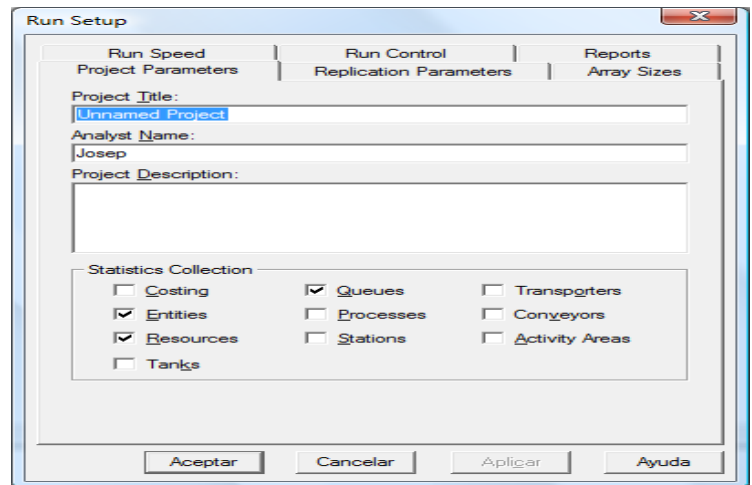


- **SETUP:** A continuació es fa un repàs d'aquelles opcions generals a tenir en compte abans d'engegar el nostre procés de simulació i que trobem sota el menú “**Run->Setup**”.

Aquestes opcions es classificaran dins d'un menú on apareixen les sis categories següents:

**1. Paràmetres del Projecte:** Aquí dins es defineixen les informacions bàsiques del projecte de simulació, inclou el nom del projecte i el de l'analista del projecte doncs aquest sortiran en els informe i també s'indiquen quins i de quin tipus han de ser les dades que es recolliran durant l'execució.

Existeixen diferents “**Check Box**” on podrem seleccionar els tipus de les estadístiques que seran recol·lectades com poden ser:

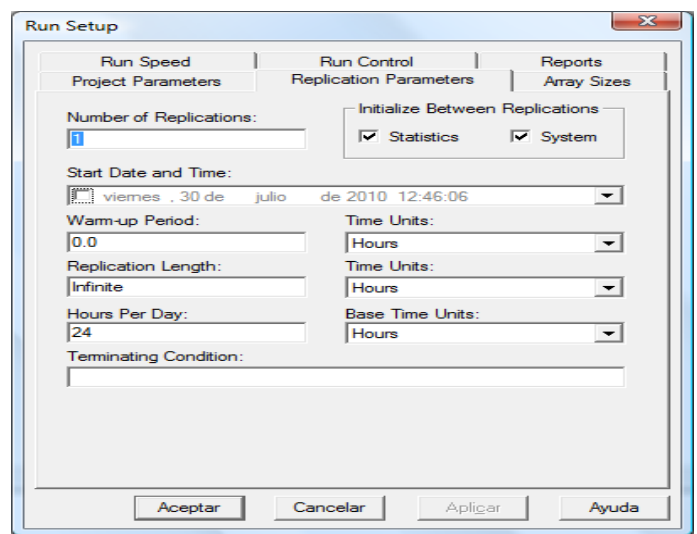


- **Els costos (Costing).** Apareixeran en els informes totes les dades relatives als atributs de temps i costos per a cada entitat.  
Per exemple: Entity.Vatime, Entity.VACost (on VA és l'abreviació de Value-Added o Valor Agregat), Entity.WaitTime, Entity.WaitCost i d'altres que permetran conèixer el cost relatiu a cada procés o a cada entitat produïda.
- **Les entitats (Entities).** S'informarà dins els informes i per a cada entitat, tant el temps en el qual l'entitat ha generat valor afegit com del que no l'ha generat, el temps d'espera, el temps de transferència i d'altres temps.
- **Els recursos (Resources).** Selecció permetrà saber per a cada recurs específic, per exemple, dades com el nombre de recursos ocupats o la seva taxa d'ocupació.
- **Els tancs (Tanks).** S'habilitaran les estadístiques per als tancs, les quals poden ser el nivell de líquid, la quantitat afegida o la que s'ha servit des d'aquest element.
- **Les cues (Queues).** Permetrà saber el temps d'espera en cada cua o el nombre mig d'entitats en la mateixa entre d'altres.
- **Els processos (Processes).** Obtindrem el temps mig de processament o el número d'entitats processades entre d'altres.
- **Les estacions (Stations).** Si s'activa es recol·lecten estadístiques sobre el nombre d'entitats que arriben a les estacions de forma individual o entren a un mòdul sempre i quan estigui activat el “**Check Box**” anomenat “**Report Statistics**” del propi mòdul.
- **Els elements de transport (Transporters).** De forma molt similar als recursos obtindrem el número de transports utilitzats o la taxa d'utilització de cada unitat.

- **Els conductors (Conveyors).** D'aquest elements se'ns informarà dades com la taxa d'utilització, el tamany de la cua acumulada quan hi ha bloqueig o el temps de bloqueig entre d'altres.
- **Les àrees d'activitat (Activity Areas).** Si s'activa permet acumular estadístiques de temps com per exemple "accum va time", "accum wait time", "total accum time", "accum va cost", "accum wait cost", "total accum cost" per a les Activity Area. Les quals estan associades a una o més "Stations" per a recollir les seves estadístiques.

**2. Paràmetres del Rèplica:** Dins d'aquests paràmetres informarem al sistema de tots aquelles informacions sobre cada rèplica de la nostra simulació que tenen un caire rellevant.

Incloent per exemple, el número de rèpliques que s'executaran, la data i l'hora d'inici de la simulació, el període d'escalfament junt amb la seva base de temps que triga el sistema en posar-se a règim (**Warm-up Period**), en el qual no es recolliran estadístiques, el temps



que durarà la cada rèplica (**Replication Length**), les hores que es consideraran per dia de simulació i si existeix una condició que en el moment de passi a ser "vertadera" o valer "1" aturarà la simulació encara que no s'arribi al temps màxim de durada indicar anteriorment si es necessita.

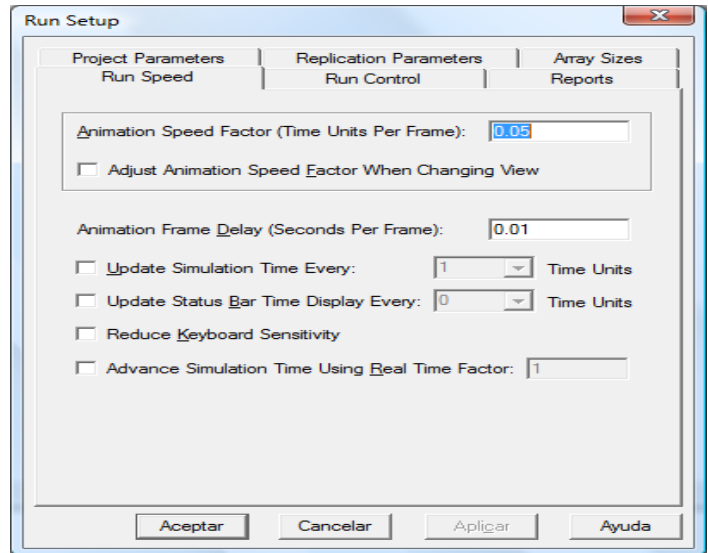
Existeix també dos "Check Box" sota el títol de "Initialize Between Replications" d'utilitat quan es facin corre més d'una rèplica i que informarà d'allò que es reinicialitzarà entre rèpliques. Si es marca "Statistics" es poden acumular les estadístiques entre rèpliques i si es marca "System" després de cada rèplica totes les entitats seran remogudes del sistema, l'estat de cada element i el de les variables seran inicialitzats i tornaran a tenir el valors originals en cada rèplica.

**Nota important:**

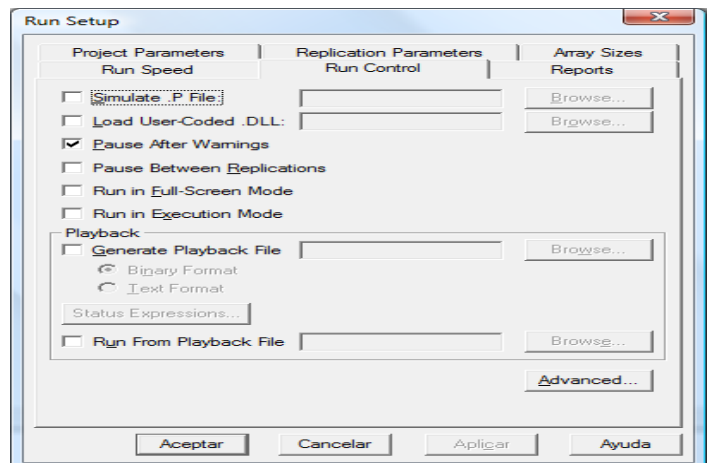
**Per una bona visualització de les estadístiques de temps es bàsic que les unitats del temps total d'execució (Time units) i les bàsiques (Base Time Units) coincideixin i s'aconsella que les seleccionem amb minuts.**

**3. Velocitat d' Execució:** La velocitat d'animació d'un model és determinada per diferents factors, incloent-hi la freqüència amb les quals les interrupcions de teclat son ateses.

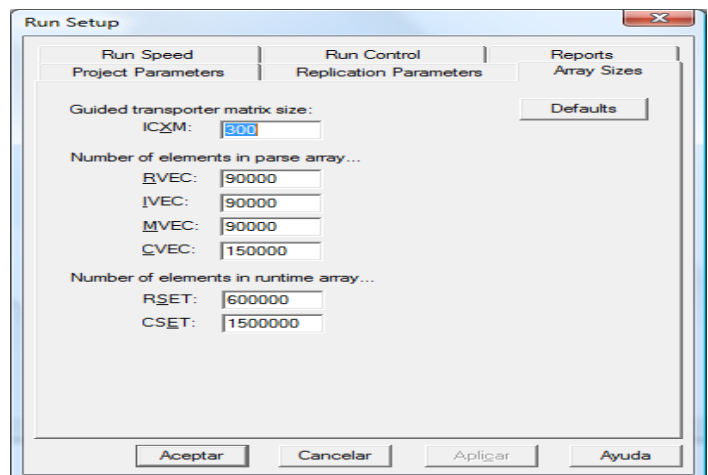
Així doncs la velocitat de l'animació dependrà del tamany de l'interval de temps simulat que existeix entre la captura d'un i altre quadre i la taxa en temps real en la qual els quadres son mostrats.



**4. Control d' Execució:** S'indiquen diferents aspectes per al control de l'execució de la simulació com el que es produeixen pauses entre avisos (Warnings), entre rèpliques o s'executi la simulació a pantalla completa entre d'altres.



**5. Paràmetres dels Informes:** També podem gestionar quan volem fer aparèixer els informes, sempre, mai o traient un avís i podem especificar la categoria per defecte de l'informe d' entre els informes que ARENA proporciona en el "Reports Panel".



**6. Tamany de les Variables:** Aquesta opció és molt complexa, s'haurà de moure en casos molt concrets quan es rep un missatge d'error indicant que la dimensió interna d'un array s'ha excedit.

S'aconsella deixar amb els valors que l'aplicació deixa per defecte i consultar l'ajuda per entendre que vol dir cadascuna d'elles.

### 3.6. DEFINICIÓ DE TEMPS DINS DEL SISTEMA ARENA.

Hi dediquem un apartat al temps, doncs simular es qüestió de temps i és bàsic entendre com tractar-lo dins del nostre procés de simulació.

El temps entre arribades de peces o el temps de retard causat per un procés, entre molts d'altres, poden ser definit segons:

1. **Constants.** El que normalment no es gaire vàlid per a un procés real, el qual es troba basat en esdeveniments discrets, es fixar un temps constant de simulació. Els esdeveniments discrets no es caracteritzen precisament per mantenir un valor constant durant tota la simulació, doncs la seqüència d'instantes en els quals l'estat del sistema pot presentar un canvi obeeix a un patró aleatori.

El que sí hem de tenir molt en compte són les constants emmagatzemades dins de les variables que l'entorn ens ofereix per defecte, doncs representaran una ajuda important alhora de plantejar els problemes a simular i moltes de les dades que s'hi guarden, tot i estar fixades quan les consultem, aniran variant de forma automàtica durant la simulació i estalviaran temps de programació del sistema.

Per exemple, si volem consultar el temps real de simulació en qualsevol moment, no hem de definir una variable que conti des de l'inici i acumuli el mateix. Sense realitzar cap programació especial, podem consultar simplement el temps real de simulació dins variable de sistema TNOW, el qual és emmagatzemat pel sistema de forma automàtica i dinàmica durant tota l'execució de la simulació.

2. **Una distribució estadística.** Es probablement el procés més utilitzat, cada distribució estadística oferida pel sistema serà capaç a partir de les dades històriques dels sistema real preses experimentalment, d'alimentar la simulació amb entitats que intentaran descriure el procés passat en el futur.

L'ajust de les dades passades a una distribució és el procés més crític de tota la simulació. Si volem que aquesta segueixi fidel a la realitat i ens doni els resultats a futur esperats, em d'encertar amb la distribució que generarà les entitats aleatòriament cap al sistema o allà on calgui generat elements aleatòriament.

Trobar la distribució és la clau per simular i ARENA porta incorporat un assistent per ajudar-nos en aquesta feina complexa, com ja he dit, vital per a la correcta simulació. El mateix és

denominat com ARENA Input Analyser i a través d'ell podem generar corbes aproximades de probabilitats.

El resultat del Input Analyser serà la distribució estadística que segueix aproximadament les dades d'entrada analitzades. Amb el resultat ja podem indicar, per exemple dins del mòdul CREATE quina serà la distribució a seguir per les entitats que aquest generarà en procés de simulació, les qual com és d'esperar seran ajustades a les dades històriques del procés sota estudi i permetran donar fiabilitat a les dades estadístiques que modelitzaran el procés a futur, un cop finalitzada la simulació.

### 3. Les expressions matemàtiques. Complexes de per si, son el complement ideal per a la programació avançada del sistema.

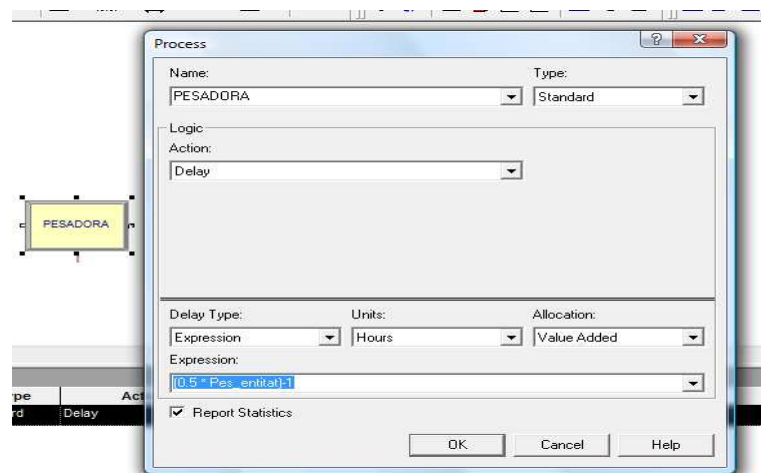
Permeten modelitzar processos complexos, doncs dins de les mateixes també podem utilitzar, constants, variable e inclús totes les distribucions estadístiques que ofereix ARENA per defecte.

Com exemple de quan hauríem d'utilitzar una expressió matemàtica, tenim la situació en la qual, per exemple, quan el temps de processament està relacionat d'alguna forma amb els atributs que porten incorporats les entitats.

Explicitant: Peces de diferent pes, porten incorporats temps diferents de procés, per a ser aixecades per un sistema de càrrega determinat. El cost d'aixecar ve donat per un temps d'espera de cada entitat PEÇA dins del mòdul **PROCESS**, denominat per exemple PESADORA.

Cada peça romandrà un temps d'espera diferent dins del mòdul PESADORA, modelitzant el procés de càrrega a partir de

l'atribut incorporat a l'entitat PEÇA denominat Pes\_entitat, mitjançant la introducció dins del camp Expression del mòdul Process de l'expressió  $(0.5 * \text{Pes\_entitat}) - 1$ . Aquesta serà doncs la expressió indicada pel camp "Delay Type", la qual tindrà hores



com a "Units" i es tindrà en compte els temps d'espera de cada entitat dins del costos. Aquests costos, es calculen automàticament, doncs cada entitat ja porta definit, prèviament, el que val cada hora d'espera seu al sistema, aquí només hem d'indicar que l'espera afegeix cost mitjançant el camp "Allocation", seleccionant "**Value Added**".

A continuació, degut a la complexitat que representa la seva assimilació, s'ha cregut oportú reproduir les ajudes que els diferents manuals consultats ofereixen, resumint les funcionalitats que estem estudiant dins d'aquest apartat.

Definir el temps, representa definir la simulació i els següents subapartats mostren una informació que haurà de ser d'obligada consulta per a tothom que s'introdueix en el món de la simulació.

La mateixa hauria de ser impresa a part, doncs servirà d'ajuda per al seguiment dels exemples vistos en la part pràctica del projecte, els quals donen sempre per entesa aquestes consideracions que deixen de ser prèvies i tothom ha de tenir a mà, abans, durant i al final de realitzar qualsevol projecte de simulació.



**3.6.1. VARIABLES I ATRIBUTS MÉS UTILITZATS EN L'ENTORN ARENA.**

<b>Expressions</b>		<b>Accés</b>	<b>Descripció</b>
<b>Denominació</b>	<b>Paràmetres</b>		
<b>ED</b>	Expression Number	<b>RO</b>	Valor d'una expressió especificat pel seu Expression Number.
<b>EXPR</b>	Expression Number [Index 1, Index 2]	<b>RO</b>	Valor d'una expressió especificat pel seu Expression Number i els seus índex.
Expression Name	[Index 1, Index 2]	<b>RO</b>	Valor d'una expressió especificat pel seu Expression Number i els seus índex.

<b>Variables en Temps de Simulació</b>		<b>Accés</b>	<b>Descripció</b>
<b>Denominació</b>	<b>Paràmetres</b>		
<b>TNOW</b>	-	<b>RW</b>	Variable que conté el temps real de simulació.
<b>TFIN</b>	-	<b>RW</b>	Variable que contindrà el temps final de simulació,

<b>Relacionats amb Entitats</b>		<b>Accés</b>	<b>Descripció</b>
<b>Denominació</b>	<b>Paràmetres</b>		
Attribute Name	(Index 1, Index 2)	<b>RW</b>	Valor de l'atribut que pertany a una entitat. Si es tracta d'un Array els paràmetres indiquen els seus índex.
Entity.Type	Entity Number	<b>RO</b>	Tipus o nom d'una determinada entitat.
Entity.Picture	Entity Number	<b>W</b>	Gràfic que identifica a l'entitat dins de l'animació.
Entity.SerialNumber	-	<b>RO</b>	Número identificador automàtic de l'entitat de només lectura.
Entity.Sequence	Entity Number	<b>RW</b>	Identifica a la propera estació on es transferirà l'entitat.
Entity.Jobstep	Entity Number	<b>RW</b>	Index dins de la seqüència de l'entitat.
Entity.CurrentStation	-	<b>RO</b>	Index de l'estació on es troba actualment l'entitat.
<b>AG</b>	Rank, Attribute Number	<b>RO</b>	Valor de l'atribut Attribute Number de l'entitat amb el Rang especificat dins del grup d'entitats actiu.
<b>NG</b>	Entity Number	<b>RO</b>	Número d'entitats dins del grup de l'entitat representativa.

<b>Variables d'us General</b>		<b>Accés</b>	<b>Descripció</b>
<b>Denominació</b>	<b>Paràmetres</b>		
Variable Name	[Index 1, Index 2]	<b>RW</b>	Valor d'una variable especificat pel seu Variable Name i els seus índex.
<b>NSTO</b>	Storage ID	<b>RO</b>	Nombre d'unitat emmagatzemades dins la unitat Storage ID.
<b>J:</b>	-	<b>RO</b>	Cerca l'índex d'una variable.
<b>TF</b>	Table ID, XValue	<b>RO</b>	Valor de la variable XValue guardada a la taula Table ID

<b>Variables de Cua</b>		<b>Accés</b>	<b>Descripció</b>
<b>Denominació</b>	<b>Paràmetres</b>		
<b>NQ</b>	Queue ID	<b>RO</b>	Nombre d'entitats dins de la cua Queue ID
<b>AQUE</b>	Queue ID, Rank, Attribute Number	<b>RO</b>	Valor de l'atribut indexat per Attribute Number de l'entitat en el Rang especificat dins la cua Queue ID

<b>Variables de Recursos</b>		<b>Accés</b>	<b>Descripció</b>
<b>Denominació</b>	<b>Paràmetres</b>		
<b>MR</b>	Resource ID	<b>RW</b>	Conté informació sobre les unitats de capacitat del recurs indexat pel seu Resource ID
<b>NR</b>	Resource ID	<b>RO</b>	Conté el nombre d'unitats ocupades pel recurs indexat pel seu Resource ID
<b>RESUTIL</b>	Resource ID	<b>RO</b>	Valor d'utilització instantània. Val 0 si NR=0; 1 si NR≥MR i NR/MR en qualsevol altre cas, sobre el valors de les unitats de capacitat i les ocupades del recurs indexat pel seu Resource ID
<b>STATE</b>	Resource ID	<b>RO</b>	Conté l'estat actual del recurs indexat pel seu Resource ID. Val -1 si es troba ocios; -2 si es troba Ocupat; -3 si es troba inactiu i -4 si es troba avariats.

<b>Variables Estadístiques</b>		<b>Accés</b>	<b>Descripció</b>
<b>Denominació</b>	<b>Paràmetres</b>		
<b>NC</b>	Counter ID	<b>RO</b>	Conté el valor actual del comptedor especificat pel seu Counter ID
<b>DAVG</b>	Dstat ID	<b>RO</b>	Conté el valor promig de l'estadístic especificat pel seu Dstat ID
<b>DMAX</b>	Dstat ID	<b>RO</b>	Conté el valor màxim de l'estadístic especificat pel seu Dstat ID
<b>DMIN</b>	Dstat ID	<b>RO</b>	Conté el valor mínim de l'estadístic especificat pel seu Dstat ID
<b>DSTD</b>	Dstat ID	<b>RO</b>	Conté la desviació estàndard de l'estadístic especificat pel seu Dstat ID
<b>DTPD</b>	Dstat ID	<b>RO</b>	Conté el valor per sobre del període de temps recol·lectat per l'estadístic especificat pel seu Dstat ID
<b>DHALF</b>	Dstat ID	<b>RO</b>	Conté el 95% de l'interval de confiança el valor de l'estadístic especificat pel seu Dstat ID
<b>DVALUE</b>	Dstat ID	<b>RO</b>	Conté el valor de l'últim valor guardat per l'estadístic especificat pel seu Dstat ID
<b>TAVG</b>	Tally ID	<b>RO</b>	Conté el valor promig del temps especificat pel seu Tally ID
<b>TMAX</b>	Tally ID	<b>RO</b>	Conté el valor màxim del temps especificat pel seu Tally ID
<b>TMIN</b>	Tally ID	<b>RO</b>	Conté el valor mínim del temps especificat pel seu Tally ID
<b>TNUM</b>	Tally ID	<b>RO</b>	Conté el nombre de valors del temps segons el seu Tally ID
<b>TSTD</b>	Tally ID	<b>RO</b>	Conté la desviació estàndard del temps segons el Tally ID
<b>THALF</b>	Tally ID	<b>RO</b>	Conté el 95% de l'interval de confiança segons el Tally ID
<b>TVALUE</b>	Tally ID	<b>RO</b>	Conté el valor de l'últim valor de temps segons Tally ID

<b>Variables de Transport</b>		<b>Accés</b>	<b>Descripció</b>
<b>Denominació</b>	<b>Paràmetres</b>		
<b>IT</b>	Transporter ID, Unit Number	<b>RO</b>	Indicador de l'estat del transport de la unitat especificada segons el seu Transporter ID . Val 0 si es troba ociós; 1 si es troba Ocupat i 2 si es troba inactiu.
<b>MT</b>	Transporter ID	<b>RO</b>	Nombre d'unitats actives en el transport especificat pel seu Transporter ID.
<b>NT</b>	Transporter ID	<b>RO</b>	Nombre d'unitats ocupades en el transport especificat pel seu Transporter ID.
<b>VT</b>	Transporter ID	<b>RO</b>	Conté la velocitat de totes les unitats en el transport especificat pel seu Transporter ID.

**3.6.2. DISTRIBUCIONS ESTADÍSTIQUES INCLOSES DINS L'ENTORN ARENA.**

<b>DISTRIBUCIÓ</b>		<b>Paràmetres</b>	<b>Aplicacions</b>
<b>Descripció</b>	<b>Abreviació</b>		
Beta	BETA	Beta, Alfa	Donada la seva habilitat per adaptar-se a un gran nombre de formes és molt utilitzada quan no es tenen dades. Per exemple per a representar proporcions aleatòries, com la proporció d'elements defectuosos dins d'una producció.
Continuous	CONT	CumP <sub>1</sub> , Val <sub>1</sub> , ..., Cump <sub>n</sub> , Val <sub>n</sub>	Per agregar de forma directa al model de dades actuals variables aleatòries contínues.
Discrete	DISC	CumP <sub>1</sub> , Val <sub>1</sub> , ..., Cump <sub>n</sub> , Val <sub>n</sub>	Per assignar una variable o atribut a un conjunt de valors que es basen en probabilitats. Per exemple, la fórmula DISCRETE(0.25, 1, 0.6, 2, 1, 3) significa que el 25% pertany a 1, el 35% (35%+25%=60%) pertany a 2 i el 40% restant, pertany a 3.
Erlang	ERLA	Mitjana, K	S'utilitza en situacions en les quals les activitats prenen lloc en fases successives i cada fase té una distribució exponencial. Per a un número molt gran de K, Erlang s'aproxima a una Normal. Per exemple per representar el temps que es requereix per completar una tasca.
Exponential	EXPO	Mitjana	S'utilitza per a modelar els temps entre esdeveniments d'arribades a l'atzar però no és adequada per a modelar temps de retard entre processos. En el mòdul Create ARENA dins la opció d'horari (Schedule) extrau automàticament una mostra d'una distribució exponencial amb una mitjana que canvia segons l'horari definit. Això és útil per exemple en aplicacions de serveis en les quals el volum dels clients varia durant el dia.
Gamma	GAMM	Beta, Alfa	Per a valors enters de Alfa, la distribució Gamma és una Erlang. S'utilitza amb la finalitat de representar el temps que es requereix per a completar qualsevol treball.
Johnson	JOHN	Gamma, Delta, Lambda, Xi	La flexibilitat d'aquesta funció permet ajustar molts conjunts de dades. El paràmetre Gamma defineix la forma, Delta defineix si es vol treballar amb límits si es > 1 i si és <1 es treballarà sense límits o sigui amb el zero al centre.

<b>DISTRIBUCIÓ</b>		<b>Paràmetres</b>	<b>Aplicacions</b>
<b>Descripció</b>	<b>Abreviació</b>		
LogNormal	LOGN	Mitjana, LogStd	S'utilitza quan la quantitat és el producte d'un gran nombre de quantitats aleatòries. També s'utilitza freqüentment per a representar els temps de treball d'una distribució segada a la dreta.
Normal	NORM	Mitjana, Desviació Estàndard	S'aplica en situacions en les quals s'aplica el Teorema Central del Límit. També s'utilitza empíricament en molts processos que aparenten tenir una distribució simètrica. Només es pot utilitzar per a quantitats positives doncs el seu rang es troba entre infinits.
Poisson	POIS	Mitjana	S'utilitza per a modelar un nombre aleatori d'esdeveniments que succeeixen dins d'un interval de temps determinat. Si el temps entre esdeveniments es distribueix exponencialment aleshores el nombre d'esdeveniments que ocorren en un determinat interval de temps té una distribució de Poisson. També s'utilitza per a modelar agrupacions de mida variable.
Triangular	TRIA	Mínim, Mitjana, Màxim	S'utilitza en situacions en les quals es desconeix la forma exacta de la distribució, però s'estima mitjançant els valors mínims, màxims i la moda que si es troben disponibles.
Uniform	UNIF	Mínim, Màxim	S'utilitza quan es considera que tots els valors dins d'un rang finit son iguals entre si o quan només es té el rang.
Weibull	WEIB	Beta, Alfa	Es àmpliament utilitzada quan la fiabilitat del model representa el temps de vida d'un dispositiu. Aleshores vol dir, que tenim multitud de parts que poden fallar independentment i quan qualsevol dels seus components falla el sistema també.

Per a introduir una distribució a ARENA, en el cas que no aparegui en la llista desplegable, s'ha d'introduir l'abreviatura corresponent a cada distribució seguida dels paràmetres entre parèntesis.

Es poden deixar espais entre els paràmetres per tal de facilitar-ne la seva lectura.

### 3.6.3. ARENA INPUT ANALYZER.

L'analitzador de dades d'entrada o Input Analyzer representa una eina molt potent la qual es troba dins l'ambient ARENA. Es pot utilitzar per a determinar quina distribució de probabilitat s'ajusta a les dades d'entrada i també per ajustar una distribució específica a les dades amb la finalitat de comparar diferents funcions de distribució o de visualitzar els efectes del canvis en els paràmetres d'una mateixa distribució. A més, el Input Analyzer pot generar grups de números aleatoris que es poden analitzar a través de les funcions d'ajust del programa.

Les dades processades mitjançant el Input Analyzer, en general, representen intervals de temps associats amb un event de duració aleatòria. Per exemple, es pot utilitzar aquesta eina per analitzar un conjunt de dades que representin el temps entre arribades, el temps de procés o el temps entre falles successives del sistema, entre d'altres aspectes.

#### 1. ENTRADA AL INPUT ANALYZER.

Dins ARENA es pot activar aquesta eina a través de la barra de menús, seleccionant Tools y, posteriorment, Input Analyzer.

També es pot cridar a aquesta eina sense obrir ARENA, podem cridar l'executable de l'aplicació denominat "Input" directament, doncs el mateix es troba a la mateixa carpeta que el programa en el directori per defecte de instal·lació.

#### 2. PREPARACIO DELS ARXIUS DE DADES.

Les dades amb les quals treballa el INPUT ANALYZER provenen generalment de medicions de temps, per la qual cosa aquestes han de ser primer recopilades dins d'un arxiu. Aquest arxiu es pot crear introduint les dades dins d'una fulla de càlcul (Excel, per exemple) o amb un processador de text (Word, Wordpad, Notepad, guardat com a document pla de text).

Suposem que les dades següents son el resultat d'un estudi de temps que es realitza en un banc i que representen el temps d'atenció d'un client en una caixa determinada.

5.4 5.0 6.3 5.7 6.1 5.2 6.9 5.8 4.9 5.7 4.9 5.8 6.2  
6.3 5.9 5.7 4.8 5.2 6.4 6.2 5.8 5.7 6.3 4.8 5.7 5.5

Es vol saber quin tipus de distribució de probabilitat s'ajusta millor a aquestes dades. Això es particularment necessari si es pretén incloure-les com un temps de procés dins d'un model de simulació

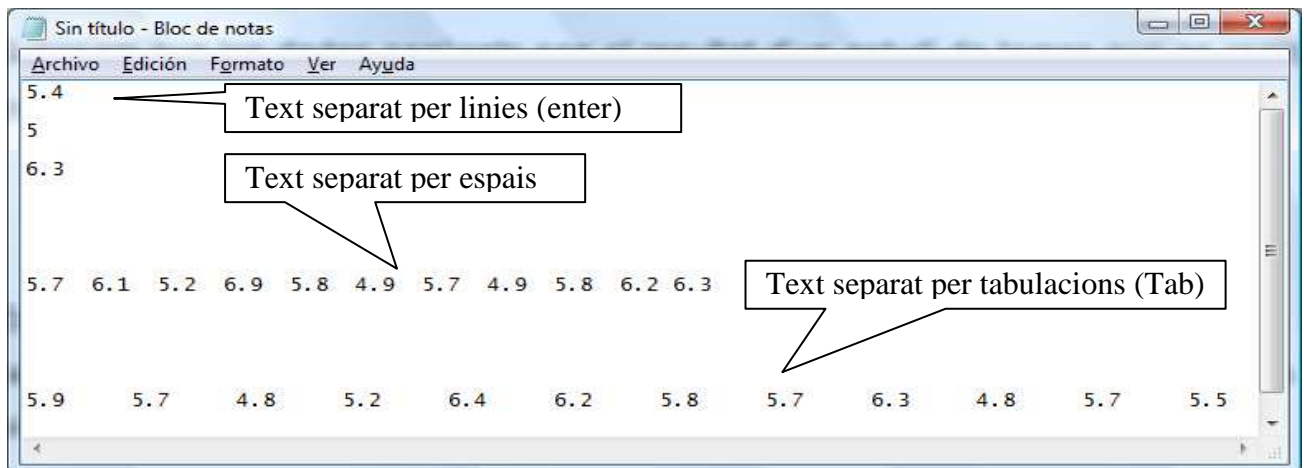
### 3. INTRODUCCIO DADES D'ENTRADA MITJANÇANT UN PROCESSADOR DE TEXT.

Quan es treballa amb un processador de text s'han de separar les dades mitjançant un o més caràcters d'espai en blanc, com espais, tabulacions i línies, els quals poden anar barrejats dins del mateix document.

Una vegada s'han entrat les dades segons el format escollit, es procedeix a donar a l'arxiu, el nom i la extensió adient. Els requeriments del Input analyzer per a reconèixer un arxiu de dades son els següents:

- Extensió .dst
- Arxiu de només text

Com a exemple, obrirem el Notepad introduint les dades següents:



L'arxiu el podem desar amb el nom de prova. A efectes de reconeixement en el Input Analyzer, dins del quadre de diàleg Nom de l'arxiu es col·loca prova.dst i es guarda sota la opció per defecte de document de text.

### 4. INTRODUCCIO DADES D'ENTRADA MITJANÇANT UNA FULLA DE CÀLCUL.

Les dades s'entren per files sota una mateixa columna de forma simple, però alhora de guardar l'arxiu ho hem de fer segons els requeriments del Input Analyzer:

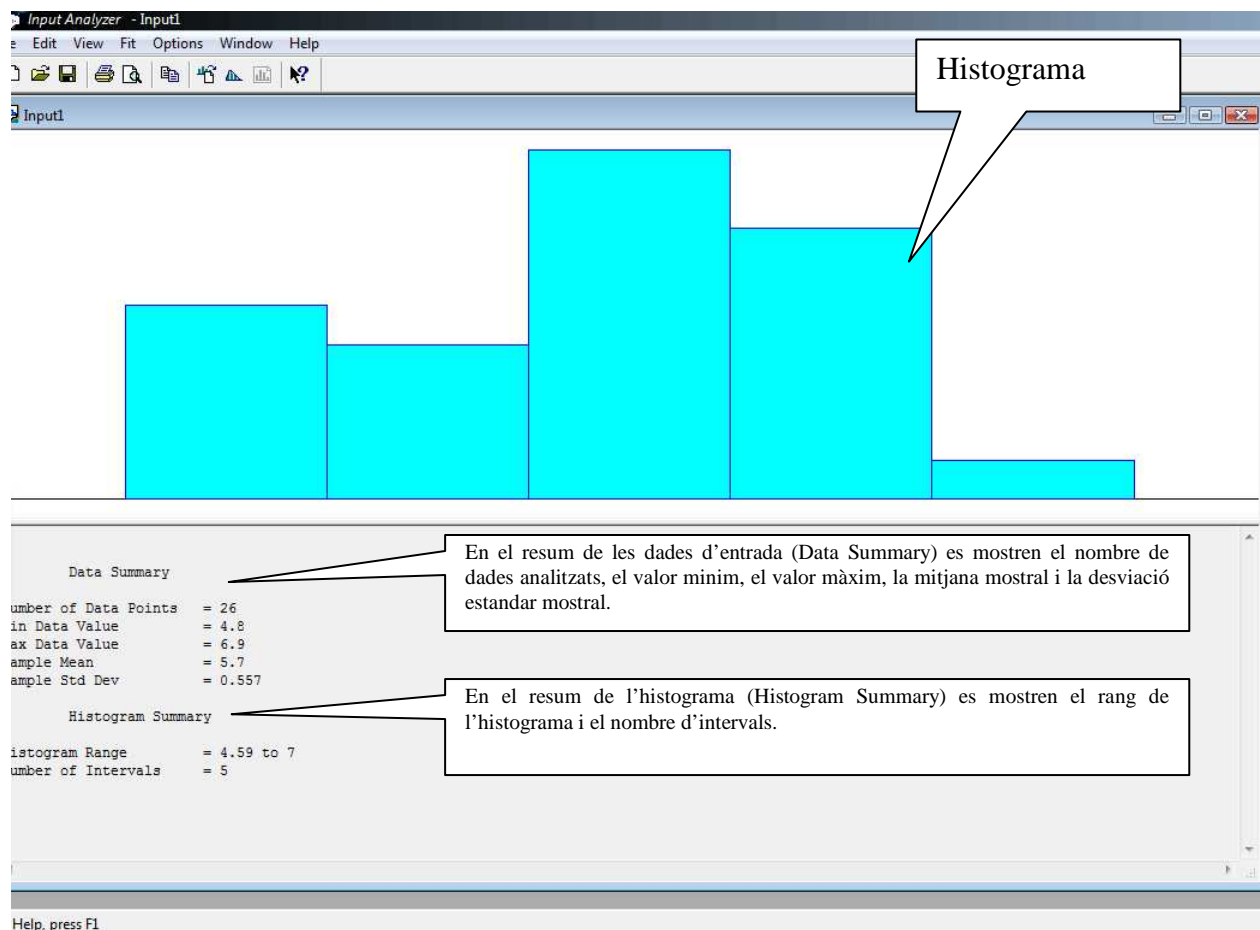
- Guardar com a Text (delimitat per tabulacions) amb l'extensió .dst.
- Els arxius que es generen, tant en fulla de càlcul com amb processador de text, no han de tenir cap tipus de capçalera. Només han d'incloure exclusivament dades.
- Les porcions decimals han de ser separades per punts i no per comes.
- L'arxiu no pot tenir cap tipus de caràcter diferent a número i punt decimal.

## 5. CREACIO D'UN ARXIU D'ENTRADA (INPUT FILE).

Per a crear un arxiu d'entrada es selecciona, de la barra de menús de la finestra del Input Analyzer, la opció File/New o s'utilitza el teclat "ctrl. + n" o premem la icona que representa un nou arxiu a la barra de treball estàndard del Input Analyzer.

Posteriorment a que ens apareix-hi un requadre en blanc, procedirem a cercar les dades d'origen des de l'arxiu on es troben les dades que es volen analitzar. Per això seleccionem des de la barra del menú principal : File -> Data File -> Use Existing, d'aquesta manera podem navegar cap al directori on tenim guardat el nostre arxiu proba.dst.

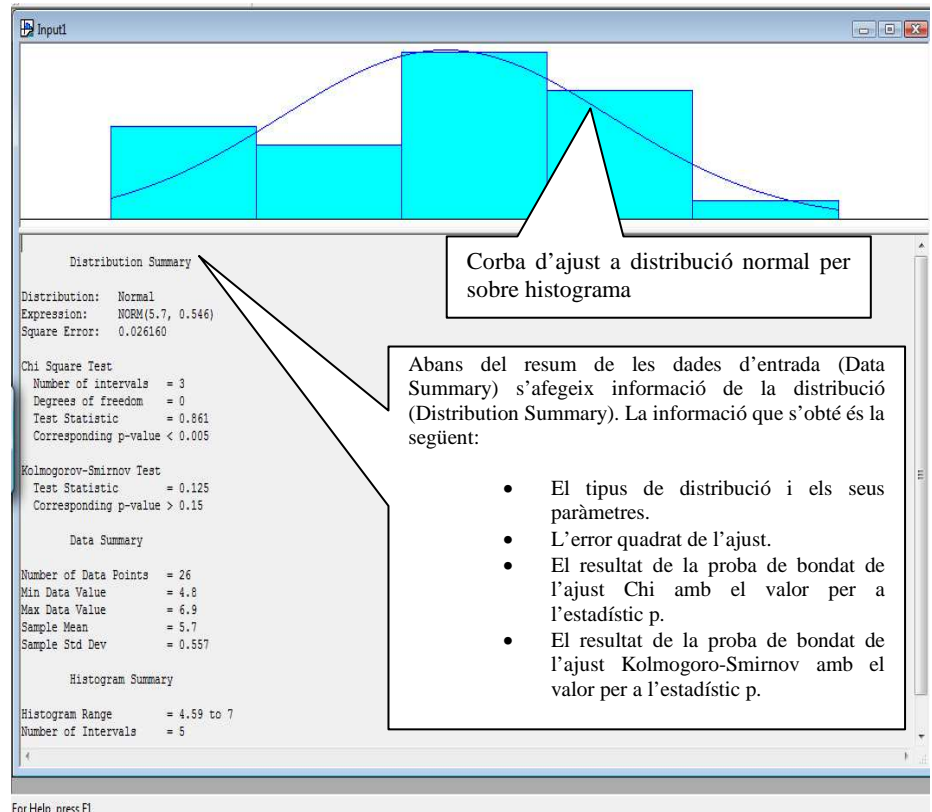
L'acció anterior permet que l' obrir l'arxiu .dst el Input Analyzer mostri automàticament d'histograma de les dades tan gràfic com analíticament.





## 6. AJUST D'UNA DISTRIBUCIÓ ESPECIFICA A LES DADES D'ENTRADA.

Després de carregar l'arxiu i veure el seu histograma, el proper pas és el d'ajustar la distribució. Per a realitzar aquesta operació es selecciona, mitjançant la barra de menús, la opció "Fit", d'aquesta manera es desplega una llista amb les



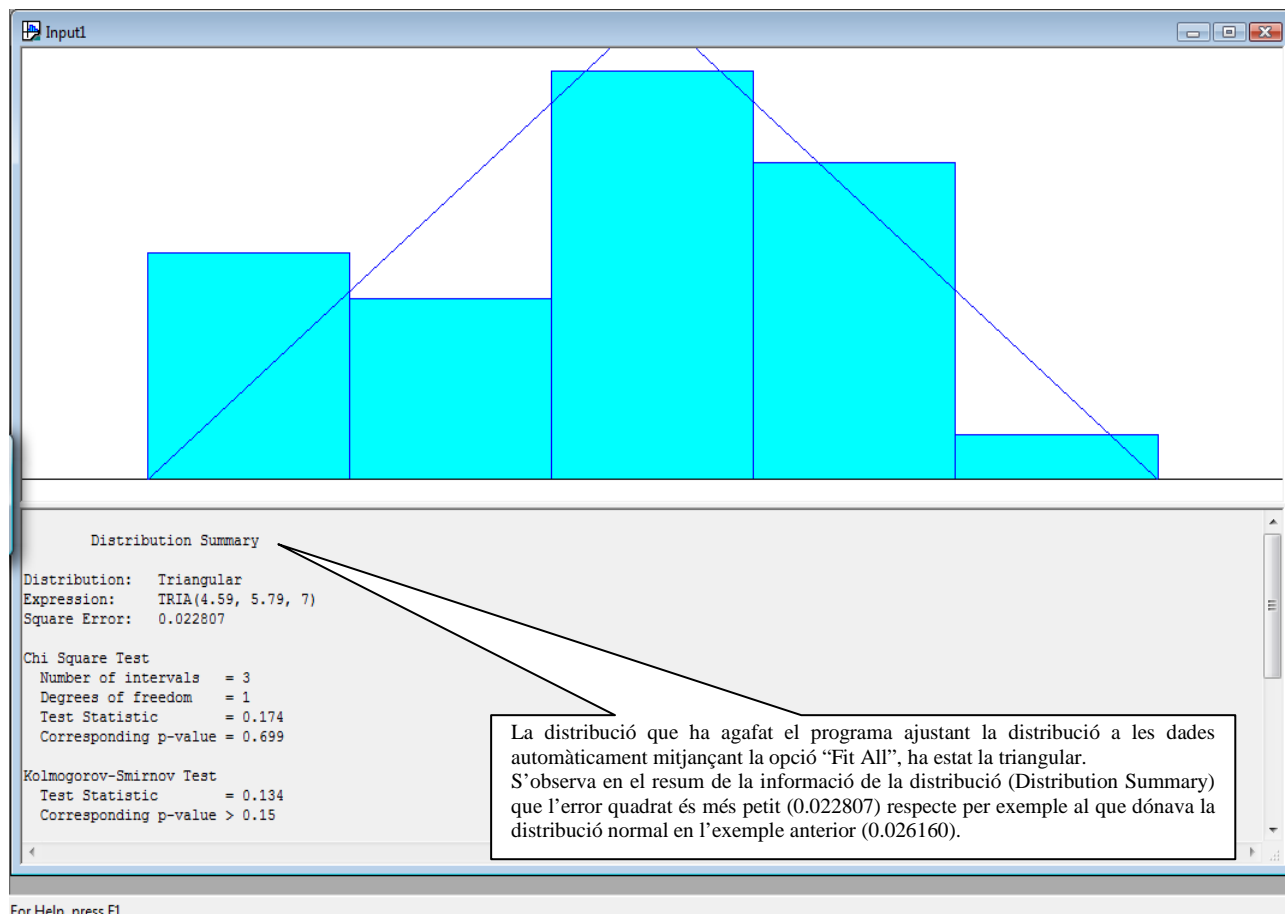
a les qual Input Analyzer considera que es poden ajustar a les dades.

Per exemple es pot donar el cas que no s'activin algunes distribucions per les dades introduïdes, com es el cas de la distribució Poisson, la qual no s'activarà quan es detectin valors sencers.

Seguint el nostre exemple, seleccionarem l'ajust per a una distribució normal i apareixerà dins l'histograma la corba d'ajust i en la part analítica, es mostraran els paràmetres per aquesta distribució.

## 7. SELECCIÓ DEL MILLOR AJUST.

El Input Analyzer te la opció "Fit All" dins del menú "Fit", que permet ajustar totes les distribucions amb las que compta aquesta eina actives per a les dades analitzades.

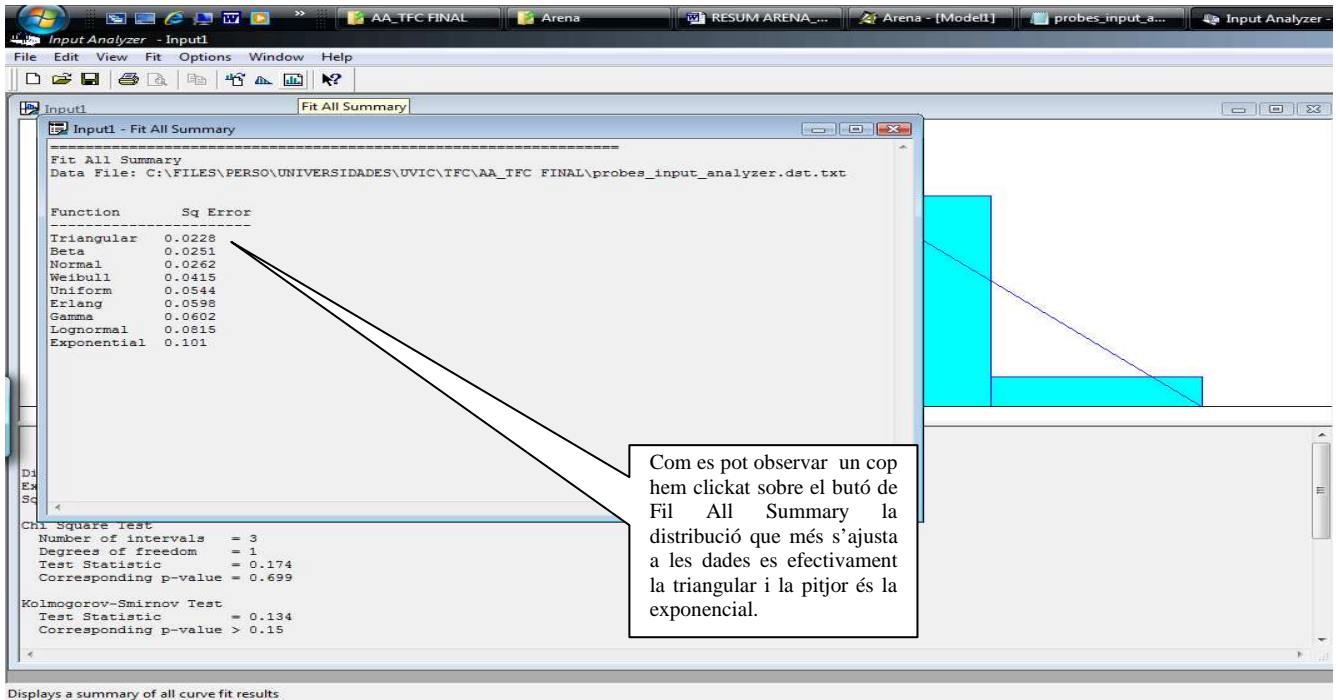


Com apunt important direm que arribats a aquest punt, les dades de l'apartat "Expression" de l'informe "Distribution Summary" per aquesta distribució ajustada a les dades són TRIA(4.59, 5.79, 7). Aquestes son les dades que transcriurem per exemple a un bloc CREATE, dins del camp "Entities per Arrival" un cop seleccionat que el "Type" del temps per arribades s'introduirà mitjançant una "Expression".

Això permetrà generar una seqüència de números aleatoris

segons la distribució de les dades històriques analitzades ajustades a una distribució triangular de mínim 0.5, mitjana de 0.75 i màxim de 1.

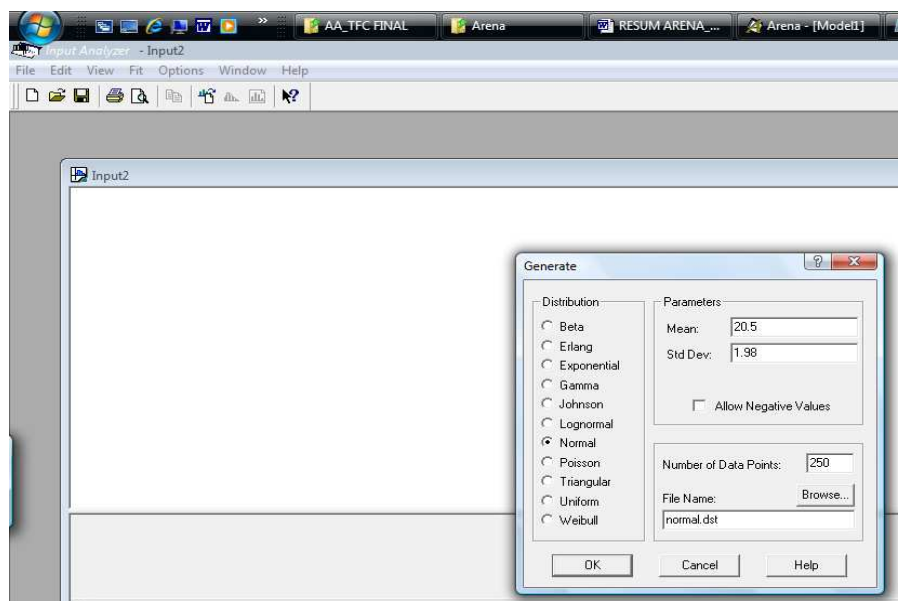
Fent-ho mitjançant Input Analyzer el procés és més llarga, però tindrem moltes més probabilitats d'èxit en la fiabilitat de les dades de la simulació, que si només col·loquem les distribucions a l'atzar obviant aquestes proves prèvies.



També podem seleccionar automàticament aquella distribució que ens doni un error més petit en les probes de bondat de cada ajust. Per a fer-ho disposem del resum dels errors de les probes d'ajust amb cadascuna de les distribucions si seleccionem la opció "Fit all summary" de la barra d'eines estàndard.

### 8. GENERACIÓ DE DADES.

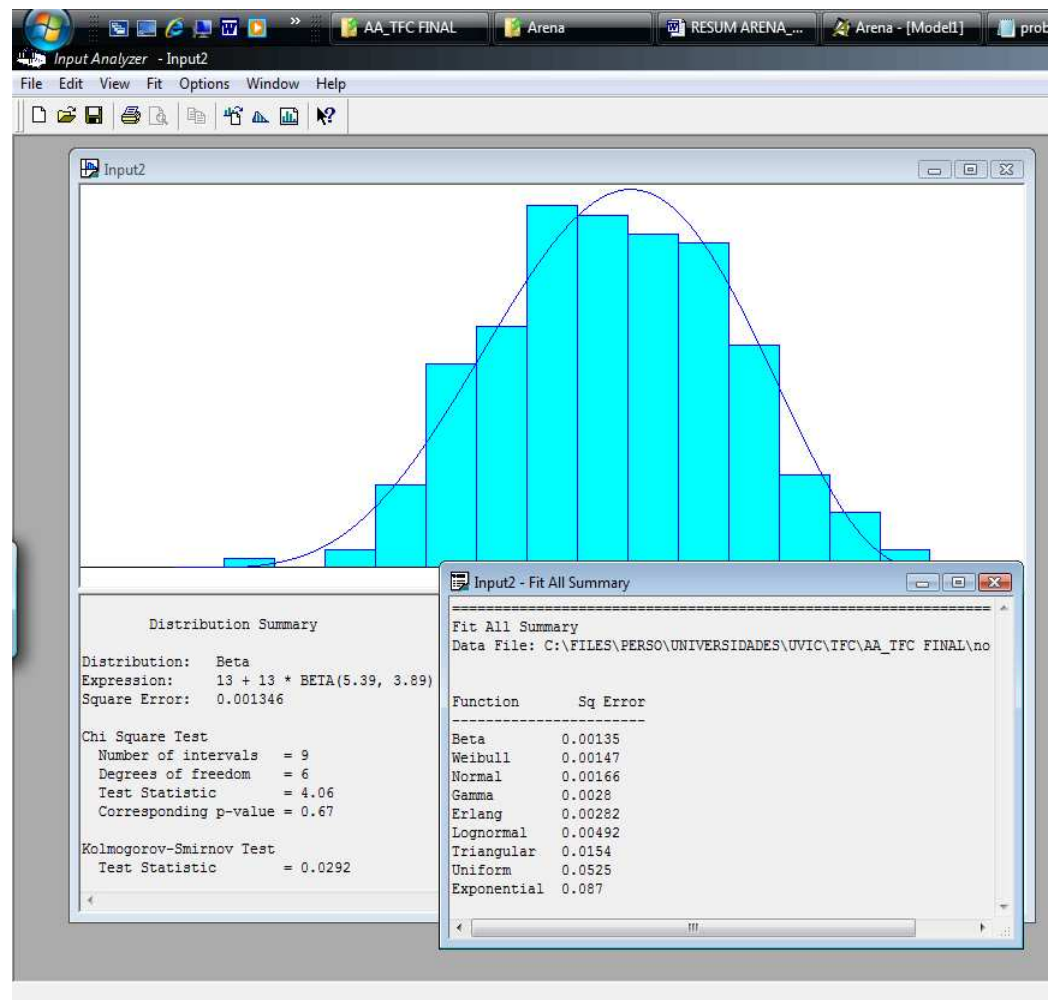
Es poden crear arxius de dades artificials, és a dir, que ni vinguin de la pràctica si no que siguin generats automàticament per el Input Analyzer. Aquesta operació es realitza creant un nou arxiu d'entrada i seleccionant la opció "File -> Data File -> Generate New". A continuació es mostra el tipus de diàleg per al qual es pot escollir el tipus de distribució que es desitja generar, així com els seus paràmetres.



També podem escollir si volem o no generar nombres negatius marcant o no la opció “Allow Negative Values”.

A modus de exemple, es generaran 250 dades que segueixen una distribució normal amb mitjana de 20.5 i desviació estàndard de 1.98.

Per a veure les dades generades, es busca l'arxiu de sortida, anomenat en el nostre exemple normal.dst i el podem obrir directament des de un editor de text com el Notepad o importar-lo cap a Excel com a text delimitat per tabulacions.



A continuació es

mostra la finestra del Input Analyzer per a les dades generades.

Com a dada curiosa observem que prement la opció de “Fit All” i mostrant el “Fit All Summary”, el propi Input Analyzer que ha generat les dades com a una normal les ajusta cap a una distribució Beta en primer lloc, a una Weibull en segon i la Normal apareix en tercera posició.

Mitjançant aquest exemple es pot veure la complexitat de l'ajust de les dades i la criticitat del mateix.

## 9. OPCIONS DEL HISTOGRAMA.

Algunes característiques de l' histograma que proporciona el Input Analyzer, seleccionant a la barra de menús “Options -> Parameters -> Histogram”.

Podem variar característiques de com el nombre d'intervals, el límit inferior i el límit superior, es poden variar segons certes condicions, les quals es mostren a la taula següent:

<b>Característiques de l'</b>	<b>Condicció.</b>
Nombre d'interval·s	Ha d'estar entre 5 i 40
Límit inferior (Low Value)	Ha de ser més gran o igual a l'enter més gran que no sobrepassi el valor més petit de les dades analitzades.
Límit superior (High Value)	Ha de ser més petit o igual a l'enter més petit que iguali o sobrepassi el valor més gran de les dades analitzades

## 10. CANVIS EN LA DISTRIBUCIÓ.

Una vegada s'hagi ajustat la distribució mitjançant la opció "Fit" be sigui per un procés manual o automàtic, s'activa la opció de canvi de paràmetres de la mateixa.

Es poden canviar seleccionant en la barra de menús "Options -> Colors -> Distribution", però hem de tenir en compte que si es canvien el paràmetres de la distribució, s'ha de realitzar de nou la prova de bondat de l'ajust.

## 11. CANVIS EN LA PRESENTACIÓ.

Es possible modificar l' des de la barra de menú per millorar la impressió amb "Options -> Colors -> Background" per a canviar-li el color de fons; "... -> Histogram Fill" per modificar cada barra del mateix; "... -> Histogram Border" per variar el perfil de cada barra; amb "... -> Plot Line" podem variar el color de la corba d'ajust de la distribució.

Cal dir que moltes vegades resultat convenient portar la gràfica de l' a un document o a una fulla de càlcul. Per a fer-lo només hem de fer un clic sobre la gràfica i ja la tenim emmagatzemades al porta papers de Windows, mitjançant les tecles estàndard ctrl. + c la podem enganxar allà on ens calgui.

### 3.6.4. ARENA EXPRESSION BUILDER.

Amb la finalitat de realitzar les tasques més senzilles, ARENA incorpora un assistent denominat ARENA Expression Builder. S'accedeix a aquesta eina fent clic al botó dret del ratolí en tots aquells camps que son susceptibles de ser omplerts amb una expressió.

Per exemple, el camp Expression en un mòdul Decide, o Condition en un mòdul Hold, configurat com Scan for Condition.

The image shows the 'Hold' dialog box and the 'Expression Builder' window. The 'Hold' dialog has fields for Name (Hold 1), Type (Scan for Condition), Condition, Queue Type (Queue), and Queue Name (Hold 1.Queue). The 'Expression Builder' window has a tree view of categories, a keypad of operators, and a 'Current Expression' field.

**Fonts de les quals procedís la expressió que es desitja construir. Es poden desplegar les diferents opcions que tenen cadascuna de les categories.**

**Operadors matemàtics i lògics en forma de botns. Qualse s'els presiona, aquests operadors s'inclouen dins de l'expressió que s'està construint.**

**Area on s'ubicarà l'expressió. Per fer aparèixer l'assistant fer click al botó dret del ratolí dins l'area sombrejada.**

**Expressió actual. Mostra l'expressio que s'està construint.**

Wait for Value	Limit	Queue Type	Queue Name
1		Queue	Hold 1.Queue

Les expressions matemàtiques poden ser formades mitjançant combinacions d'enters, constants, variables atribut o distribucions aleatòries.

Les expressions condicionals poden ser formades i son avaluades com a expressions numèriques, a una condició certa se li assigna el valor numèric 1, mentre que si és falsa se li assigna un 0.

Les expressions numèriques son avaluades com a fals si el seu valor resultant és 0 i com a cert si n'és diferent de 0.

A continuació definim les principals fonts o elements que trobem dins del Expression Builder.

### **Basic Process Variables. Relacionades amb els següents elements.**

<b>Element</b>	<b>Exemples</b>
Entities	Costos, nombre d'entitats que entren, nombre d'entitats que surten, etc...
Process	Costos, nombre que entra al mòdul, nombre que surt, WIP, etc...
Queue	Nombre en cua, temps promig d'espera, suma dels atributs de les entitats a la cua, etc...
Resource	Costos, Estat (Ocupat, Ociós, Fallida) o Ús (Percentatge de temps ociós), capacitat activa, etc...
Schedule	Valor actual.
Set	Nombre de membres del grup, posició d'un membre dins d'un grup, recursos ocupats dins d'un Set, etc...
Variable	Valor actual, valor mínim i màxim, valor promig, etc...

### **Advanced Process Variables. Relacionades amb els següents elements.**

<b>Element</b>	<b>Exemples</b>
Advanced set	Nombre dins del grup, nombre de membres del grup.
Expression	Valor actual.
Stateset	Valor de l'estat.
Statistics	Counter (Valor actual, límit), Frequency (Nombre d'ocurrència, temps promig en categoria, etc...), Tally (Nombre d'observacions, valor promig, etc...), Time-Persistent (Mínim, màxim i promig, etc...).
Storage	Número emmagatzemat.

### **Advanced Transfer Variables. Relacionades amb els següents elements.**

<b>Element</b>	<b>Exemples</b>
Conveyor	Estat, longitud, velocitat, etc...
Distance	Distància entre estacions.
Sequence	Estació dins de la seqüència.
Transporter	Estat, velocitat, nombre d'entitats ocupades, etc...

## Random Distributions. Accés s les distribucions de probabilitat incorporades.

Dins d'aquesta categoria es disposa d'ajuda per a la construcció d'expressions relacionades amb les distribucions de probabilitat que incorpora ARENA per defecte. Per exemple, per a la distribució Normal l'assistent mostra els camps en els quals s'ha d'entrar la mitja i la desviació estàndard.

## Math Functions. Expressions matemàtiques definides per defecte.

Dins d'aquesta categoria es troben les 20 funcions matemàtiques definides dins ARENA per defecte. Cada funció incorpora una llista de paràmetres entre parèntesis. Aquests paràmetres poden ser especificats com a constants o expressions quan s'utilitzen dins d'un model. Les funcions geomètriques (ACOS, ASIN, ATAN, HCOS, HSIN, HTAN, COS, SIN, TAN) prenen els valors especificats en radians.

Llistat de funcions matemàtiques:

Function	Description
ABS( a )	Absolute value
ACOS( a )	Arc cosine
AINT( a )	Truncate
AMOD( a1 , a2 )	Real remainder, returns (a1-(AINT(a1/a2)*a2))
ANINT( a )	Round to nearest integer
ASIN( a )	Arc sine
ATAN( a )	Arc tangent
COS( a )	Cosine
EP( a )	Exponential (e <sup>a</sup> )
HCOS( a )	Hyperbolic cosine
HSIN( a )	Hyperbolic sine
HTAN( a )	Hyperbolic tangent
MN( a1 , a2 , ... )	Minimum value
MOD( a1 , a2 )	Integer remainder, same as AMOD except the arguments are truncated to integer values first
MX( a1 , a2 , ... )	Maximum value
LN( a )	Natural logarithm
LOG( a )	Common logarithm
SIN( a )	Sine
SQRT( a )	Square root
TAN( a )	Tangent



## Mathematical Operators & Logical Operators. Operadors bàsics i de comparació.

Els següents operadors es troben disponibles i les expressions son avaluades utilitzant la seva prioritats. Els parèntesi s'avaluaran primer i els operadors amb la mateixa prioritats s'avaluaran d'esquerra a dreta.

Operator	Operation	Priority
<b>Math Operators</b>		
**	Exponentiation	1 (highest)
/	Division	2
*	Multiplication	2
-	Subtraction	3
+	Addition	3
<b>Logical Operators</b>		
.EQ. , ==	Equality comparison	4
.NE. , <>	Non-equality comparison	4
.LT. , <	Less than comparison	4
.GT. , >	Greater than comparison	4
.LE. , <=	Less than or equal to comparison	4
.GE. , >=	Greater than or equal to comparison	4
.AND. , &&	Conjunction (and)	5
.OR. ,	Inclusive disjunction (or)	5

**Nota:** == és un operador lògic; = és un operador d'assignació. Utilitzi == per avaluar si dos elements tenen el mateix valor; utilitzi = per fixar un valor, igual que amb el mòdul Assign.

## Simulated Control Variables. Relacionades amb l'execució del model.

Element	Exemples
MREP	Nombre màxim de rèpliques del model que s'executaran. Es assignable per l'usuari *.
NREP	Nombre de la rèplica que s'executa en aquell instant. No és assignable per l'usuari.
TNOW	Temps que ha corregut des de l'inici de la rèplica actual. No és assignable per l'usuari
TFIN	Temps que manca per a la finalització de la rèplica actual. Es assignable per l'usuari *.

\* Els valors d'aquestes variables es poden modificar en temps de disseny a través del menú Run/Setup/Replications Parameters o en temps d'execució, incorporant l'expressió corresponent dins d'un mòdul Assign.

### **Entity-Related Variables.**

Es refereixen a les variables que estan relacionades amb l'entitat, com atributs, costos, temps i atributs definits per l'usuari.

També s'inclouen les relacions amb els grups o lots (entitats representatives d'un grup).

<b>Element</b>	<b>Exemples</b>
Atribut	Tipus d'entitat, número de sèrie, seqüència, pas dins de la seqüència.
Variables de grup	Nombre d'entitats dins del grup, suma d'atributs, etc...

### 3.7. MÒDULS BÀSICS DEL SISTEMA ARENA (Template Basic Process).

#### ▪ MÒDULS LÒGICS.

Formen part activa del model, descriuen la lògica del procés lo més visual possible. Son interconnectats entre ells formant una xarxa d'informació i comandes a través de la quals hi passaran les entitats al seu través, les quals patiran modificacions en els seus atributs en funció de la lògica simulada. Estan formats pels següents mòduls:

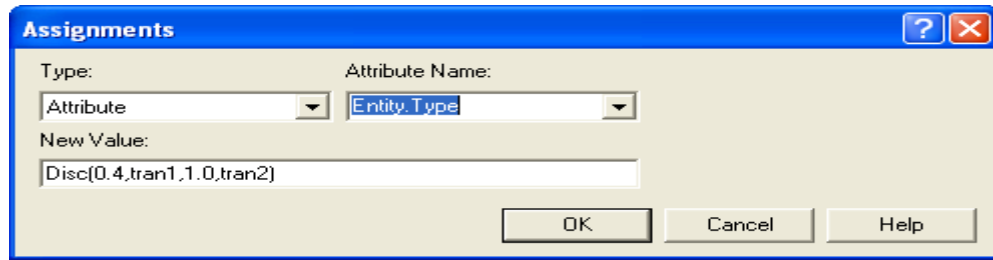
1. **CREATE:** En aquest mòdul s'assigna l'atribut **Entity Type**, el qual permetrà definir animació i costos inicials per a l'entitat. Mòdul que es pren com a punt de partida del model i on es creen les entitats que es processaran al sistema.
2. **DISPOSE:** Es col·loca al final del model i és on es procedeix a retirar l'entitat del model. Es recol·lecten les estadístiques corresponents a l'entitat.
3. **PROCESS:** A aquest mòdul les entitats que hi entren experimenten una operació que involucra la utilització del recurs, el retard que provoca el temps de processament així com la seva alliberació.

Aquest mòdul també pot complir la funció d'un sub model.

També és on s'especifica la **CATEGORIA DEL COST QUE PERTANY AL TEMPS D'UTILITZACIÓ: Valor agregat, No valor agregat, Transferència, Espera i Altres.**

4. **DECIDE:** Permet direccionar el flux d'entitats d'acord a una regla de decisió, la qual pot estar basada en: **una condició, una probabilitat o una expressió.**
5. **BATCH:** Permet formar lots o grups d'entitats d'un tamany qualsevol i prèviament definit. Poden ser permanents o temporals i de qualsevol entitat o d'un tipus específic. Quan es crea un lot s'obté una nova entitat que representa el grup format.
6. **SEPARATE:** Funció oposada al bloc Batch doncs separa els lots creats per aquesta. Aquest mòdul també es pot utilitzar per tal de fer diverses còpies d'una entitat.
7. **ASSIGN:** Canvia el valor d'una variable, atribut, figura, nivell, seqüència o qualsevol altre variable del sistema. Es possible fer diverses assignacions en un sol mòdul Assign.

**Exemple:** Dos tipus de client entren a un banc. El 40% retirarà diners a caixa (tran1) i la resta al caixer automàtic (tran2). Tenim una sol'entitat -> client, però per tal de diferenciar-les utilitzem els atributs, en l'exemple creem mitjançant Assign dues entitats noves tran1 i tran2 mitjançant l'expressió : Disc(0.4,tran1,1.0,tran2).



(1) Si aquest dos atributs venen d'un sol mòdul Create, compartiran la mateixa distribució d'arribades, però s'han de declarar dues noves entitats mitjançant el mòdul de dades Entity-Basic Process, **son entitats virtuals creades a partir d'atributs**.

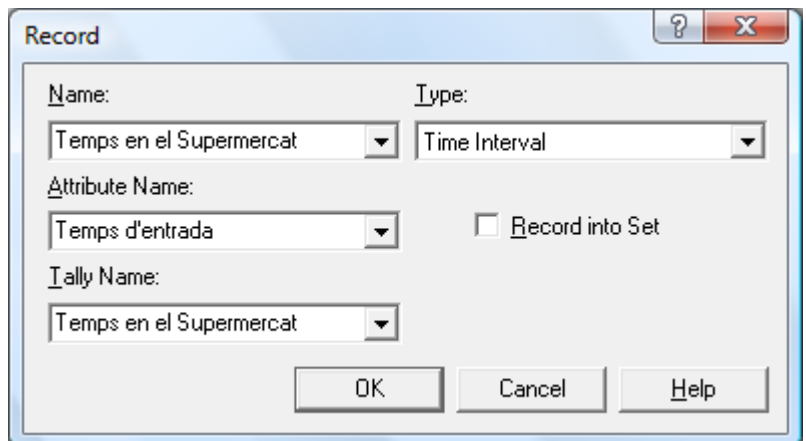
Entity - Basic Process							
	Entity Type	Initial Picture	Holding Cost / Hour	Initial VA Cost	Initial NVA Cost	Initial Waiting Cost	Initial Tran Cost
1	Cliente	Picture.Report	0.0	0.0	0.0	0.0	0.0
2	tran1	Picture.Report	0.0	0.0	0.0	0.0	0.0
3	tran2	Picture.Report	0.0	0.0	0.0	0.0	0.0

Double-click here to add a new row

(2) Si volem que tinguin distribucions independents tran1 i tran2 no caldrà fer Assign d'una entitat existent, si cadascuna ha de tenir la seva pròpia distribució es necessari un mòdul Create per cadascuna.

8. **RECORD:** S'utilitza per tal de recol·lectar estadístiques en el model de simulació. També es pot emprar com a comptedor.

Per exemple, si volem saber quan triga un consumidor des de el moment en que entra a un supermercat fins a l'hora en que realment arriba a la caixa. En el moment en el que entra l'entitat al sistema li assignem el temps actual de simulació (TNOW) dins un atribut i un cop que l'entitat hagi passat per tots els processos necessaris, un cop arribi a caixa, la fem entrar dins el bloc "**RECORD**".



Aquest bloc calcularà la diferencia entre el temps anterior i el nou temps actual de simulació i grava el resultat dins d'una variable del tipus "**Tally**", la qual serà mostrada en l'informe final.

Per altre banda també podríem trobar per exemple la taxa d'arribades d'entitat a un cert procés. El qual es tan simple com entrar dins del mòdul "Record" i posar-hi els paràmetres que mostra la figura:



## ▪ MÒDULS DE DADES

Venen en forma de fulles de treball, no apareixen directament en el model gràfic doncs no tenen caràcter actiu dins la simulació. Són editats a través de formularis i la seva funció és la declaració de les especificacions per a cada element de flux, fixant els seus valors inicials i les seves propietats. Formats pels següents mòduls:

1. **ENTITY:** En aquesta fulla de treball es defineixen, per a les diferents categories, l'atribut Entity Type, la primera animació que se li assignarà a cada entitat creada i els seus respectius costos inicials.
2. **QUEUE:** En aquesta fulla de treball veurem les diferents cues de treball creades automàticament amb el mòdul lògic Process o manualment a través d'ella. També podem veure el tipus de regla d'alliberació que segueix cada cua (FIFO, LIFO, etc).
3. **RESOURCE:** Els recursos que s'utilitzaran es declaren en aquest mòdul, on també es defineixen les seves característiques com la seva capacitat, el cost d'operació i les falles.

Un molt bon exemple de la utilització d'aquest mòdul de dades es troba molt detallada dins del projecte anterior <sup>[9]</sup> si es fa la seva pràctica 6, anomenada assemblatge i prova de peces electròniques (pàgina 79).

4. **VARIABLES:** Dins d'aquesta fulla es defineixen els valors inicials de les variables emprades dins del model i, en el cas de les assignacions la seva dimensió.
5. **SCHEDULE:** Dins d'aquesta fulla ARENA ofereix la possibilitat d'establir un patró de canvis en la capacitat d'un recurs.

Amb base temps, els canvis de capacitat poden tenir una durada constant o aleatòria i es repeteixen periòdicament en el model. També s'ofereixen opcions de com en gestionen els canvis en la capacitat un cop aquest succeeixin (apropiatu, espera o ignorància).

Només es permet un horari per recurs, per exemple, capacitat per a processar 5 unitats per 6 hores, 3 unitats per 2 hores, 1 unitat per 1 hora, etc.

Aquesta part es troba molt detallada dins del projecte anterior <sup>[9]</sup> si es fa la seva pràctica 6, anomenada assemblatge i prova de peces electròniques (pàgina 79).

- 6. SETS:** S'utilitza quan es requereix crear grups repetitius de recursos, figures, cues, comptadors, estadístiques, etc., amb la finalitat de facilitar el modelatge d'un sistema determinat.

### 3.8. MÒDULS AVANÇATS DEL SISTEMA ARENA (Template Advanced Process).

#### ▪ MÒDULS LÒGICS.

Formen part activa del model, moltes de les característiques d'aquest "template" ja es troben integrades dins del "template" bàsic, com els blocs "Process" o "Assign". Però com el seu propi nom diu, aquest "template" ha de ser utilitzat principalment per a realitzar un modelatge més detallat i poder recollir dades més específiques o identificar els problemes més subtils de la planta.

Formen part del "template Advanced Process" i estan formats pels següents mòduls:

1. **DELAY:** Aquest mòdul retarda una entitat una quantitat de temps especificat. Quan una entitat entra en aquest mòdul, una expressió "time delay expression" és avaluada i delimita el seu temps de permanència dins del mòdul.

El temps de retard es pot comptabilitzar en forma de "value added" en el cas de ser un procés o quelcom que augmenti el valor de l'entitat, "non value added" en el cas d' un temps de refredament, assecat de peces o qualsevol altre activitat que no afegeixi valor, "transfer" per al temps dedicat al transport de l'entitat d'un procés a altre, "wait" per a qualsevol temps d'espera, o també "other" en el cas del que l'endarreriment no encaixi en qualsevol de les categories anteriors.

2. **DROPOFF:** Aquest mòdul retira un nombre específic d'entitats d'un grup i les envia cap a una altra mòdul, segons lo especificat per la connexió gràfica. Els mateixos poden ser atributs específics de les entitats o totes elles segons la regla especificada dins del mòdul.

La variable NG (Nombre d'entitats dins del grup de l'entitat representativa) es pot utilitzar per tal d'avaluar el tamany del grup de reunió de l'entitat actual. Aquest valor pot ser utilitzat per tal de retirar totes les entitats del grup i mantenir encara l'entitat representativa al sistema per a qualsevol altre direcció.

La diferència entre utilitzar aquest bloc i el mòdul bàsic lògic "Separate" té aplicació quan després del segon s' utilitza la opció "Split". Aleshores totes les entitats originals son separades del grup per a ser utilitzades individualment i es perd el valor de l'entitat representativa del grup. Si utilitzem "Dropoff" permet seleccionar quins membres del grup seran separats i en el cas que es remoguin tots mantenir l'entitat representativa del grup. Encara que algunes entitats continuïn agrupades després de "Dropoff" sempre poden ser posteriorment utilitzant també el bloc "Separate".

En el cas que passi una entitat amb un tamany de grup zero, el mòdul “Dropoff” simplement la ignorarà i la deixarà passar pel punt de sortida original.

3. **HOLD:** En entrar dins d'aquest mòdul una entitat serà col·locada dins d'una cua en espera d'una senyal (Wait for signal), per tal d'esperar que una determinada condició es torni verdadera (Scan for condition) per a ser alliberada o també pot esperar indefinidament (Infinite hold).

En l'últim cas, posteriorment s'haurà d'utilitzar un bloc “Remove” per tal que segueixi el seu procés cap a d'altres mòduls.

Si una entitat està esperant una senyal (Wait for signal), també es pot utilitzar un mòdul “Signal” per donar-li l'ordre d'alliberament cap a una altra mòdul. El codi d'alliberament pot ser diferent per a cada entitat si utilitzem variables en lloc de constants i a través d'aquesta senyal, també podem determinar el nombre d'entitats que alliberarem quan es rebí l'ordre que s'espera.

Quan una entitat entra en aquest mòdul i el mateix està configurat com “Scan for Conditions” i no es troba cap entitat a la cua, la condició s'avalua immediatament per a veure si es certa.

En aquest cas s'allibera acte seguit, però en canvi si la condició es falsa, l'entitat romandrà a la cua fins que la condició sigui vàlida. En canvi, quan una entitat arriba a un bloc on ja hi ha entitats esperant, es col·loca en espera de ser servida mitjançant una política FIFO. El valor zero retorna fals i qualsevol altre valor retorna verdader quan s'avaluen expressions numèriques, en canvi, si la condició està basada en un atribut, la condició canvia cada vegada que arriba una altra entitat.

4. **MATCH:** Aquest mòdul sincronitza un nombre específic d'entitats, fent esperar unes i altres en diferents cues. Per tal que es realitzi l'alliberació hi ha d'haver com a mínim una entitat en cadascuna de les cues, valor que s'especifica dins del camp “Number to Match).

També es pot especificar un atribut, aleshores la sincronització es durà a terme quan hi hagi a la cua el nombre d'entitats especificades amb el mateix valor per aquell atribut, abans d'iniciar-se la sincronització mitjançant l'alliberament d'una entitat de cada cua.

Aquest mòdul també pot utilitzar-se en seqüència amb un mòdul bàsic Batch per a sincronitzar números d'entitats diferents per cada cua. Per exemple, si volem sincronitzar dues peces tipus A amb una de tipus B, primer agrupem les A mitjançant un mòdul Batch i les sincronitzem amb les B mitjançant el mòdul “Match”, per tornar a separar les A amb un bloc “Separate” i obtenir la sincronització desitjada.

5. **PICKUP:** Si volem retirar un número d'entitats consecutives d'una cua però començant des de un lloc específic de la cua necessitarem utilitzar aquest bloc.



Les entitats retirades de la cua d'entrada son adicionades al final del grup d'entitats que estan passant pel bloc "Pickup".

Si no volem tenir un error de simulació, la quantitat d'entitats a ser retirades de la cua no pot superar el nombre d'elements que hi hagin a la mateixa. Mitjançant un mòdul "Decide" podem avaluar el valor de la variable NQ (Nombre d'elements a la cua) per tal d'evitar errors d'aquest tipus. Utilitzant la variable NG (Nombre d'entitats dins del grup de l'entitat representativa) podem verificar el tamany del grup representat per l'entitat que passa en aquell instant pel bloc "Pickup".

6. **ReadWrite:** Mitjançant aquest mòdul es possible llegir dades d'un arxiu d'entrada, o des del teclat i assignar els valors de les dades dins d'una llista de variables o atributs. També es possible escriure dades en una sortida especificada, com un arxiu o a la pantalla.

Quan una entitat arriba a aquest mòdul, l'arxiu especificat és examinat per a verificar si es troba obert i en cas negatiu l'arxiu és obert automàticament.

Els valors dels atributs, variables o expressions ("other") llistats son llegits o escrits d'acord a un format especificat prèviament.

7. **RELEASE:** S'utilitza per alliberar unitats d'un recurs del qual s'havia prèviament apoderat una entitat. Aquests mòdul es pot utilitzar per alliberar recursos individuals o en conjunt i per a cada recurs alliberat es poden especificar el seu nom o la quantitat.

Qualsevol entitat esperant a una cua per aquell recurs alliberat agafarà el seu control immediatament i mitjançant la variable de sistema "NR" es pot consultar el nombre actualitzat d'unitats ocupades pel recurs especificat. Quan una entitat executa un mòdul "Release", "NR" es decrementada segons la quantitat alliberada, a menys que els recurs sigui immediatament reservat per una altra entitat.

Obtindrem un error de simulació si s'alliberen més unitats d'un recurs de les que prèviament s'han reservat i l'alliberació múltiple d'un recurs serà feta en l'ordre en el que apareixen dins el mòdul "Release".

8. **REMOVE:** Quan una entitat arriba al mòdul "Remove", aquesta retira una única entitat d'una posició específica dins d'una cua especificada i l'envia al mòdul connectat a continuació. L'entitat que ha causat la retirada segueix i es processada abans que les entitats que provenen de la cua.

La ultima entitat de la cua pot retirar-se especificant el lloc de la mateixa mitjançant la paraula clau "NQ" o com a variable "NQ.Nom\_de\_la\_cua".

Abans d'enviar una entitat cap a un mòdul "Remove" s'ha de verificar que la mateixa existeix dins de la cua mitjançant el mòdul avançat "Search". El mateix omple una variable global J

amb el lloc de l'entitat desitjada, o amb el valor 0 si cap entitat de la cua té les característiques desitjades per a ser remoguda de la cua.

Obtindrem un error de simulació si el mòdul "Remove" intenta remoure entitats amb una posició superior al tamany de la cua.

No existeix la possibilitat de remoure entitats que es trobin dins de cues externes.

- 9. SEIZE:** Aquest mòdul assigna unitats d'un o més recursos a una entitat determinada. Es pot utilitzar per assignar unitats d'un recurs determinat, un membre d'un conjunt de recursos o un recurs definit per un mètode alternatiu, com poden ser un atribut o una expressió.

Quan una entitat entra dins d'aquest mòdul, s'espera en una cua específica fins que tots els recursos sol·licitats estiguin simultàniament disponibles. També s'ha d'especificar el tipus d'assignació per a l'ús del recurs.

Entitats que reserven el recurs amb valors de prioritats més baixos tenen prioritat sobre aquelles amb valors més alts. Nombres negatius son transformats en 0 i si vèries entitats intenten reservar el mateix recurs i tenen la mateixa prioritat, les entitats amb major temps d'espera rebran els recursos sol·licitats.

- 10. SEARCH:** Realitza una recerca dins d'una cua, un bloc (Batch) o una expressió per tal de trobar la posició (Rank) d'una entitat.

Quan es busca dins de cues o blocs, el valor de la variable global J rep la posició (Rank) de la primera entitat que satisfà la condició de recerca o el valor 0 si la recerca ha estat negativa.

Anàlogament quan es busca mitjançant expressions, la variable global J rep el primer valor que satisfà la condició especificada o 0 si cap valor de J dins d'un interval especificat satisfà la condició de recerca.

Quan una entitat arriba a un mòdul "Search", J rep el valor d'inici (starting índex) i la condició de recerca és validada. Si al finalitzar la recerca es manté el valor de J vol dir que la recerca ha tingut èxit, en cas contrari el valor de J serà incrementat o decrementat i la condició tornarà a ser avaluada. Aquest procés es repeteix fins que es satisfaci la condició de recerca o el comptedor intern del mòdul hagi arribat al valor de la variable global J.

En cas que no existeixen entitats a la cua, al bloc o no s'hagi satisfet la condició de recerca, J rebrà el valor de 0.

- 11. SIGNAL:** Aquest mòdul envia una senyal per a tots els mòduls del tipus "Hold" que estiguin configurats com "Wait for Signal", alliberant el major nombre especificat d'entitats.

Quan una entitat entra dins del mòdul, "SignalValue" és avaluat i el codi de la senyal és enviat. En aquest moment les entitats presents en els mòduls "Hold" que estaven esperant la

mateixa senyal son retirades de les seves respectives cues. L'entitat que ha originat la senyal continua el seu procés fins que trobi una espera, una cua o sigui retirada del sistema.

- 12. STORE:** Si volem omplir un lloc d'emmagatzement d'entitats les hi adicionarem mitjançant aquest mòdul i les retirarem posteriorment mitjançant el mòdul "Unstore".

Quan una entitat arriba a aquest mòdul, l'emmagatzement en qüestió és incrementat i l'entitat segueix cap al mòdul següent.

"Storages" son utilitzats per mostrar animacions d'una entitat, mostren el temps que aquestes han estat aturades en algun punt del sistema i posteriorment són processades per d'altres mòduls. També es poden recollir estadístiques vers al nombre d'entitats emmagatzemades.

Mitjançant la variable "NSTO" podem comptar el nombre d'entitats en un magatzem especificat. Si aquesta variable creix indefinidament durant la simulació, és possible que no hem col·locat posteriorment un bloc "Unstore" i les entitats no s'estiguin retirant del magatzem.

- 13. UNSTORE:** Retira entitats d'un magatzem determinat. Quan una entitat arriba a aquest mòdul l'emmagatzement en qüestió es decrementa i l'entitat continua fins el proper mòdul.

Un altre condició sinequanum és que tota entitat que arriba a aquest mòdul ha d'haver passat prèviament per un mòdul "Store", la qual cosa, en cas que no es produeixi serà avisat per pantalla.

Totes les entitats han de ser retirades de tots els magatzems abans de sortir del sistema a través d'un mòdul "Dispose".

- 14. ADJUST VARIABLE:** Aquest mòdul ajusta el valor d'una variable a un valor objectiu determinat i pot ser utilitzar amb la opció "Rotate By Expression" en imatges i recursos globals per a presentar les rotacions d'animació de les imatges. Per exemple, rotar una imatge 180° a una velocitat de 5° per segon.

També es pot utilitzar per aproximar o animar un continu augment o disminució d'una variable en el temps.

Quan una entitat entra dins d'aquest mòdul, el nom de la variable s'ajusta al valor objectiu especificat i l'entitat es retinguda en el mòduls fins que l'ajust s'ha completat.

El valor "Update Interval" especifica l' de temps entre les assignacions efectives del valor de la variable durant el període de canvi. Un interval d'actualització petit produeix una animació més suau en les imatges que giren segons la referència d'aquesta variable, així com estadístiques més acurades persistents en el temps i recollides per la variable. En canvi un interval d'actualització més gran produeix majors velocitats d'execució.

- **MÒDULS DE DADES.**

Venen en forma de fulles de treball, no apareixen directament en el model gràfic doncs no tenen caràcter actiu dins la simulació. Son editats a través de formularis i la seva funció és la declaració de les especificacions per a cada element de flux, fixant els seus valors inicials i les seves propietats. Formen part del “template Advanced Process” i estan formats pels següents mòduls:

**15. ADVANCED SET:** Al igual que el mòdul SET del panell de processos bàsics, aquest mòdul permet prendre grup o arregles d'elements, en aquest cas més avançats, com grups de cues o seqüències i, en general, de qualsevol tipus.

**16. EXPRESSION:** Emmagatzema expressions que poden ser utilitzades o cridades des de qualsevol part del model. La utilitat de les expressions radica en que aquestes poden tenir operacions (suma, productes, etc..) o distribucions de probabilitat (normal, exponencial, etc). Les expressions es construïran mitjançant l'assistent denominat Expression Builder, comentat dins l'apartat 1.3.4 anterior.

**17. FAILURE:** En aquest mòdul es defineixen les característiques d'una fallida o parada d'un recurs. S'especifica el tipus de fallida (per conteig o per temps), el temps de reparació, les unitats del tipus de reparació, etc. Aquestes fallides es referencien en el mòdul de dades del recurs que es troba dins del panell de processos bàsics.

**18. FILE:** Dins d'aquest mòdul de dades es defineixen els arxius amb els quals es treballarà dins del model. Es el cas d'escriptura, en aquest mòdul s'assigna el nom i la extensió que porta l'arxiu que es genera al sistema operatiu.

Els arxius d'escriptura i lectura que s'utilitzaran dins del model s'hauran d'emmagatzemar en el mateix directori.

**19. STATESET:** En aquest mòdul es creen grups d'estat de recursos definits per l'usuari, o es canvien els noms dels estats predeterminats o automàtics (Busy, Idle, Inactive i Failed). Idle pots ser renombrat com Ociós i Busy com Ocupat, opció que s'ha de programar dins d'aquest mòdul.

En casos més avançats poden haver diferents estats definits per l'usuari, per exemple, allistament, muntatge i tall per a un mateix recurs, dins de l'estat automàtic Busy.

**20. STATISTIC:** S'encarrega d'especificar estadístiques definides per l'usuari respecte a entitats o variables del model.

Mitjançant aquest mòdul de dades es poden generar estadístiques sobre la utilització d'un determinar recurs, promitjos ponderats en el temps d'una variable, intervals de temps per a entitats, entre d'altres aspectes.

Aquesta part es troba molt detallada dins del projecte anterior <sup>[9]</sup> si es fa la seva pràctica 6, anomenada assemblatge i prova de peces electròniques (pàgina 79).

Aquest mòdul també dóna la possibilitat d'escriure estadístiques recol·lectades dins d'un arxiu amb format especial, el qual es pot analitzar mitjançant la eina ARENA OUTPUT ANALYZER, la qual serà analitzada dins l'apartat 4.7.2.

**21. STORAGE:** Defineix els llocs en els quals es fixa l'animació de les entitats o emmagatzenaments.

Els emmagatzenaments es representen mitjançant una espècie de fila, però a diferència d'aquesta, la representació de l'entitat apareix només quan aquesta es troba realitzant diferents operacions. Dins d'un emmagatzement es poden tenir estadístiques, com per exemple, el nombre d'entitats en un moment determinat.

### 3.9. TRACTAMENT DE FALLIDES.

En alguns processos industrials es necessari tenir en compte les variacions que es poden presentar davant les fallides (**FAILURES**) de les màquines.

**ARENA** entén com a una fallida qualsevol alteració que es presenti dins del normal funcionament d'un recurs, per exemple, la vida útil d'una determinada peça en una màquina que genera la interrupció del servei en un caixer d'un banc.

Per introduir les fallides és molt important tenir en compte la seqüència demanada per la seva programació:

1. **Donar d'alta el model d'avaria dins del mòdul de dades avançat FAILURE.** En el qual s'especifica un nom diferents per a cada fallida a tractar dins del model i també :
  - i. El tipus de fallida (**Type**), el qual pot ser:
    - **Count.** L'avaria succeeix en funció del nombre de peces processades.
    - **Time.** L'avaria succeeix en funció a un temps establert.
  - ii. El temps que la màquina estarà treballant (**Up Time**) i les seves unitats. (**Type=Time**).
  - iii. Els temps que es trigarà en solucionar l'avaria un cop s'aturi la màquina (**Down Time**).

Failure - Advanced Process							
	Name	Type	Up Time	Up Time Units	Down Time	Down Time Units	Uptime in this State only
1	Avaries Embalatge	Time	WEIB( 5 , 1 )	Minutes	NORM( 5 , 1 )	Minutes	

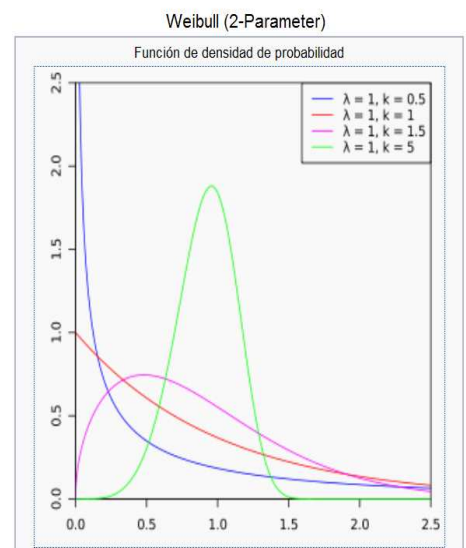
Double-click here to add a new row.

En el cas d'escollir el tipus **Time**, de totes les distribucions estadístiques la que s'aproxima més quan la fiabilitat del model representa el temps de vida d'un dispositiu és la distribució Weibull amb els seus dos paràmetres Beta i Alfa. Modelitza el sistema amb una molt bona aproximació quan les causes de fallida son independents i qualsevol d'elles pot fer aturar el sistema.

El paràmetre Beta també s'anomena **k** i és el paràmetre de forma. El mateix té diferents lectures en funció del seu valor:

1. Si  $k < 1$  vol dir que la taxa de fallides decreix amb el temps.
2. Si  $k = 0$  la taxa de fallides es constant en el temps.
3. Si  $k > 1$  vol dir que la taxa de fallides augmenta amb el temps.

El paràmetre Alfa s'anomena mitjançant la seva lletra grega  $\lambda$  i és el paràmetre d'escala de la distribució.



2. Fer referència el model d'avaria creat en el pas anterior, afegint el seu nom dins del camp **Failure** de tots els recursos especificats dins del mòdul de dades bàsic **RESOURCE**.

Resource - Basic Process										
	Name	Type	Capacity	Schedule Name	Schedule Rule	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures
1	Operari 1	Based on Schedule	Pla recuperacio	Pla recuperacio	Ignore	0.0	0.0	0.0		0 rows
2	Envasadora	Fixed Capacity	1	1	Wait	0.0	0.0	0.0		1 rows

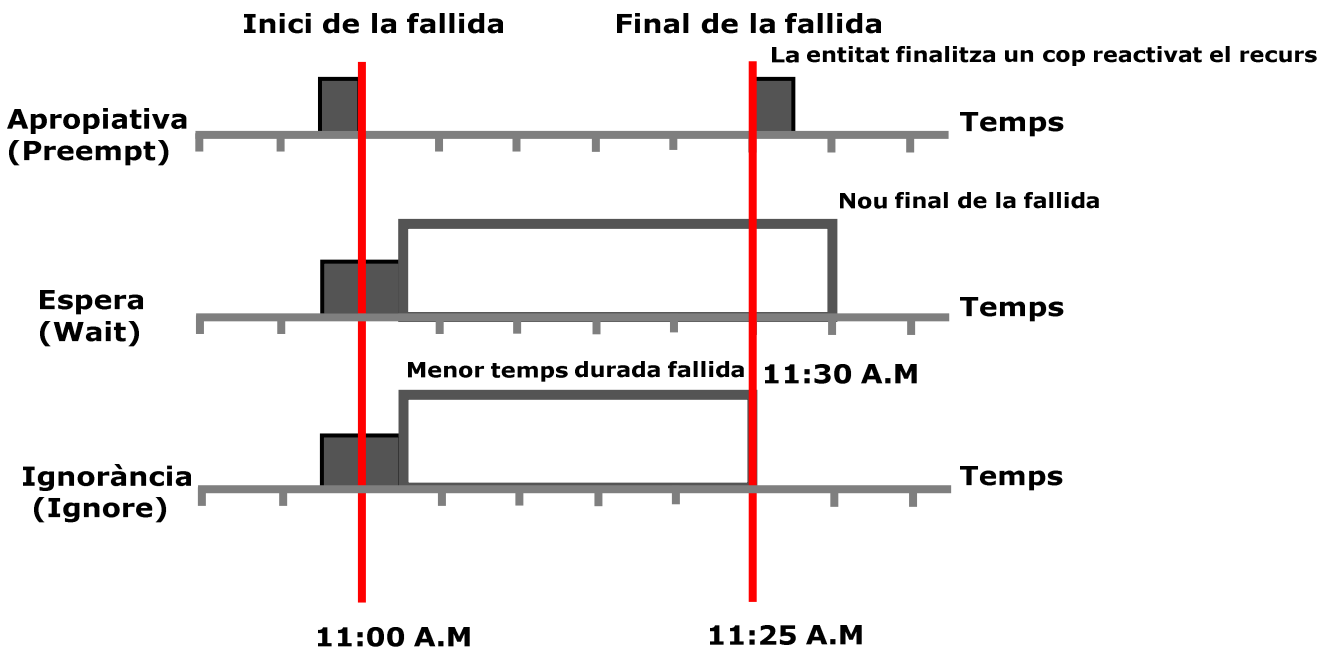
Failures		
	Failure Name	Failure Rule
1	Avaries Embalatge	Wait

El tractament de la fallida (**Failure Rule**) compleix la mateixa regla que per als horaris (apropiativa, espera e ignorància) i el recurs pot ser inhabilitat per complet, si el seu funcionament es basa en conteig d'unitats o temps decrementalment, fins que la seva capacitat es 0.

Un mateix recurs pot tenir múltiples fallides, per exemple, canvi d'eina cada 100 peces, 15 minuts de descans de l'operari cada dues hores, etc...

Una fallida es pot presentar durant el processat d'una peça, quan això passa ARENA contempla varies possibilitats.

A continuació es presenta un breu diagrama on es mostren les tres possibilitats de tractament de les fallides i tot seguit detallarem la descripció de cadascuna d'elles.



- 1. Preempt (Apropiativa).** La fallida succeeix immediatament. Si una entitat es troba en procés, aquesta roman inacabada i finalitzarà quan el recurs es torni a reactivar.
- 2. Wait (Espera).** Si en el moment d'ocórrer la fallida, una entitat es troba en procés, aquesta finalitza la seva operació i el temps de durada de la fallida romandrà constant. Amb la qual cosa el temps total de fallida s'augmentarà en funció de la durada del procés en execució.
- 3. Ignore (Ignorància).** La fallida té lloc instantàniament i si es troba una entitat en procés, aquest finalitza la seva operació i el seu temps de finalització es resta del temps total de fallida. Això fa que el temps de duració de la fallida, en algunes ocasions, sigui més petit que el valor especificat.

Aquest apartat s'ha introduït en aquest projecte com a resum breu, doncs el mateix queda molt ben resumit dins del projecte anterior, *Estudi i guia docent per a modelatge i simulació de sistemes mitjançant l'entorn ARENA* <sup>[9]</sup>, si es fa la seva pràctica 6, anomenada assemblatge i prova de peces electròniques (pàgina 83).



## **4. METODOLOGIA DE SIMULACIÓ.**

**Dins el procés de simulació amb ARENA, bàsicament:**

- 1. Creem entitats.**
- 2. Utilitzem recursos del sistema mitjançant les entitats.**
- 3. Distribuïm el flux de les entitats a través del model unitàriament o en grups.**
- 4. Analitzem els costos del sistema.**
- 5. Animem la simulació.**
- 6. Generem i analitzem reports.**
- 7. Utilitzarem també d'altres eines i processos avançats.**

## 4.1. CREACIÓ D'ENTITATS.

Aquestes es creen bàsicament mitjançant el mòdul lògic **CREATE**. Cal tenir en compte:

- a) Com a tipus d'entitat s'assigna per defecte l'atribut *Entity.Type*, el mateix s'utilitza per definir animacions i costos inicials.
- b) Tipus d'arribades de les entitats.
  - **Random:** Aleatori. Es refereix a una distribució exponencial, el camp value s'omple amb el valor de la mitja de la distribució en segons, minuts, hores o dies.
  - **Schedule:** Planificat segons el nom de l'element schedule que conté la programació d'arribades.
  - **Constant:** El temps entre arribades és un temps constant introduït segons valor numèric en segons, minuts, hores o dies.
  - **Expression:** Segons una distribució. Exemple: **NORM(5.89,0.65)**.
  - **Altres:** També es poden assignar el número d'entitats que arribaran per a cada event d'arribada, el temps que triga en arribar la primera entitat i el número màxim d'arribades, un cop arribat el mateix no es crearan més entitats.

## 4.2. UTILITZACIÓ DE RECURSOS MITJANÇANT LES ENTITATS.

Els recursos a priori tenen capacitat finita fins al final de la simulació i passen per diferents estats durant la seva execució: **OCUPAT, OCIÓS, INACTIU o AVARIAT**. Però un mateix recurs pot ser demanat per moltes entitats simultàniament, aleshores hem de considerar aquests punts:

a) Una entitat que sol·licita un recurs:

- **Seize**-> Pren el control del recurs si aquest es troba disponible.
- **Queue**-> Si el recurs no es troba disponible, espera en la cua associada al recurs.

b) Una entitat que té el control del recurs:

- **Release**-> El recurs s'allibera quan el deixa d'utilitzar. Li dona pas a una altra entitat que es troba en la cua d'espera del recurs.
- El recurs pot continuar efectuant processos que siguin necessaris inclòs en d'altres processos en paral·lel, fins a finalitzar el seu cicle en aquest recurs i alliberar-lo amb Release.

c) Tipus d'accions dins d'un recurs, mòdul PROCESS dins la macro Logic:

**7. Delay:** Durant el temps especificat l'entitat fa un retard però sense ocupar recursos.

**8. Seize, Delay:** L'entitat ocupa un recurs, fa un retard, però no allibera el recurs.

**9. Seize, Delay, Release:** L'entitat ocupa un recurs, fa un retard i allibera el recurs.

**10. Delay, Release:** Una entitat que prèviament ha pres el control d'un recurs, fa un retard i allibera el recurs.

d) Prioritats d'un recurs, mòdul PROCESS dins la macro Logic:

El camp **Priority** s'utilitza quan diverses entitats demanen el mateix recurs en diferents parts del model. Les entitats amb valors de prioritats baixos s'atenen primer que aquelles que tenen valors de prioritats alts.

e) Tipus de retards d'un recurs, mòdul PROCESS opció Delay Type:

Aquí s'ofereix la possibilitat d'utilitzar una expressió o una constant. Per menú podem configurar directament que treballi amb una **constant**, amb les distribucions **Normal, Triangular o Uniform** o dins la opció del desplegable trobem també - **Expression**, on es despleguen la resta de les distribucions que es poden emprar i on estan incloses les anteriors, però s'han de definir com a funció en el camp de text Expression.

### 4.3. DISTRIBUCIÓ DEL FLUX DE LES ENTITATS A TRAVÉS DEL MODEL.

En ocasions es fa necessari modelar situacions en les quals es tenen diverses entitats en un mateix procés, cadascuna de les quals amb característiques definides que determinen la seva pertinença a un grup específic.

Per exemple:

- a) Dos tipus de clients entren a un banc. Uns volen treure diners i els altres volen fer pagaments a compte.
- b) Tres tipus de clients arriben a una benzinera. Alguns arriben per a fer benzina, d'altres per rentar el cotxe i d'altres per comprar a la botiga queviures.
- c) En una empresa productora d'ordinadors, abans del seu assemblatge, les diferents parts són entitats diferents: plaques base, disc dur i pantalla entre d'altres.

Per a diferenciar les entitats es fan servir els atributs. Quan se li assigna un atribut específic a una certa proporció de les entitats que arriben al sistema, s'està creant un grup especial d'aquestes.

Els atributs poden ser definits per l'usuari però ARENA per defecte associa a cada entitat creada l'atribut Entity Type, el valor del mateix pot ser modificat per l'usuari tantes vegades com vulgui i aquest és el punt clau a tenir en compte alhora de fer el seguiment de la simulació i en el posterior anàlisi dels resultats.

#### 4.3.1. Distribució del flux de les entitats a través del model unitàriament.

Com s'ha comentat anteriorment mitjançant el mòdul lògic bàsic **Assign** es pot canviar el valor d'una variable, atribut, figura, nivell, seqüència o qualsevol altra variable del sistema. Això dona diferents possibilitats de distribució de les entitats pel model mitjançant un altre mòdul lògic bàsic, l'anomenat **Decide** el qual les pot distribuir de tres maneres:

1. Divisió del flux d'entitats basat en el seu tipus, mòdul DECIDE.

Aquí s'utilitzarà com a base de divisió els atributs de cada entitat si es fa servir la opció **2-way by Condition** per avaluar una opció vertader o fals o bé N-way by conditions amb lo que tindrem múltiples opcions de divisió.

2. Divisió del flux d'entitats segons percentatges, mòdul DECIDE.

En aquest cas simplement la base de divisió és percentual sense tenir en compte l'atribut amb les mateixes opcions anteriors.

### 3. Divisió del flux d'entitats basat en una condició del sistema, mòdul DECIDE.

En aquesta tercera divisió es contempla el cas que el flux d'entitats es vegi afectat per un condició del propi sistema.

Aquesta condició es representa mitjançant una variable i el seu valor numèric simbolitza els diferents estats que pot experimentar el sistema.

En aquest cal que seleccionem dins de la opció “**If**” del mòdul Decide el terme variable o expression. Si per exemple escollim **expression**, poder fer la divisió del flux de dades amb la opció **2-way by Condition i** per exemple si volem dividir quan el nombre d'elements d'una cua és inferior a 4, utilitzarem l'expressió “**NQ(caixers.queue)<4**” dins el camp **Value**.

#### 4.3.2. Distribució del flux de les entitats a través del model en grup.

En processos industrials en els quals treballem amb grans volums de producció és força comú la utilització de lots o grups de productes. El tamany del grup format depèn dels requeriments de les operacions que s'han de realitzar. Per exemple, podem tenir un tamany de lot per als productes inspeccionats, una altra per la transferència entre màquines o per a la venda o sortida.

Els lots construïts poden ser reduïts de nou a membres individuals per tal de processar individualment les entitats que els formen.

ARENA permet formar lots mitjançant el mòdul bàsic vist anteriorment denominat **Batch** i per desfer-los s'utilitza el bloc també bàsic **Separate**.

Cal tenir-ne algunes consideracions alhora de crear lots, com poden ser:

#### a) Creació de lots permanents o temporals, mòdul BATCH.

Cal tenir en compte que els únics lots que es podran desfer son els temporals.

#### b) Error en temps d'execució.

Un lot temporal no pot sortir mai del sistema, s'ha de separar o bé transformar en lot permanent si no volem tenir aquest tipus d'error.

#### c) Les entitats que entrin al bloc BATCH no surten d'ell fins que no es formi la totalitat del lot.

#### d) Existeix un regla per a la formació de lots (Rule) dins del mòdul BATCH.

Es poden formar per a qualsevol tipus d'entitat (any entity) o per a entitats amb un valor igual d'atribut (by attribute). En aquest últim cas s'ha d'especificar l'atribut en un camp especial que s'habilita quan s'escull aquesta opció.

#### e) Existeix un criteri d'assignació d'atributs per al lot creat dins del mòdul BATCH.

Permet especificar la forma en la qual el programa tractarà la transferència dels atributs que porten les entitats individualment, els quals poden ser diferents per a cada entitat que entra a formar part del lot.

El grup té els mateixos atributs que les entitats que el formen, però amb diferent valor segons a certs criteris o regles.

<b>Criteri</b>	<b>Descripció</b>
Primer (first)	Assigna als atributs de l'entitat representativa els valors dels atributs de la primera entitat que va arribar al mòdul BATCH.
Últim (last)	L'entitat representativa pren els valors dels atributs de la última entitat que va arribar al mòdul BATCH, és a dir la que completa el tamany del lot.
Suma (sum)	El valor dels atributs de l'entitat representativa es forma a partir de la suma dels valors dels atributs de les entitats que conformen el lot. Per exemple, si es té un atribut denominat <b>tipus</b> per a les entitats individuals, la representativa tindrà una altra atribut també denominat <b>tipus</b> però el seu valor serà l'equivalent a la suma dels valors dels atributs de mateix nom en cadascuna de les entitats individuals.
Producte (product)	Anàlogament al criteri de la suma, però en aquest cas l'entitat representativa tindrà una altra atribut també denominat <b>tipus</b> però el seu valor serà l'equivalent al producte dels valors dels atributs de mateix nom en cadascuna de les entitats individuals

**f) Atribut Entity.CreateTime en lots permanents.**

Sempre es transfereix amb el criteri first sense tenir en compte la regla que s'utilitzi al crear el lot.

**g) El mòdul Separate també fa un tractament posterior mitjançant unes regles als atributs de les entitats separades del lot.**

Aquestes regles afecten a cadascuna de les entitats regenerades i s'indiquen dins del camp "Member Atributes" del mòdul. Junt amb la seva funció i descripció s'exposen a la taula següent:

<b>Criteri</b>	<b>Funció</b>	<b>Descripció</b>
Retain Original Entity Values	Retindrà els valors originals	Quan el lot es desfà les seves entitats conserven els valors que originalment tenien els seus atributs corresponents.
Take All Representative Values	Assignar tots els valors representatius	Aquesta opció assigna els valors de l'entitat representativa a cada entitat regenerada unitàriament.
Take Específic Representative Values	Assignar solament valors representatius específics	Aquesta opció permet assignar els valors de l'entitat representativa a certs atributs de les entitats regenerades individualment. Aquests atributs s'especifiquen dins d'un quadre de llista que s'obre quan s'escull aquesta opció. Els atributs no seleccionats conservaran els seus valors originals després de desarmar el lot.

### 4.3.3. Sincronització del flux de les entitats a través del model.

En el medi real existeixen diferents operacions en les quals s'uneix un determinat nombre de components per a fabricar un producte final, això correspon a una operació d'assemblatge.

La funció del mòdul MATCH és la de sincronitzar el nombre d'entitats que se l'indiqui. Això vol dir que si el nombre es 3, per exemple, aquest mòdul generarà tres cues independents (una per a cada component), i només alliberarà la primera entitat de cada cua quan les tres estiguin disponibles. Cal dir que la condició d'assemblatge implica que de les tres s'ha de formar un component.

Si la regla de sincronització és per atribut, aleshores, les entitats s'acumulen en cada fila i només s'alliberaran quan a cada fila hi hagi una entitat amb el mateix valor d'atribut, sense importar la seva posició dins de la cua.

<b>Especificacions del mòdul avançat MATCH</b>		
<b>Camp</b>	<b>Descripció</b>	<b>Exemple</b>
<b>Name</b>	Nom del mòdul, ha de ser únic	VIDEO COMPONENTS
<b>Number to match</b>	Nombre d'entitats per a les quals tindrà lloc la sincronització, es genera una cua per a cada entitat. Ha de ser numèric	3
<b>Type</b>	Tipus de regla de sincronització. Pot ser per a qualsevol entitat (Any Entity) o per atributs (Based on Attribute).	Based on Attribute
<b>Attribute name</b>	Nom de l'atribut que s'utilitzarà com a referència per a sincronitzar les entitats que entren. Sols s'activa quan el tipus de regla es basa en atributs (Based on Attribute).	diàmetre

Poden haver entitats amb diferents valors d'atributs dins de la mateixa cua, però el mòdul MATCH farà coincidir sempre les que tinguin el mateix valor per a l'atribut especificat.

Posteriorment a la sincronització es sol utilitzar el mòdul BATCH per formar un lot o grup de tamany permanent, el qual estarà representat per una entitat (component ja assemblat) amb el nombre d'entitats que arribin a ell.

El nombre d'entitats que s'agruparan amb el mòdul BATCH ha de coincidir amb el nombre d'entitats que es vol sincronitzar mitjançant el mòdul MATCH.

Per a resumir aquest procediment es pot dir que el mòdul MATCH s'encarrega d'assemblar les entitats d'acord amb el nombre especificat i d'alliberar-les cap el mòdul BATCH, el qual les uneix en un grup permanent format per una sol'entitat.

#### 4.3.4. Distribució del flux de les entitats a través del model duplicant-les.

En molts casos, tant en el medi industrial com el de serveis, és possible trobar situacions en les quals és necessari duplicar una entitat, la qual pot simplement una peça o tota una ordre de comanda.

Per exemple:

- a) Dins d'un procés de trossejat entra una làmina de dimensions específiques a la màquina i surten sis rectangles com a producte acabat o subproducte.
- b) En un laboratori clínic a un pacient se li extrau una mostra de sang i se'l fa esperar dins la sala d'espera en espera del resultat. En aquest cas la mostra de sang és un duplicat del pacient, que haurà d'esperar el torn per a ser analitzada.
- c) En una companyia de distribució, una ordre de comanda pot ser processada per diferents departaments alhora, la mateixa haurà de ser duplicada, processada en paral·lel i consolidada al final.

Aquesta duplicitat d'entitats es realitza també mitjançant el bloc **SEPARATE**. El qual duplica entitats que hi entren de forma unitària si s'escull la opció "**Duplicate Original**" dins l'apartat Type.

Nota important:

- a) Quan es dupliquen entitats es tenen el número de duplicats + 1, que correspon a l'entitat original, dins del sistema.
- b) Duplicar té un cost associat.

Depenent del número d'entitats a duplicar es pot aplicar un número de 0 a 100 dins d'un camp del mòdul **Separate**. El mateix indica el percentatge dels costos i el temps de la unitat original que es distribuirà equitativament entre el número de duplicats i l'entitat original; mestres tant, l'entitat original conserva el percentatge restant.

Per exemple si col·loquem 90 i el nombre de duplicats és 3, la proporció per a cada rèplica és del 30% del cost i temps, mentre que l'original es queda amb el 10% restant.



c) **L'assignació de cost i temps només es realitza per aquests tipus d'atributs.**

La relació dels mateixos és la següent:

Atributs de Propòsit Especial		Atributs de Cost	Atributs de Temps
Entity.SerialNumber		Entity.VACost	Entity.VATime
Entity.Type		Entity.NVACost	Entity.NVATime
Entity.Jobstep		Entity.WaitCost	Entity.WaitTime
Entity.Picture		Entity.TransferCost	Entity.TransferTime
Entity.Station		Entity.OtherCost	Entity.OtherTime
Entity.Sequence			
Entity.HoldCostRate			

d) **L'atribut de propòsit especial Entity.CreateTime, el qual emmagatzema el temps en el qual una entitat es creada en el model es manté només en l'entitat original.**

Aquest atribut pren un nou valor per a cada entitat duplicada i l'original continua acumulant el temps d'aquesta entitat al sistema.

e) **Els atributs de propòsit especial i aquells definits per l'usuari no segueixen el criteri anterior.**

Aquests indiquen la seqüència, l'estació en la qual es troba l'entitat, etc., i estan formats per la relació de la taula mostrada.

Es dupliquen com a una còpia de l'atribut original, tant en identificació com en valor.

#### 4.3.5. Retenció d'entitats i posterior alliberament mitjançant events.

Existeixen situacions en les quals el flux normal d'un producte (entitats) és interromput fins que es produeixi un determinat event. Aquesta interrupció generalment exigeix que el producte s'aturi dins d'un espai físic (cua), la qual cosa provoca un retard que en alguns casos forma part del procés normal i no penalitza el cost de la simulació.

<b>Especificacions del mòdul avançat HOLD</b>		
<b>Camp</b>	<b>Descripció</b>	<b>Exemple</b>
<b>Name</b>	Nom del mòdul, ha de ser únic	HALL (Recepció hotel)
<b>Type</b>	Funció del mòdul; pot ser de tres tipus: esperar una senyal (Wait for Signal), avaluar una condició (Scan for Condition) o esperar de manera infinita (Infinite Hold).	Wait for Signal
<b>Wait for value</b>	Identificació de la senyal que s'espera. Aquest camp resulta útil per a diferenciar les senyals quan s'utilitza més d'una funció. Esperar una senyal, només s'activa quan el tipus es Wait for Signal. Ha de ser numèric.	1
<b>Límit</b>	Quantitat màxima d'entitats que després de trobar-se retingudes s'alliberaran un cop rebuda la senyal. Ha de ser numèric.	5
<b>Queue type</b>	Tipus de cua. Pot ser: Interna (amb la qual cosa no es mostren ni recullen estadístiques), normal (Queue), grup de cues (Set), indexada a un cua específica basada en un atribut o expressió (Attribute/Expression).	Queue (cua normal)
<b>Nom de la cua</b>	Nom de la cua (es actiu si el tipus de cua es Queue). Per lo general, el nom de la cua es genera automàticament amb el mateix nom del mòdul HOLD seguit d'un punt i la paraula reservada Queue, però l'usuari pot canviar la descripció abans del punt.	HALL.Queue

ARENA permet manegar aquest tipus de situacions mitjançant la implementació del mòdul lògic avançat "HOLD", les especificacions del mateix es donen a continuació.

El mòdul HOLD bàsicament compleix les tres funcions especificades al camp Type:

- 1. Wait for Signal.** Les entitats es retenen dins del mòdul HOLD en espera de ser alliberades per una senyal, la qual obligatòriament ha de ser enviada per un mòdul SIGNAL des d'una altra instància del model.

El mòdul "SIGNAL" envia doncs la senyal d'alliberació al mòdul o mòduls que contenen entitats que estiguin esperant. Per això es requereix el pas d'altres entitats no retingudes en lloc per aquest mòdul, les especificacions del mateix el mostren a continuació:

<b>Especificacions del mòdul avançat SIGNAL</b>		
<b>Camp</b>	<b>Descripció</b>	<b>Exemple</b>
<b>Name</b>	Nom del mòdul, ha de ser únic	TANC BUIT
<b>Signal value</b>	Valor de la senyal. El mateix serà enviat a tots els mòduls HOLD, inclús els que es trobin dins dels submodels (Configurats com a Wait for Signal) i que tinguin aquest mateix valor dins del seu camp Wait for Value. Ha de ser numèric.	1 (S'enviarà a tots els mòduls HOLD amb valor 1 dins del camp Wait for Value)
<b>Límit</b>	Valor màxim de les entitats que s'alliberaran. Numèric.	10

Amb la finalitat d'il·lustrar millor el funcionament dels mòduls Wait for Signal, a continuació es mostren dues possible situacions :

- **Situació 1.** Un mòdul SIGNAL té un valor de senyal 1 i la quantitat que es disposa alliberar és de 10. Es troben tres mòduls HOLD, cadascú amb 20 unitats retingudes, valor de senyal 1 i límit 5 unitats.

**Resultat 1.** El mòdul SIGNAL de 20 unitats, allibera 5 unitats del primer mòdul de les 20 que es troben retingudes, 5 del segon mòdul i cap del tercer.

- **Situació 2.** Un mòdul SIGNAL té un valor de senyal 1 i la quantitat que es disposa alliberar és de 12. Es troben tres mòduls HOLD, cadascú amb 20 unitats retingudes, valor de senyal 1 i límits de 5, 6 i 3 unitats respectivament.

**Resultat 2.** El mòdul SIGNAL allibera de 20 unitats, 5 unitats del primer mòdul de les 20 que es troben retingudes, 6 del segon mòdul totes les del límit (amb la qual cosa van 11 entitats alliberades de 12 possibles) i només 1 unitat del tercer (per arribar a les 12 que mana el mòdul SIGNAL alliberar com a màxim), encara que puguin alliberar-se més del tercer mòdul HOLD (fins a 3), no s'alliberaran.

**2. Scan for Condition.** Retindre entitats en un lloc determinat mitjançant un mòdul HOLD i alliberar-les només quan una condició específica es compleixi és una situació bastant comú dins d'un sistema.

A continuació es presenten alguns exemples.

- **Situació 1.** Un model de manufactura es deixen de passar peces mentre hi hagi capacitat d'atenció en una màquina.
- **Situació 2.** Es bloqueja una comanda en un sistema d'inventari, fins que s'arribi al punt mínim de comanda.
- **Situació 3.** En un sistema d'inspecció continu s'espera una seqüència de 10 unitats conformes per a canviar el tipus de mostreig.

Quan es presenta qualsevol d'aquests casos es necessari recórrer a la funció Scan for Condition del mòdul HOLD, el qual examina tot el sistema i retindrà l'entitat fins que es compleixi una condició per alliberar-la. En el cas que varies entitats es trobin esperant una senyal, la primera d'ella s'allibera i recorre els mòduls lògics posteriors fins que es pari o surti del sistema. Aquest procediment es pot donar amb un retard (recurs amb un temps de procés) o una altra mòdul HOLD (mòduls com DECIDE, ASSIGN, entre d'altres, no aturen l'entitat). Aleshores, el programa torna a avalua de nou la condició per a la propera entitat

que espera dins del mòdul HOLD. La mateixa serà alliberada o no d'acord amb el valor de la condició, a diferencia de l'acció WAIT, que allibera més d'una entitat al mateix temps quan ocorre un event de senyal.

3. **Infinite Hold.** Espera de forma infinita fins que posteriorment s'utilitzi un bloc "REMOVE" per tal que s'alliberi una entitat del bloc HOLD i segueixi el seu procés cap a d'altres mòduls. Aquest cas el desenvoluparem dins l'apartat següent, el qual implica l'estudi de la situació més complexa en el que respecta a la distribució del flux de les entitats a través del model.

#### 4.3.6. Recerca i extracció d' entitats d'una cua mitjançant entitats de control.

Tant en l'entorn industrial com en el de serveis es comú trobar-se amb situacions en les quals es necessari que, d'un determinat grup o cua, es realitza una selecció d'entitats amb característiques específiques. Per exemple:

- **Problema.** A la direcció d'una planta transformadora de cafè l'interessa separar el grans més de més tamany per a exportar-los, aquests es separen dels demés per a continuar un procés de refinament.

**Resolució.** Mitjançant un mòdul CREATE es generaran les arribades de cafè segons la distribució escollida.

A continuació mitjançant un mòdul ASSIGN s'assignarà un atribut anomenat diàmetre a cada entitat generada.

Segons les condicions que s'estipulen en l'exemple, les entitats ja creades es dirigiran cap a un emmagatzemament indefinit en espera de ser remogudes segons les condicions de diàmetre predefinides.

A continuació es mostra el diagrama del model per al procés principal en ARENA:



Tot i l'argument teòric, aquest model que s'està estudiant té una entitat en espera indefinida dins del mòdul HOLD Espera. Es fa necessari emprar una lògica independent per tal que prengui control del sistema i vagi avaluant cada entitat que entra al mòdul Hold Espera.

En aquest exemple l'avaluació s'ha de fer mitjançant la creació d'una entitat que inspeccioni cada posició de la cua HOLD Espera i revisi que la condició especificada es compleixi.

### Creació de l'entitat de control.

Aquest procediment es crea mitjançant un mòdul CREATE que genera una entitat en el temps 0 i amb un màxim d'arribades 1, tal i com es presenta a continuació.

L'entitat de control circula per un procés independent del principal i ha d'esperar a que hagi una o més entitats en cua del procés principal per avaluar la condició i retirar-ne alguna.

S'ha d'evitar remoure una entitat d'una cua buida, obtindrem un error en temps d'execució. Es recomanable doncs, que abans de realitzar aquesta operació, es verifiqui si a la cua hi ha elements.

Aquesta selecció es farà mitjançant la implementació de tres mòduls, a continuació de mòdul que ha creat l'entitat de control, a saber:

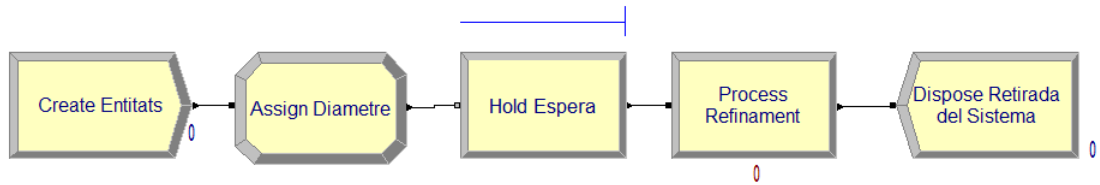
- a) **HOLD**. Reté l'entitat de control i l'allibera en cas que hi hagin elements en espera d'anàlisi a la cua del mòdul Hold Espera del procés principal.
- b) **SEARCH**. Busca el conjunt d'entitats que compleixen les condicions especificades al procés principal. Aquest mòdul treballa amb tres opcions:
  1. Search a queue.
  2. Search a batch.
  3. Search an expression.

Aquest mòdul **SEARCH** retorna, dins de la variable **Global J**, la posició en la cua per a la primera entitat que compleix amb les condicions especificades i en aquest apartat només tractem la primera opció, Search a queue.

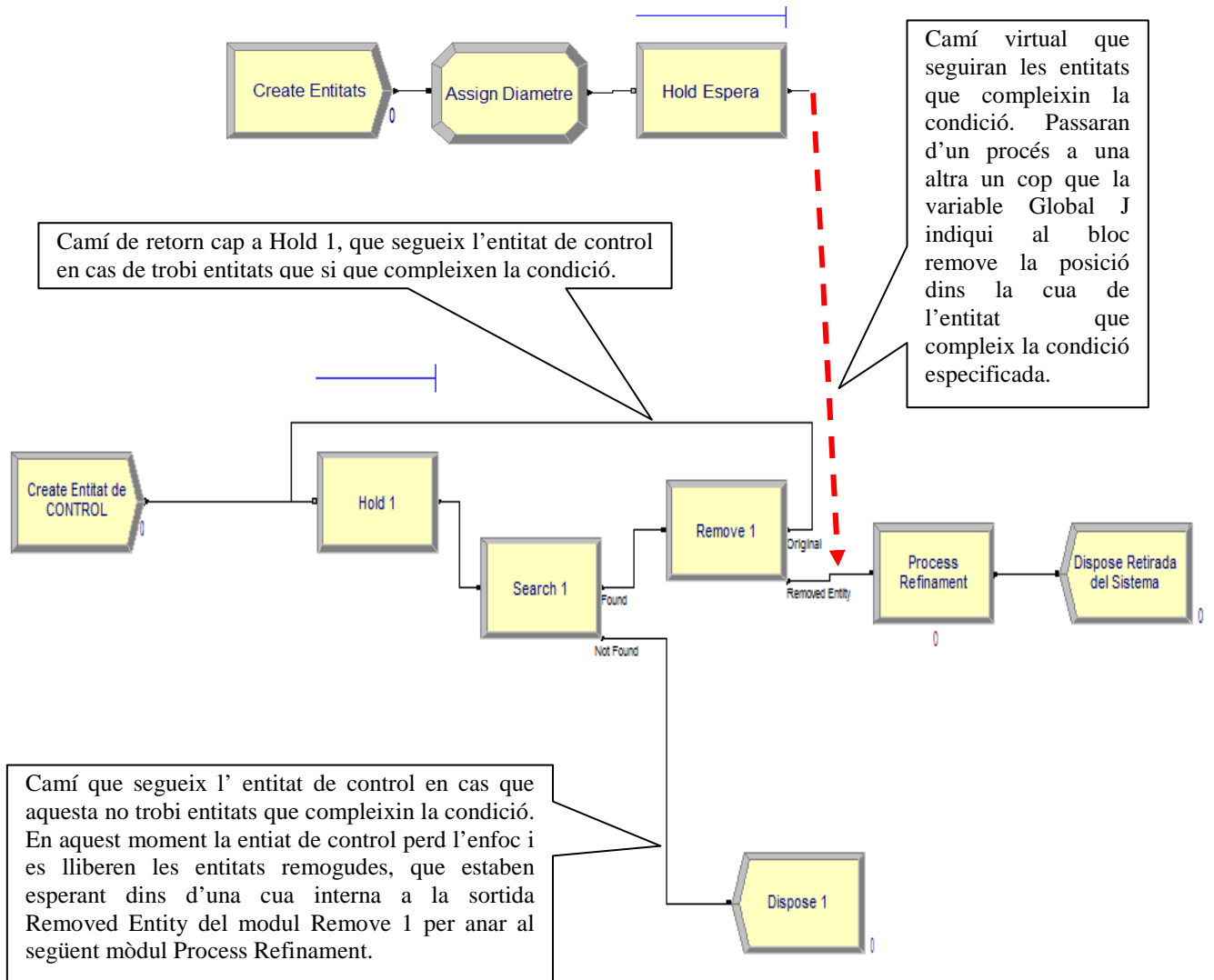
- c) **REMOVE**. Té dues sortides, per una surt l'entitat remoguda de la cua HOLD Espera del procés principal (Removed Entity) i per l'altre retornarà l'entitat de control al mòdul SEARCH, anterior per tal que verifiqui novament si hi ha elements a la cua HOLD Espera del procés principal.

L'assimilació d'aquest procés és difícil doncs les entitats remogudes de la cua HOLD Espera del procés principal salten, sense cap mena de connexió física, al procés lògic independent per a finalitzar el seu procés en cas que l'avaluació de la condició sigui certa. Removent l'entitat d'aquesta cua per realitzar les següents tasques de Process Refinement dins el procés lògic i finalment retirar-les amb el Dispose Retirada del Sistema dins d'aquest procés.

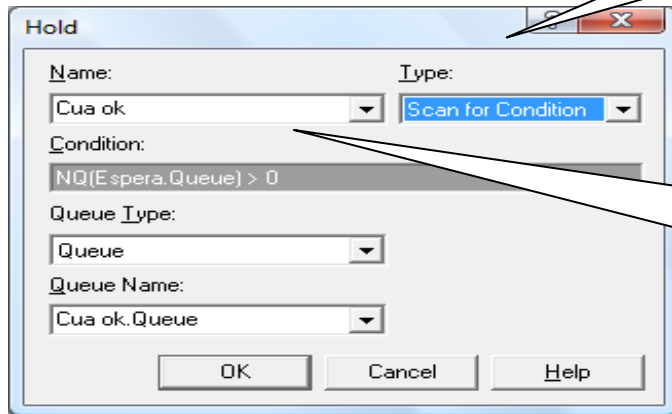
A efectes pràctica es com si executéssim el següent procés en continu:



Però el mateix es realitza en dos processos independents sense connexió física.



Per a verificar que la cua no estigui buida, utilitzem el mòdul HOLD 1, el qual està configurat com Scan for Condition, en el que només es permet el pas de l'entitat en cas que hi hagi més d'una esperant a la cua.

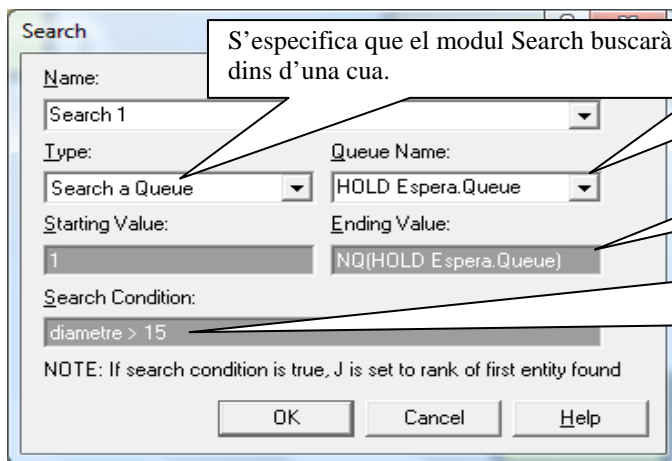


S'utilitza un Type Scan for Condition per a permetre el pas d'entitats nomésament si hi ha entitats a la cua.

S'especifica que abans d'alliberar una entitat del modul HOLD el nombre en la cua d'espera sigui més gran que 0. O sigui que es verifica que hi hagi algun element a la cua.

Una vegada que s'hagi verificat la condició de

la cua, es procedeix a realitzar la recerca d'entitats específiques mitjançant el mòdul SEARCH.



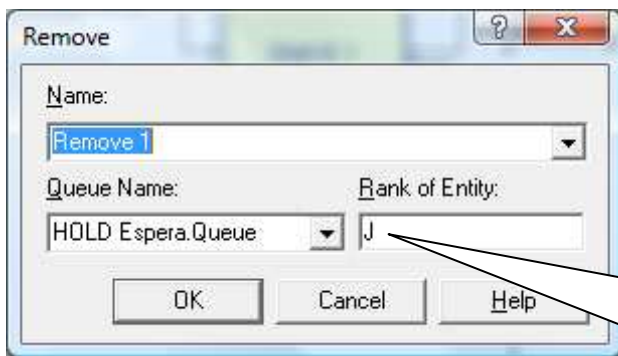
S'especifica el nom de la cua en la qual es farà la recerca.

S'especifica que el modul Search buscarà dins d'una cua.

S'indiquen les posicions d'inici i final de la recerca en la cua especificada mitjançant un valor enter inicial i una expressió final que donara el tamany en temps de real de la cua de recerca.

S'especifica la condició que s'avalua i en el moment que es compleixi que el valor de l'atribut diametre de l'entitat en espera a la cua especificada es major a 15, s'alliberarà aquesta entitat de la cua HOLD Espera que té una espera de temps infinita.

Finalment removem les entitats mitjançant el mòdul REMOVE anomenat Remove 1. Tenint en compte



que el camí que segueix l'entitat que remou o entitat de control sortirà per la sortida marcada com a Original i té sempre prioritat sobre l'entitat remoguda de la cua, la qual sortirà per la sortida marcada com a Removed Entity.

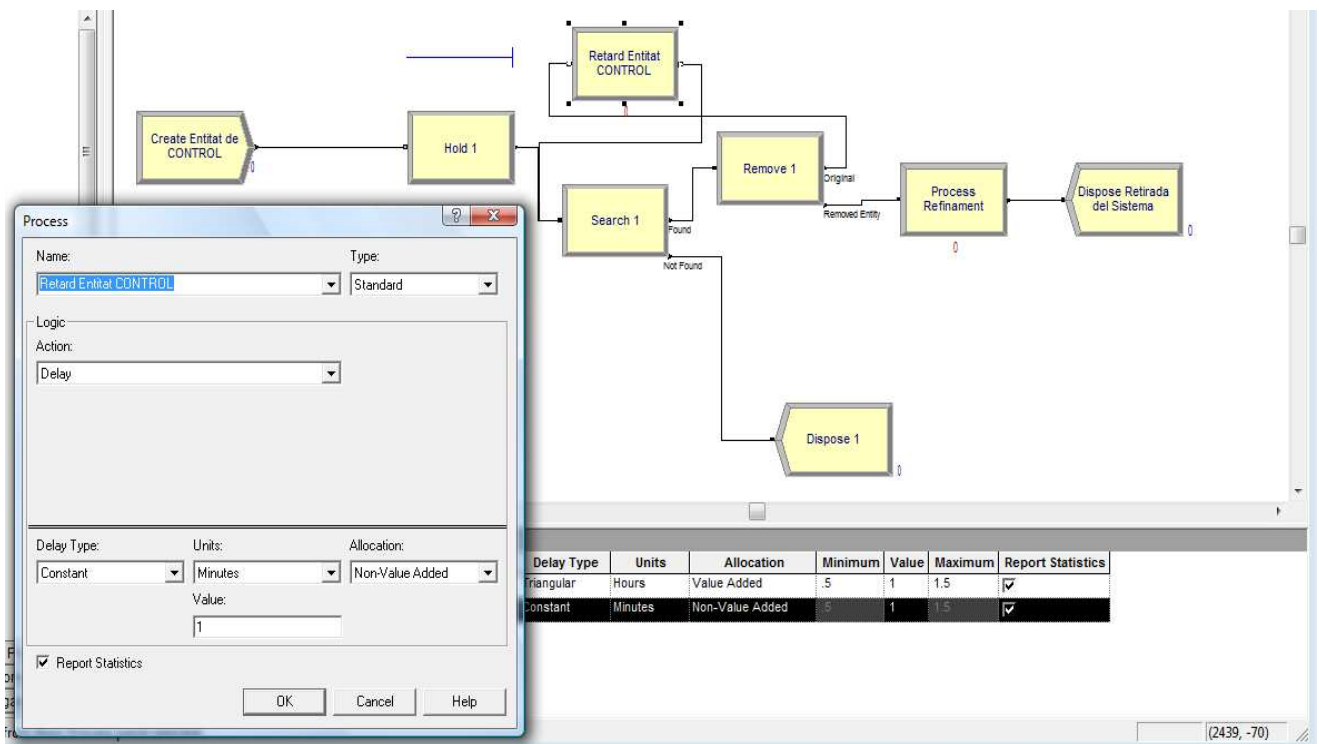
S'especifica la posició en la qual es troba l'entitat que es desitja remoure. Es coloca només la J, tal i com s'ha dit, en aquesta variable s'emmagatzema per part del modul anterior SEARCH, la posició de la primera entitat de la cua especificada que compleix amb la condició especificada.

L'entitat original es connecta de nou al mòdul SEARCH del principi per a tornar a verificar novament si hi ha cua. Això permet que mentre hi hagi cua, l'entitat de control continuï buscant fins que no trobi cap entitat amb la condició especificada.

Tal i com podem apreciar a l'esquema anterior, l'entitat de control sempre es troba recirculant sense esperes fins que acaba la recerca dins la cua, condició que passarà com em dit quan no quedi cap entitat que compleixi la condició enunciada.

Les entitats remogudes esperen en una cua interna fins que l'entitat de control tingui una espera i aleshores perdi l'enfoc de la simulació. Si col·loquem un retard en el bucle de retorn cap el mòdul SEARCH de l'entitat de control, d'una durada per exemple d'un minut, mitjançant un mòdul PROCESS, aconseguirem enviar cada entitat remoguda cap el mòdul PROCESS posterior immediatament. Això evita que totes les entitats esperin dins la cua interna de sortida del mòdul REMOVE fins la pèrdua de l'enfoc de l'entitat de control, al final de tota l'avaluació de la cua HOLD Espera.

Així el model complet del procés de la lògica independent amb el retard de l'entitat de control quedaria de la següent manera:





### 4.3.7. Recollida d'un grup d'entitats d'una cua i transport a una altra part del model.

En la construcció de models es poden presentar situacions en les quals es necessari seleccionar un grup d'entitats d'un conjunt de tamany més gran, per traslladar-lo posteriorment a unes altres instàncies del model ARENA. Aquesta aplicació és possible mitjançant la implementació ordenada del mòdul **HOLD, SEARCH, PICKUP I DROPOFF**.

A continuació es mostren exemples típics d'aquesta situació:

- a) Selecció d'una mostra de tamany 5 per trobar la mitja i la desviació estàndard.
- b) En un sistema de transport, quan en una ruta d'autobús s'especifica aquest arriba a una parada, només els viatgers que es trobin esperant el transport d'aquesta ruta pugen als diferents autocars.

Els mòduls lògics avançats **PICKUP** i **DROPOFF** es poden utilitzar per a remoure entitats d'una cua, amb certes diferències respecte al mòdul **REMOVE** vist en l'apartat anterior.

El mòdul **PICKUP** permet que una entitat de control reculli, d'una cua, una o més entitats dins d'un rang determinat.

Per exemple, de la cua recull entitats de la 1 fins a la 5. Aquest mòdul forma un grup amb les entitats recollides, el qual està representat per aquella que recull (entitat de control). La dimensió o tamany del grup s'emmagatzema dins d'una variable NG (Number in Group). Una vegada s'ha format el grup, les entitats recollides es remetran implícitament a on s'envii l'entitat de control.

Amb la finalitat de descarregar una o varies entitats del grup format, utilitzarem el mòdul lògic avançat **DROPOFF**. Dins d'aquest mòdul s'especifica la quantitat d'entitats que es descarregaran del grup i la posició que ocupen dins del mateix. El grup d'entitats, es pot considerar doncs com una successió d'entitats, es a dir, tal com una cua.

Mitjançant una combinació dels mòduls **SEARCH-PICKUP** també es pot inspeccionar una cua recollint entitats que compleixin una condició específica. Quan ja no es troben més entitats que compleixin la condició, el valor de la variable Global J serà 0 i la cua es deixarà d'inspeccionar.

Per exemple, un ascensor que es para en els pisos 4, 6 i 8 recollirà només passatgers el destí dels quals compleixi aquesta condició.

Mitjançant la combinació **SEARCH-DROPOFF** es pot inspeccionar el total del grup format (NG, Number in Group), amb la finalitat de buscar i dipositar entitats que compleixin una condició específica. Novament, quan ja no es trobin més entitats que compleixin la condició, el valor de la variable Global J serà 0 i el grup es deixarà d'inspeccionar.

Per exemple, quan un ascensor arriba a un pis determinat, es baixen d'ell només les persones per a les quals el destí sigui aquell pis.

## 4.4. ANÀLISI DELS COSTOS DEL SISTEMA.

Quan es desenvolupa una activitat productiva es generen costos, els quals varien segons la naturalesa de l'activitat. En general els costos es poden generar dins d'activitats que donen o no valor afegit al producte que es fabrica i la nostre missió és utilitzar la simulació per tal de minimitzar-los.

Podem tenir costos d'espera o transferència, un cost de processament o costos especials. L'entorn ARENA possibilita una classificació de costos segons les operacions que es realitzin, mitjançant una eina incorporada en el propi programa.

### Categories dels costos en ARENA

Costos de valor Agregat	Value added cost, VAcost
Costos que no agreguen valor	Non value added cost, NVAcost
Costos de transferència	Transfer cost
Costos d'espera	Wait cost
Altres costos	Other cost

Com a notes importants dins l'apartat de costos tenim:

- a) L'usuari ha d'establir prèviament la classificació de les activitats que generen costos.

Aquí s'ofereix el camp "**Allocation**", el qual apareix en tots els mòduls que poden afectar al temps d'estada d'una entitat dins del sistema i podem seleccionar qualsevol de les categories vistes abans.

- b) Cost inicial de l'entitat generada.

Dins la fulla de treball del mòdul bàsic de dades "**Entity**" podem inserir els costos inicials de cada entitat segons el seu tipus (Entity Type) abans de començar la simulació.

Entity - Basic Process									
	Entity Type	Initial Picture	Holding Cost / Hour	Initial VA Cost	Initial NVA Cost	Initial Waiting Cost	Initial Tran Cost	Initial Other Cost	Report Statistics
1	Truck	Picture.Report	50	0.0	0.0	12	0.0	0.0	<input checked="" type="checkbox"/>
2	Van	Picture.Report	30	0.0	0.0	0.0	0.0	0.0	<input checked="" type="checkbox"/>

Aquí s'ofereix la possibilitat d'introduir un cost per unitat de temps en el sistema (**Holding Cost/Hour**) o cost de permanència, el mateix va expressat en unitats monetàries per hora.

- c) El cost de permanència en temps d'execució es canvia mitjançant el mòdul Assing.

Donat que el cost de permanència ve assignat per l'usuari, el seu valor es pot modificar mitjançant un mòdul lògic com "**Assign**", doncs és un atribut de l'entitat anomenat "**Entity.HoldCostRate**".

Pel fet de ser un atribut també pot prendre valors diferents per a cada entitat, ser una distribució, formar grups (**Sets**) o d'altres aplicacions pròpies dels atributs.



- d) El cost de processament es defineix en la pròpia fulla de treball de recurs (Resource). D'aquesta categoria solament el cost per unitat processada al recurs cada cop que una unitat pren el control d'una unitat de recurs (**Per use**) i el cost per hora d'ocupació (**Busy/Hour**) d'un recurs es carreguen a l'entitat.

El cost per hora del recurs sense utilitzar (Idle/Hour) només es mostra en els informes de costos del recurs, però no forma part del cost de processament de l'entitat.

Adicionalment a l'entitat se li carrega el de permanència com hem vist abans però dins la categoria del cost al qual pertany l'activitat que es desenvolupa en el recurs.

- e) El cost generat per el temps que passa una entitat en una cua es assignat, per defecte, a la categoria **WAIT** i avaluat amb el valor de l'atribut **EntityHoldCostRate**, en cas que aquest sigui més gran que 0.

## 4.5. ANIMACIÓ DE LA SIMULACIÓ.

La practicitat del sistema ARENA es tal que realitzar l'animació de la simulació és molt simple, doncs s'aprofita el moviment de les entitats pel sistema per animar-les.

La presentació final de la simulació, sortint del model lògic tradicional on només tenim la imatge freda dels blocs a la pantalla, depèn de la imaginació de l'analista i del nivell de detall que es vulgui utilitzar.

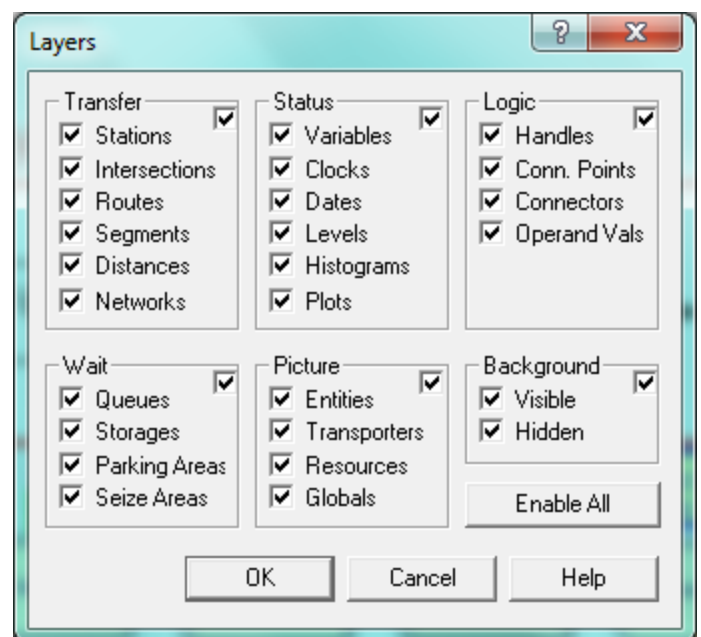
L'animació és sinònim de vida i si aconseguim crear un model gràfic i animat per a cadascun dels nostres projectes, amb tota seguretat la pràctica del modelatge amb ARENA s'extindrà a tot aquell que visualitzi els resultats de la nostre feina.

Ara bé l'animació del model pot arribar a tenir una complexitat molt elevada i aquest apartat no pretén donar un repàs a les formes de graficar bàsiques, donat que les mateixes queden molt ben resumides amb la part teòrica i les pràctiques que apareixen en el projecte anterior, *Estudi i guia docent per a modelatge i simulació de sistemes mitjançant l'entorn ARENA* <sup>[9]</sup>.

### 4.5.1. CAPES DE PRESENTACIÓ. Layers.

Selecció de la opció de menú principal "View -> Layers..." apareixerà el següent formulari, mitjançant el mateix podem fer aparèixer les capes que hi destina arena per a col·locar tots els elements gràfic.

Per exemple, un cop hem dibuixat les rutes que seguiran els elements de la simulació, com podem veure més endavant, podem fer que aquestes desapareixien visualment de la pantalla. Amb la qual cosa aconseguirem tenir una animació de més qualitat, evitant ensenyar les línies que seguiria el gràfic representatiu de l'entitat que es mou d'una a altre estació.



### 4.5.2. MODELS DE SIMULACIÓ GRÀFICA.

Pel que fa a les formes avançades de graficar, algunes ja apareixen també en aquest projecte precedent, però serà en aquestes en les quals centrarem el nostre repàs, dins d'aquest estudi avançat de simulació.

Com a guia per a l'animació i graficació del model, començarem resumint dins d'una taula els diferents models de simulació gràfica que ofereix l'entorn ARENA. Classificats segons es template on es guarden les icones dels blocs que haurem d'utilitzar, els tipus d'animacions que podem representar amb cada model, el nivell de complexitat alhora de graficar, el nombre de les pàgines que fan referència a l'apartat del projecte anterior (*Estudi i guia docent per a modelatge i simulació de sistemes mitjançant l'entorn ARENA* <sup>[9]</sup>, on ja s'implementa l'estudi de molts dels elements que es poden graficar i ha de ser d'obligada consulta previ a continuar amb els conceptes avançats d'aquest projecte) i l'apartat d'aquest projecte amb la referència dels elements nous que tractarem.

**Resum dels model de simulació gràfica de l'entorn ARENA**

<b>Template</b>	<b>Models</b>	<b>Nivell</b>	<b>Pàgines Referència Projectes</b>
<b>Basic Process</b>	Animació d'entitats, dels recursos i de les cues del model.  Graficació del model mitjançant la graficació de rellotges i dates, la representació d'histogrames i gràfics dinàmics mitjançant la opció Plot.	Bàsic	<b>PROJECTE ANTERIOR</b> Pàg. 31 -> Animació de recursos i cues. Pàg. 33 -> Gràfics dinàmics. Pàg. 45 -> Animació del recurs. Pàg. 46 -> Animació de les entitats. Pàg. 91 -> Modificació de l'animació de les cues. Pàg. 93 -> Modificació animació de les entitats. Pàg. 93 -> Addició d'animació als recursos. Pàg. 95 -> Addició de variables i gràfiques. Pàg. 96 -> Addició de complements i figures. <b>PROJECTE ACTUAL</b> Apartat 4.5.3.6 -> Graficació de rellotges. Apartat 4.5.3.7 -> Graficació de dates. Apartat 4.5.3.8 -> Representació d'histogrames.
<b>Advanced Process</b>	Animació de les cues mitjançant els emmagatzemaments.	Mitjà	<b>PROJECTE ACTUAL</b> Apartat 5.2.1 -> Emmagatzemaments.
<b>Advanced Transfer</b>	Animació del camí entre estacions, de les rutes lògiques, dels elements transportadors i de l'arribada d'entitats entre blocs.	Mitjà	<b>PROJECTE ANTERIOR</b> Pàg. 99 -> Addició de rutes lògiques. Pàg. 100 -> Animació d'estacions. Pàg. 101 -> Animació de rutes lògiques. Pàg. 102 -> Modificació de l'animació d'arribada d'entitats.

En els següents apartats veurem cadascun dels models d'animació i graficació d'ARENA, classificats segons el template on trobarem els mòduls necessaris per a modelitzar la presentació gràfica dels nostres models.

### 4.5.3. ANIMACIÓ I GRAFICACIÓ MITJANÇANT EL Template Basic Process.

Amb el mateix podem graficar i animar: Les entitats, els recursos, les cues, les variables, addicionar complements i figures externes al model, graficar rellotges, graficar dates, representar histogrames i realitzar gràfics dinàmics graficant variables o expressions en el temps mitjançant la opció PLOT.

#### 4.5.3.1. ANIMACIÓ D'ENTITATS.

Consisteix en assignar-li una figura o un dibuix a l'entitat que es mou dins del mòdul.

Per a qualsevol model en general, la figura inicial que adopten les entitats que ingressen al sistema es defineix dins de la fulla de treball de la pròpia entitat (ENTITY).

##### 4.5.3.1.1. Modificació en l'animació de les entitats.

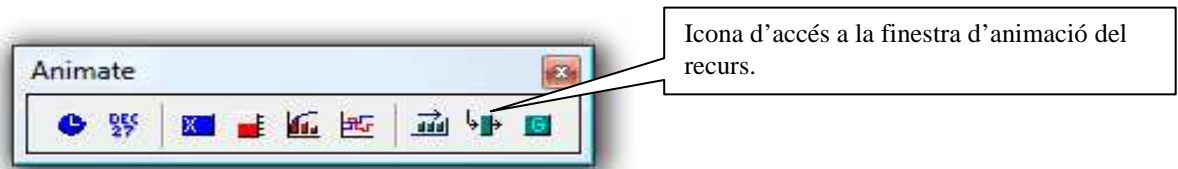
Les diferents figures o dibuixos que s'utilitzaran en el model es poden definir i editar en la finestra de canvi de figures (*Entity picture placement*) i les mateixes poden ser extretes de la llibreria gràfica interna que proporciona ARENA o bé els podem introduir externament.

Si es vol realitzar un canvi de la figura assignada inicialment a l'entitat dins del mòdul de dades ENTITY, posteriorment i en temps d'execució, aquest pot ser realitzat mitjançant el mòdul ASSIGN, un cop les entitat hagin entrat i sortit d'ell. Dins del quadre de diàleg d'aquest mòdul es presenta la opció de canviar l'atribut especial d'animació EntityPicture pel nou valor desitjat, amb la seva corresponent sintaxi i tenint en compte que la seva descripció i figura han d'haver estat definits prèviament, dins la finestra del canvi de figures.

##### 4.5.3.2. ANIMACIÓ D'UN RECURS.

Consisteix en assignar una figura específica a cada estat que el recurs pot adoptar, per exemple, una per quan es troba ocupat, una altra per quan es troba ocios i una altra per a quan es troba en reparació.

L'animació es realitzarà a través de la barra denominada Animate, la qual conté la icona que permet accedir a la finestra d'animació de recursos (Resource picture placement).

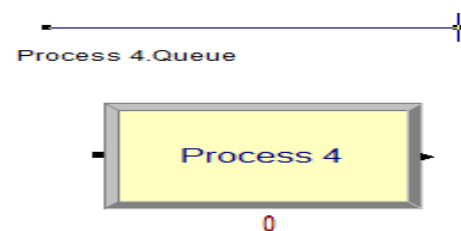


Per a veure exemples d'animació amb detall hem de repassar l'apartat 8.1.2 (pàgina 45) del projecte precedent.

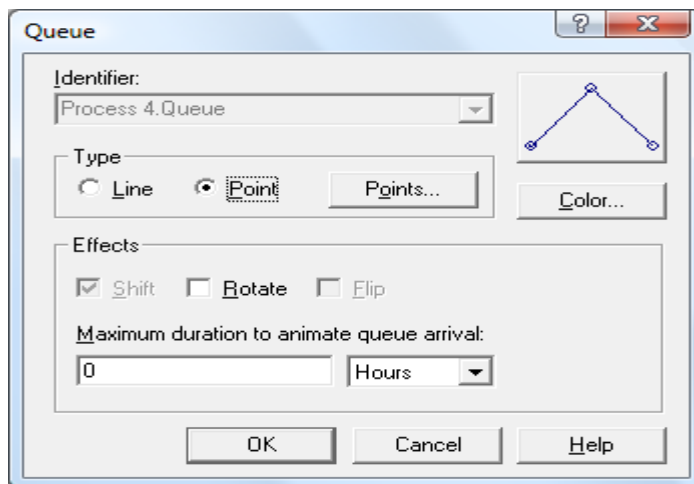
### 4.5.3.3. ANIMACIÓ DE LES CUES DEL MODEL.

Tot mòdul que provoqui esperes i generi cues dins del model mostrarà la cua corresponent mitjançant el següent gràfic.

El mateix presenta una representació molt simple i no té animació, unes poques figures representatives de l'entitat s'aturaran sobre la línia que marca la cua en temps d'execució.



Per tal d'animar les cues, el que es fa és canviar el seu tipus de representació per defecte, una línia horitzontal, per punts, fent doble clic al gràfic que representa la cua se'n obrirà aquesta nova finestra :



Dins la qual ja podem canviar el Type de la gràfica a punts.

El punts hauran de ser distribuïts pel model a modus de camí animat pel qual volem que passin les figures de les entitats. Les mateixes conforme vagin arribant a la cua, simularan un efecte d'animació gràfic mentre van passant per cadascun dels punts que formen la cua i ja es pot considerar que aquestes han passat a tenir vida

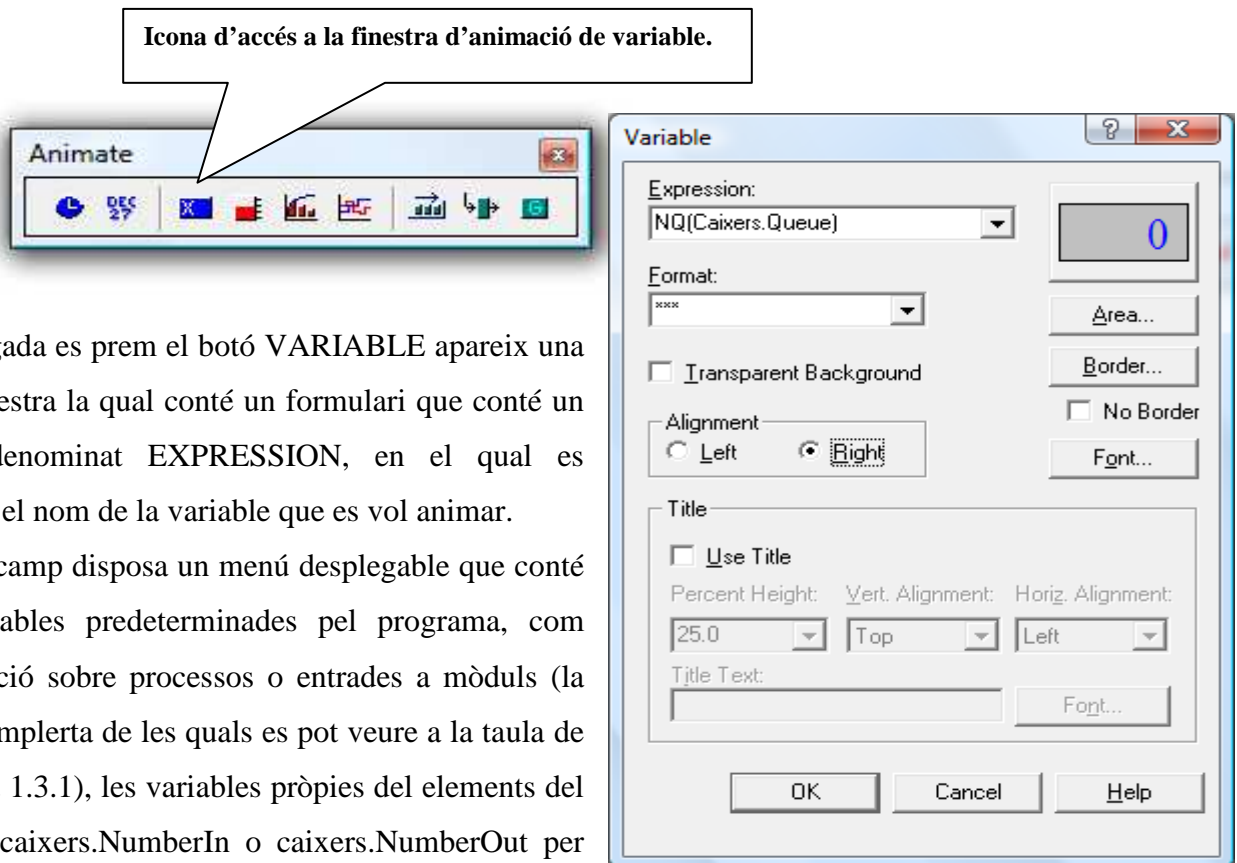
gràfica.

Per a veure exemples d'animació de cues amb detall hem de repassar l'apartat 9.3.1 (pàgina 91) del projecte precedent, Estudi i guia docent per a modelatge i simulació de sistemes mitjançant l'entorn ARENA.

#### 4.5.3.4. ANIMACIÓ DEL VALOR D'UNA VARIABLE EN EL MODEL.

Consisteix en mostrar de forma numèrica o gràfica el valor actual en temps de simulació d'una variable, amb la finalitat de verificar el funcionament de la lògica emprada, estudiar el seu comportament en el temps o senzillament, per raons estètiques.

Per mostrar el valor numèric d'una variable, ARENA compte amb una eina gràfica denominada VARIABLE, la qual es troba dins de la barra ANIMATE.



Una vegada es prem el botó VARIABLE apareix una altra finestra la qual conté un formulari que conté un camp denominat EXPRESSION, en el qual es col·loca el nom de la variable que es vol animar.

Aquest camp disposa un menú desplegable que conté les variables predeterminades pel programa, com informació sobre processos o entrades a mòduls (la llista completa de les quals es pot veure a la taula de l'apartat 1.3.1), les variables pròpies del elements del model (caixers.NumberIn o caixers.NumberOut per exemple).

Es possible que la variable que es desitja animar no es trobi a la llista, doncs es necessita que aquesta es construeixi mitjançant una expressió mitjançant l'assistent Expression Builder (comentat dins l'apartat 1.3.4).

Per exemple, suposem que volem mostrar en tot moment el nombre de clients que es troben dins la cua d'un caixer. Per graficar la variable col·locarem un quadre de variables, dins del qual, omplirem el valor del camp Expression amb la expressió formada mitjançant l'assistent Expression Builder següent :

NQ(caixers.Queue)

Dins del formulari VARIABLE també li donarem format al quadre gràfic variable, com nombre de dígit, color de l'àrea i del perfil i el tamany i l'estil de la font.



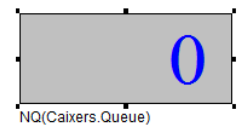
**El camp Format.**

Es refereix al format del número que es mostrarà el DISPLAY i ve donat per les posicions decimals i enteres que es requereixen més el signe, en el cas que es tractin valors negatius.

Aquestes posicions s'indiquen mitjançant asteriscs o signes de multiplicació. L'usuari pot introduir aquestes posicions o agafar qualsevol dels formats predeterminats que ofereix ARENA, entre els que es troben:

- \* -> Nombre d' un dígit enter positiu, doncs no hi ha espai per al signe.
- \*\* -> Nombre d' un dígit que accepta valors negatius.
- \*\*.\*\*\*\* -> Nombre de dos dígits enters i quatre decimals si és positiu i un enter i quatre decimals en el cas de sigui negatiu.
- \*.\* -> Nombre d'un dígit enter i un decimals si és positiu sense signe. Quan accepta valors negatius no accepta valors decimals.

Quan finalitza l' edició del formulari DISPLAY, el cursor es converteix en una creu, la qual cosa indica que s'està en modus dibuix i si es manté un clic sostingut, podem dibuixar un rectangle mostrat a la figura, per modificar el tamany del DISPLAY en pantalla.

**4.5.3.5. ADDICIÓ DE COMPLEMENTES I FIGURES AL MODEL.**

Podem addicionar nous complementes i figures al model mitjançant la finestra del canvi de figures. A la qual s'hi accedeix mitjançant la opció del menú principal “*Edit -> Entity Pictures*” i el funcionament d'aquesta finestra i el canvi de les figures que podem utilitzar en el nostre model el podem veure amb detall si repassem l'apartat 9.3.3 del projecte anterior <sup>[9]</sup> (pàgina 93).

**4.5.3.6. GRAFICACIÓ DE RELLOTGES.**

Es pot mostrar, per exemple, un comptador del temps de simulació transcorregut en forma de rellotge analògic (circular) o digital (rectangular). Aquesta funció és graficada mitjançant un rellotge, el qual es fa aparèixer al nostre projecte si fem clic sobre la icona en forma de rellotge incorporada dins de la barra ANIMATE.

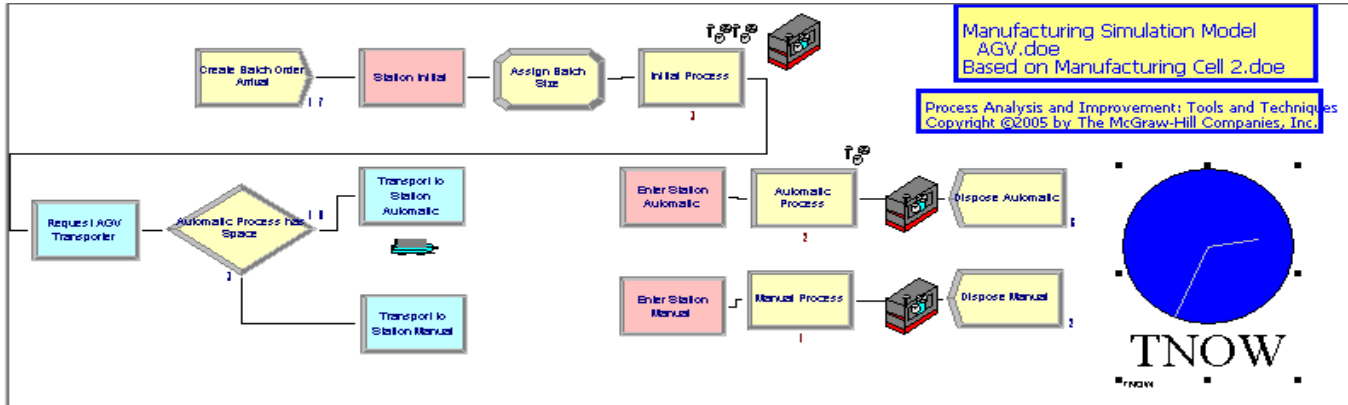
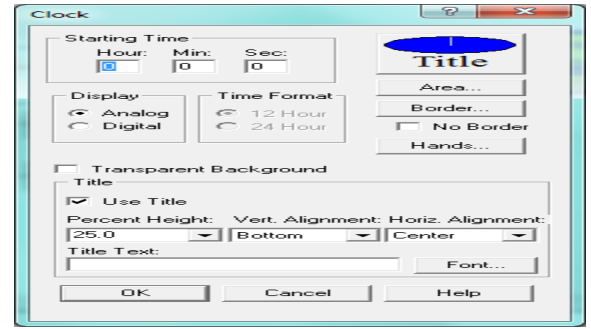
Té l'aspecte que es mostra en el model d'exemple que es mostra a la pàgina següent i simplement, per tal de graficar el temps real de

simulació, s'ha d'indicar com a *Starting Time* dins el formulari que apareix quan es fa clic a la icona del rellotge:



Hour = 0 ; Min = 0 ; Sec = 0

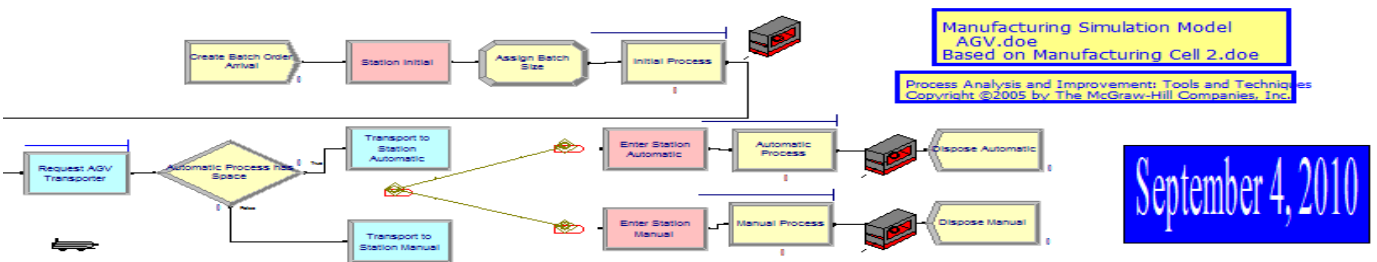
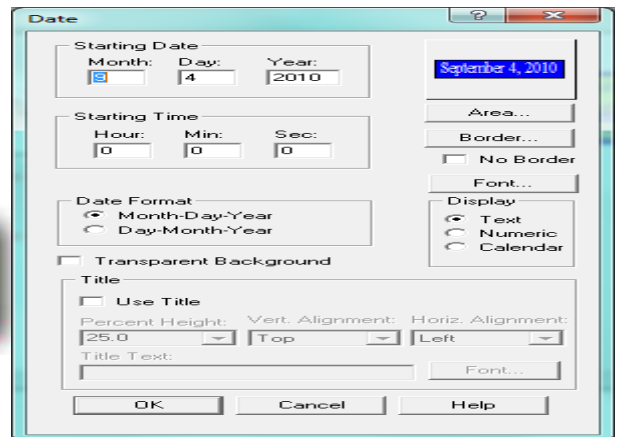
Aquest és el temps a partir del qual s'inicia el comptador i es pot col·locar com a títol TNOW, tenint en compte que aquest no te res a veure amb el valor que prem la variable TNOW, la qual conté el temps real de simulació.



### 4.5.3.7. GRAFICACIÓ DE DATES.

De forma anàloga a la graficació d'un rellotge, actuarem si volem introduir un calendari dins del nostre model.

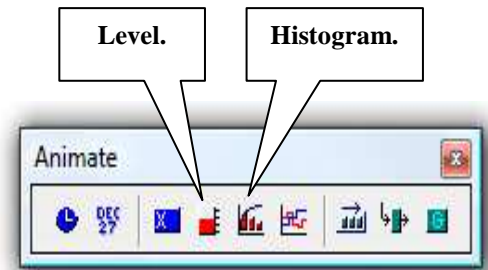
Fem click sobre la icona Date de la barra ANIMATE i ens apareix el formulari per indicar una data i hores d'inici, les quals s'aniran incrementant conforme passi el temps de simulació. Tal i com s'observa a la figura del model inferior.



### 4.5.3.8. REPRESENTACIÓ D'HISTOGRAMES.

ARENA pot anar acumulant valors d'una variable en sèries de quantitats predefinides. Una vegada tenim aquestes sèries, les mateixes seran representades mitjançant un histograma graficat en temps real de simulació.

Per tal d'incorporar un histograma al nostre model, farem servir el botó HISTOGRAM de la barra ANIMATE i al fer-hi clic amb el ratolí, apareixerà un formulari. Dins del qual i mitjançant l'editor ARENA Expression Builder, construirem l'expressió que graficarà les dades.



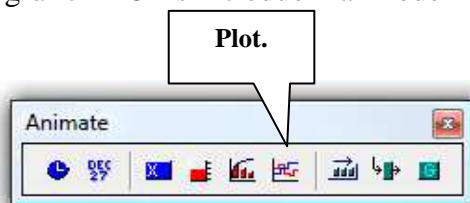
Donat que l'histograma es tracta d'una acumulació de dades d'una expressió, la mateixa també es pot fer servir en cas que es vulgui graficar una sola barra de l'histograma en forma de nivell. Això es realitzarà si premem el botó LEVEL de la barra ANIMATE i omplint la informació necessària per tal de fer aparèixer el gràfic de nivell per pantalla.

Amb la opció de gràfic LEVEL també pot ser utilitzada per representar nivells, com el d'un dipòsit, però aquesta opció és molt més completa si utilitzem els mòduls específics del *Template Flow Process*.

### 4.5.3.9. GRÀFICS DINÀMICS MITJANÇANT LA OPCIÓ PLOT.

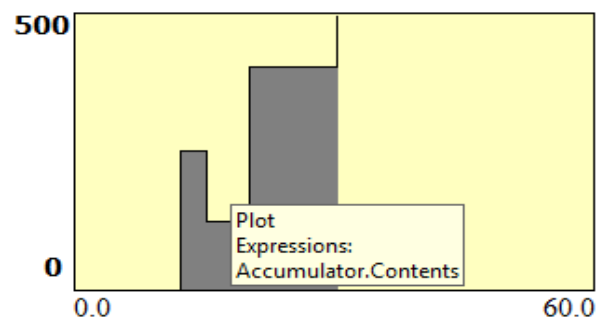
És un recurs d'animació de variables en forma de gràfic o l'eix X representa el temps i l'eix Y representa el valor de la variable d'interès en el temps.

El gràfic PLOT s'introdueix al model mitjançant la icona PLOT de la barra ANIMATE i si fem clic amb el ratolí apareixerà un formulari, amb el qual podrem graficar diverses expressions o variable al mateix temps.



Aquest menú i la graficació mitjançant aquest element és força complexa i la mateixa es troba molt ben desenvolupada dins l'estudi del projecte anterior <sup>[9]</sup>, concretament dins l'apartat 7.2.4 titulat Gràfics dinàmics (pàgina 33).

A la figura de la dreta es pot observar el resultat de graficar una variable anomenada *Accumulator.Contents*. La mateixa pot prendre valors entre 0 i 500 durant tot el temps de simulació i el gràfic ombrejat en color gris mostra aquest valor dins d'una escala de temps d'entre 0 i 60 minuts.



#### 4.5.4. ANIMACIÓ MITJANÇANT EL *Template Advanced Process*.

Per associar els blocs d'aquest template al nostre projecte, hem partit del menú principal, on escollirem : “ **File -> Template Panel -> Attach...** ” i seleccionarem el fitxer “**AdvancedProcess.tpo**”.

Ja vistos en l'apartat 3.8 d'aquest projecte, els mòduls que porta incorporat aquest template, porten funcions especialitzades que donen un major grau de detall i control al flux de les entitats.

El flux de les entitats pel model serà l'encarregat d'anar animant la simulació al seu pas pels elements que les retenguin, per això dins d'aquest *template* s'incorporen dos mòduls denominats **STORE** i **UNSTORE**. Els quals juguen un paper bàsic dins de l'animació avançada mitjançant els anomenats emmagatzemaments i seran tractats especialment dins de l'apartat següent.

##### 4.5.4.1. Retenció d' entitats i posterior alliberament. Emmagatzemaments.

En ARENA els emmagatzemaments tenen una aplicació diferent a la que usualment remet el significat d'aquest concepte. S'utilitzen quan una entitat entra a una sèrie de mòduls lògics que porten un temps associat, però si ho desitja, l'usuari pot veure l'animació de l'entitat fixada en un lloc específic mentre se li practiquen les diferents operacions. Això permet col·locar mecanismes de control de la lògica de la simulació per les diferents parts del model on ens calgui la monitorització del procés.

Una clara utilitzat dels emmagatzemaments s'evidencia quan s'utilitzen els mòduls **DELAY**, ja que en aquests l'entitat experimenta una demora, però la mateixa no pot se visualitzada mentre dura el procés.

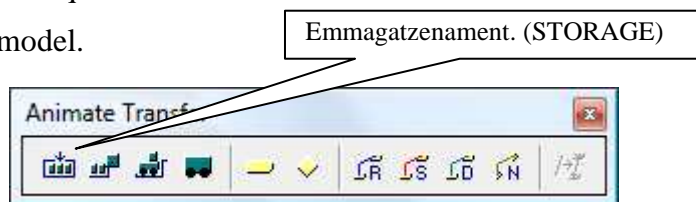
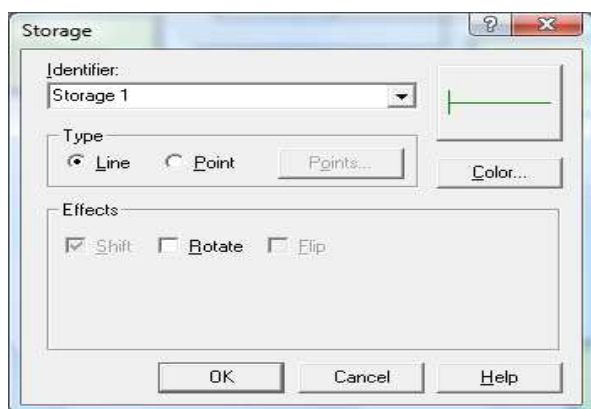
Per a visualitzar l'animació o figura de l'entitat que ingressa al mòdul lògic avançat **DELAY** s'utilitzen conjuntament els mòduls lògics avançats **STORE** i **UNSTORE** abans i després del mateix respectivament.

Especificacions del mòdul avançat <b>STORE</b>		
Camp	Descripció	Exemple
<b>Name</b>	Nom del mòdul, ha de ser únic	MAGATZEM ENTITAT
<b>Type</b>	Tipus d'emmagatzemament. Pot ser convencional (Storage), per mitjà de grups, a través de l'ús d'un atribut com a índex (Attribute) o d'una expressió també com a índex (Expression).	Storage

### Especificacions del mòdul avançat UNSTORE

Camp	Descripció	Exemple
<b>Name</b>	Nom del mòdul, ha de ser únic	LLIBERAR ENTITAT
<b>Type</b>	Tipus d'emmagatzemament del qual es retira l'entitat d'acord a la classe d'emmagatzemament que se li va donar (Storage, per mitjà de grups, Attribute o Expression).	Storage
<b>Storage name</b>	Nom de l'emmagatzemament del qual es retira l'entitat. Solsament s'activa quan és del tipus Storage.	Estorage 1

Amb STORAGE indicarem al sistema que aquesta entitat es visualitzarà en un lloc fixa (emmagatzemament) a partir d'aquest punt del model.



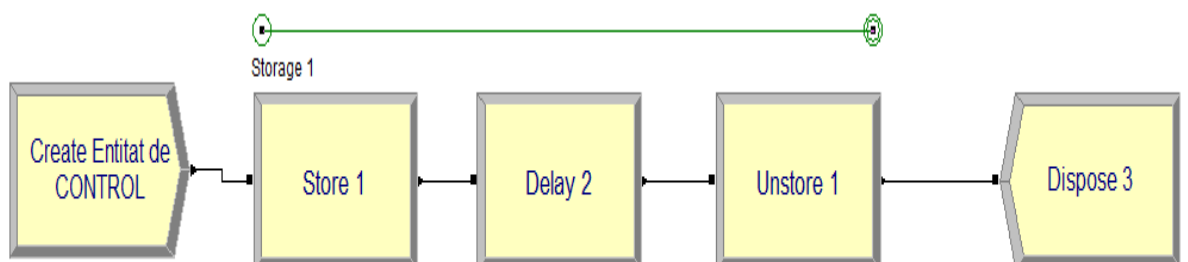
El lloc en el qual es desitja visualitzar l'animació de les entitats, el emmagatzemament, s'ha de crear mitjançant el botó Storage de la barra Animate Transfer.

Quan es fa clic sobre el botó Storage, el punter del mouse es transforma en una creu amb la finalitat

d'ubicar l'emmagatzemament en el lloc desitjat.

Aquest procediment es fa amb un clic sostingut si és del tipus Line i en cas de sigui del tipus Point, es fa clic per a cada posició del punt, finalitzant amb doble clic.

A la figura inferior es mostra el diagrama lògic de blocs del model en ARENA.



### 4.5.5. ANIMACIÓ MITJANÇANT EL *Template Advanced Transfer*.

En el modelatge de sistemes productius es freqüent trobar que les transferències de material es realitzen mitjançant un element transportador. A continuació es mostren alguns exemples d'aquesta situació:

- A través d'un dispositiu manual, un operari transfereix un lot de peces d'una estació a una altra.
- Els clients d'un complex turístic poden ser transportats des d'un punt d'entrada fins a diferents llocs i més tard ser recollits i portats de nou a la sortida.
- Els sistemes de distribució de productes en una zona geogràfica.

Els transports representen una limitació per a la transferència de les entitats, ja que tenen una capacitat finita i un cost associat. Aquest aspecte converteix al transport en un recurs més del procés productiu, per això ha de ser motiu acurat d'estudi.

Una entitat susceptible de ser transferida sol·licita un transport, si aquest es troba disponible i d'acord amb les seves característiques (velocitat), es dirigirà cap a l'estació on es troba l'entitat sol·licitant. Una vegada el transport es trobi en aquesta estació, l'entitat que serà transferida es carrega i es mou cap a l'estació de destí. Una vegada arribi, segons sigui el cas, l'entitat pot alliberar a l'element transportador per tal que aquest realitzi altres funcions o pot seguir mantenint el control d'aquest i reservar-lo en l'estació de destí, per tal que continuï sent transferida pel mateix element cap a altres destins.

És possible que es tingui més d'un element transportador (flota), si aquest és el cas, aleshores es pot sol·licitar una unitat de transport específica. L'assignació del transport es pot realitzar d'acord a certes regles, a saber:

- **CYCLICAL.** Assigna de manera cíclica els elements transportadors d'una flota a les entitats sol·licitants. Es a dir, existeix un ordre a la flota, l'últim element de transport utilitzat es col·loca en l'última posició per tornar a ser reassignat.
- **RANDOM.** L'assignació del transport es fa de forma aleatòria d'entre els elements disponibles.
- **PREFERRED ORDER.** Es seleccionarà primer la unitat de transport que tingui l'ordre més baix dins de la flota.
- **SPECIFIC MEMBERS.** Sol·licita un membre específic o grup d'elements de la flota.
- **LARGEST DISTANCE.** Sol·licita l'element de transport que es trobi més llunyà de l'estació des d'on l'entitat requereix el servei.

- **SMALLEST DISTANCE.** Sol·licita l'element de transport que es trobi més a prop de l'estació des d'on l'entitat requereix el servei.

Serà de gran utilitat per a l'estudi dels mòduls que incorpora aquest template, familiaritzar-nos amb el concepte d'estació i els transportadors fixes o conductors, anomenats conveyors dins del sistema ARENA.

Les estacions són utilitzades especialment per aïllar parts del model que funcionin independentment d'altres amb connexió lògica o física. Mentre que els conveyors són uns tipus de dispositius utilitzats per a moure entitats d'una estació a una altra.

S'entén millor, si diem que bàsicament les estacions s'utilitzen amb finalitats d'animació avançada del sistema.

La idea és que tots els blocs lògics que no comportin el moviment físic d'entitats (en relació al sistema real) siguin agrupats dins d'una estació i per tal que les entitats viatgin d'una estació a una altra és necessari que s'utilitzi qualsevol dels dos tipus de transport que disposa ARENA.

Per exemple, una peça que arriba a una màquina de control numèric (CNC) per tal de ser processada, patirà diverses accions passant per diferents mòduls del model però no es mourà i serà al final de tot el procés quan un tipus de conveyor la portarà cap a l'estoc. En aquest cas, tots els blocs utilitzats per representar les accions executades per la màquina poden ser agrupats en una sol'estació anomenada CNC i el transport entre aquesta i una altra estació denominada STOC serà fet per un element transportador que denominarem TRANSFER.

Per altra banda els conveyors poden venir representats, per exemple, per a una cinta transportadora. Aquests dispositius venen definits pel parell d'estacions que els connecten i per la distància entre elles, denominades segments. Les entitats han de ser carregades sobre o dins del conveyor, ocupant una o més cel·les d'aquest, dins de l'estació d'origen per poder ser transportades aleshores cap a l'estació de destí.

### 4.5.5.1. Mòduls de transferència avançada.

Per associar els blocs d'aquest template al nostre projecte, hem partit del menú principal, on escollirem :  
 “ **File -> Template Panel -> Attach...** “ i seleccionarem el fitxer “**AdvancedTransfer.tpo**”.

#### ▪ MÒDULS LÒGICS.

Formen part activa del model, aquest “template” ha de ser utilitzat principalment per a realitzar un modelatge més de les animacions per als nostres projectes de modelatge.

Formen part del “template Advanced Transfer” i estan formats pels següents mòduls:

1. **ENTER:** Es el primer mòdul d'un conjunt utilitzat per tal de definir una o més etapes de processament. Dins d'ell es defineix una estació o conjunt que correspon amb un espai físic o lògic on es durà a terme el procés.

Una entitat es pot moure cap aquest mòdul de dues formes: transferida per l' estació associada com un mòdul o mitjançant una connexió gràfica.

Quant una entitat arriba al mòdul ENTER, un retràs de descàrrega por ser seleccionat i algun equipament de transferència que s'hagi utilitzat per fer-la efectiva pot ser alliberat.

Si un recurs, transport o conveyor va ser utilitzat per transferir l'entitat cap aquest estació, aquesta pot ser utilitzada per alliberar-lo mitjançant el camp Transport In. Escollint “Release Resource” s'allibera un recurs, si “Free Transporter” es escollit el transport especificat dins del camp “Transporte Name” serà alliberat (FREE, FREED) i si s'escull “Exit Conveyor”, el mateix sortirà (IS EXITED)

2. **LEAVE:** Aquest mòdul s'utilitza per a transferir una entitat cap a una estació o mòdul. La mateixa pot ser transferida de dues maneres, gràficament o referenciant la estació, el conductor o el transportador.

Quant una entitat arriba a aquest mòdul, pot esperar per tal d'obtenir accés a l'equipament de transferència utilitzat (recurs, transportador o connector) i quan s'hagi obtingut, l'entitat pot patir encara un retràs de carregament, essent a continuació transferida d'aquest mòdul cap el seu destí posterior.

Quan l'entitat sol·liciti un transportador, aquest ha de ser prèviament definit dins del mòdul de dades TRANSPORTER i un mòdul DISTANCE ha de ser utilitzat per definir el camí que el transportador haurà de seguir entre estacions.



Quan s'accedeix a un conductor o Conveyor , aquest ha de ser prèviament definit dins d'un mòdul de dades CONVEYOR i una altra mòdul SEGMENT ha de ser utilitzat per tal de definir com ha de ser el flux de les entitats sobre ell.

El mòdul LEAVE és normalment el bloc de tancament d'una regió que defineix un àrea de treball. I aquest és l'últim d'una sèrie que s'inicia, per exemple, amb un mòdul ENTER, seguit per un SEIZE, un DELAY, un RELEASE o un PROCESS que unifica aquest tres i finalment s'acaba aquesta regió amb aquest mòdul LEAVE.

- 3. PICKSTATION:** Aquest mòdul possibilita que una entitat pugui seleccionar una estació d'entre múltiples estacions especificades, segons una lògica de decisió definida dins del mòdul. Aquest procés de selecció està basat en un valor mínim o màxim d'una variable o expressió.

Quan s'utilitza un transportador o conductors per a realitzar la transferència d'una estació a una altra, és necessari que l'entitat tingui anteriorment reservat per a ella un d'aquest dos tipus de movimentadors. Blocs com REQUEST per els transportadors o ACCESS per als conductors, han d'haver estat accionats en posicions anterior al flux d'entitats que entri dins del mòdul PICKSTATION.

- 4. ROUTE:** Aquest mòdul transfereix una entitat cap a una estació específica o cap la propera estació dins d'un circuit seqüencial d'estacions definida per a una entitat. Es pot definir un retràs aplicable per aquesta transferència.

Quan una entitat entra en aquest mòdul, el seu atribut Entity.Station és alterat incorporant el nom de l'estació de destí on s'enviarà aquell'entitat en un temps predefinit dins del camp Route Time.

La variable NE pot ser utilitzada per

descobrir el nombre d'entitats que s'estan dirigint cap a una estació determinada, enviades a través d'aquest bloc ROUTE, el TRANSPORT o el CONVEY.

- 5. STATION:** Aquest mòdul defineix una estació o un conjunt d'estacions, les quals corresponen a un lloc físic o lògic on tindrà lloc el seu processament.

Si el mòdul defineix un conjunt d'estacions (Station Set), aquest estarà definint realment múltiples llocs de processament.

Una entitat es mourà directament d'un dels mòduls on s'inicia la transferència (ROUTE, TRANSPORTER o CONVEY) cap al pròxim bloc STATION, sense importar el lloc de la pantalla on aquest estigui.

No hi ha connexió física entre les illes formades per les estacions de treball. Aquestes es connecten a través de la transferència de les figures en el model gràfic, les quals seguiran un camí o flux lògic predefinit per un mòdul com ROUTE.

- 6. ACCESS:** Aquest mòdul assigna una o més cel·les d'un conveyor a una entitat per tal de moure-la d'una estació a una altra.

Quan una entitat arriba a aquest mòdul, espera fins que el nombre apropiat de cel·les contigües estiguin buides i alineades com a una estació.

L'atribut Entity.Station de l'entitat ha de tenir un valor vàlid quan aquesta intenta accedir a un conveyor.

Aquest és un error molt comú que es pot donar normalment i és causat per un mòdul SEPARATE. Millor dit, quan un grup és format dins d'un bloc BATCH, les entitats perden els seus atributs individuals i tots s'agrupen dins de l'entitat representativa del grup. Si aquesta abandona l'estació com a membre del grup, la seva Entity.Station no es canviarà. Quan el grup passi pel mòdul SEPARATE i si la opció "Retain all original values" es troba activada (és la opció per defecte), l'entitat tornarà a tenir com a valor del seu atribut Entity.Station el que tenia abans de convertir-se en membre del grup.

D'aquesta forma, quan ella intenta accedir al conveyor, la seva entitat d'origen no correspondrà amb la que el bloc ACCESS li demana per assignar-li una cel·la del conveyor.

- 7. CONVEY:** Aquest mòdul és el responsable del transport d'una entitat des d'una estació fins a una altra mitjançant un conveyor.

El temps que trigarà el transport d'una estació a una altra està basat en la velocitat del conveyor, especificada dins del mòdul de dades CONVEYOR, i la distància entre estacions s'especificarà dins del mòdul de dades SEGMENT.

Quan una entitat entra en aquest mòdul, el seu atribut Entity.Station és definit amb l'estació de destí. Si el tipus d'estació de destí és definit com a "Sequential", la propera estació vindrà predefinida pels atributs Entity.Sequence i Entity.Jobstep de l'entitat que acaba d'entrar al mòdul.

És necessari que l'entitat tingui accés a les cel·les del conveyor, mitjançant el bloc ACCESS abans de ser conduïdes cap a la seva estació destí o abans d'entrar al mòdul CONVEY.

Si no s'especifica el nom del conveyor que s'utilitzarà, l'entitat es transportarà mitjançant l'últim conveyor al qual aquesta havia tingut accés anteriorment, mitjançant un mòdul ACCESS.

- 8. EXIT:** En aquest mòdul s'alliberen les cel·les ocupades per l'entitat dins d'un conveyor. Si una altra entitat està esperant aquestes cel·les disponibles en la mateixa estació on es produeix l'alliberament, aquesta accedirà al conveyor.

Sembla obvi, però no improbable que sigui una font d'errors, que l'entitat hagi de tenir prèviament accés a un conveyor mitjançant un bloc ACCESS, per tal que després aquesta pugui sortir a través d'aquest mòdul.

En cas que no s'especifiqui el nom del conveyor, l'entitat referenciada alliberarà les cel·les de l'últim que l'ha transportat.

Una entitat que tingui ocupades cel·les d'un conveyor no ha de ser precisament conduïda per aquest, aquesta pot simplement accedir a les cel·les, provocar un retràs i a continuació alliberar-les. Això pot servir per tal de simular, per exemple, un bloqueig o qualsevol element que provoqui l'aturada del conveyor, però en aquests casos és millor utilitzar els mòduls específics per a simular fallides.

- 9. START:** Aquest mòdul recupera un conveyor del seu estat inactiu. En el moment d'activació de nou del conveyor, l'estat pot venir d'inactiu (3), d'espera o Idle (0), d'actiu o Moving (1) o que el mateix es trobi bloquejat o Blocked (2). En aquest moment existeix la possibilitat d'alterar la velocitat del conveyor.

Les entitats poden continuar dins del conveyor quan aquest es troba inactiu i romandran en les mateixes posicions que abans de tornar a activar-se.

Si apliquem START sobre un conveyor que ja es troba actiu, aquesta ordre no sorgirà cap efecte sobre el mateix.

- 10. STOP:** Aquest mòdul passa a inactiu el conveyor al que fa referència. Quan una entitat entra en aquest mòdul, el conveyor indicat s'atura immediatament i la seva variable interna d'estat ICS passa a valer 3 (inactive), independentment del tipus de conveyor o el nombre d'entitats que estiguin sent conduïdes.

Les entitats conduïdes continuaran dins del conveyor quan aquest es troba inactiu i romandran en les mateixes posicions, fins que alguna entitat travessi un mòdul START que torni a activar el conveyor.

Si apliquem STOP sobre un conveyor que ja es troba aturat, aquesta ordre no sorgirà cap efecte sobre el mateix que abans d'aturar-se.

Alterar la velocitat no alterarà l'estat d'un conveyor representada per la variable VC i carreguem el valor de velocitat zero no significa per l'estat el mateix que aturar-la amb STOP.

**11. ACTIVATE:** Aquells transportadors desactivat prèviament mitjançant el mòdul HALT, seran activats de nou quan una entitat de control entri a aquest mòdul.

Una unitat de transportador romandrà inactiva en el lloc on va ser aturada fins que una altra sol·liciti la seva activació.

En el moment d'activament de nou del transportador, la seva variable IT passarà de valdre 2 (inactive) a 0 indicant que passa a ociós (Idle) o 1 si es troba ocupat (busy) en el cas que existeixin entitats per a ser transportades per aquest transportador.

El bloc HALT és equivalent al bloc START utilitzat per activar conductors.

**12. ALLOCATE:** Aquest assigna un transportador a una entitat que realitza una petició de transport, de manera similar a com opera el mòdul SEIZE quan aquest utilitza recursos. Si el transport no es troba immediatament disponible, l'entitat espera dins d'una cua fins que aquest arribi.

ALLOCATE només assigna una unitat de transport a l'entitat i NO mou l'entitat que realitza l'envio del transportador amb aquest i tampoc canvia l'estat del transport de lliure a ocupat. El que és útil, per exemple, quan es desitja realitzar altres activitats abans o mentre que el transport s'està movent des de la seva ubicació anterior, fins on es troba l'entitat sol·licitant.

Una entitat que entri a aquest mòdul ha de tenir de totes maneres un valor vàlid dins del seu atribut Entity.Station, doncs és el mètode d'assignació d'unitats de transport per a ser destinades a transport de llarga distància (Largest Distance) o curta distància (Smallest Distance) i aquesta informació podrà ser utilitzada per a realitzar els càlculs de distàncies.

Quan hi ha més d'una unitat de transport per a ser assignada, utilitzarem les regles de selecció i si dues entitats amb la mateixa prioritat intenten obtenir el mateix transportador en mòduls diferents, agafarà el control la que es trobi més propera en distància al lloc on es trobi el transportador.

Els estats de les variables del transportador, NT e IT, s'alteren quan aquest és assignat a qualsevol entitat.

La variable NT representa el nombre d'entitats en estat ocupat (busy) dins del conjunt de transportadors e incrementa el seu valor en 1 quan una nova és assignada.

La variable IT indica l'estat d'una unitat en particular de transport i rep el valor 1 quan aquest passa a l'estat ocupat (busy).

**13. FREE:** Aquest mòdul allibera el transportador més recentment utilitzat.

Si no existeixen entitats que esperen l'alliberació d'un transport, aquest passarà a l'estat ociós dins de l'estació on va tenir lloc la seva alliberació, a no ser que estigui especificat diferent dins del mòdul Advanced Transfer lògic TRANSPORTER.

<b>Especificacions del mòdul Advanced Transfer FREE</b>	
<b>Camp</b>	<b>Descripció</b>
<b>Name</b>	Nom del mòdul, ha de ser únic
<b>Transport name</b>	Nom o identificador del transport o flota a la qual pertany el transport alliberat.

**14. HALT:** En aquest mòdul canvia l'estat d'una entitat transportadora i passa aquest a inactiu (Inactive).

Si en el moment d'entrar una entitat a aquest mòdul, l'element de transport que ha de canviar l'estat es troba ocupat, el seu estat serà ocupat e inactiu fins que l'entitat controladora sigui alliberada. En canvi, si l'estat del transportador és ociós (Idle) quan una entitat el defineix com a inactiu mitjançant aquest mòdul, el mateix passarà a estat inactiu immediatament.

El transportador en aquest estat no podrà donar servei a cap entitat sol·licitant de transport, fins que sigui activat de nou.

El bloc HALT és equivalent al bloc STOP utilitzat per a desactivar conductors.

**15. MOVE:** Provoca el moviment d'un transportador, el qual ja ha estat assignat prèviament, cap a la seva ubicació destí però sense transportar amb ell l'entitat de control que ha entrat en aquest mòdul i ha activat aquest procés.

L'entitat de control romandrà dins d'aquest mòdul fins que el transportador hagi arribat al seu destí i aleshores passarà de nou a l'estat disponible, per a ser moguda cap a una altra mòdul o per a realitzar altres tasques dins del sistema.

El temps que triga el transportador en arribar al destí depèn de la seva velocitat, especificada en el mòdul de dades TRANSPORTER i la distància entre estacions la indicarem dins del mòdul de dades DISTANCE.

L'entitat ha d'haver agafat control prèviament sobre el transportador mitjançant un mòdul del tipus REQUEST o ALLOCATE.

Si el nombre d'unitats de transport a ser mogudes no s'ha especificat, serà utilitzat el mateix nombre que recentment s'hagi assignat o demanat (Allocated or Requested).

**16. REQUEST:** Aquest mòdul assigna a una entitat que es vol moure, una unitat d'un transportador disponible determinat. Mitjançant una entitat específica del transportador o una regla d'assignació predeterminada.

L'entitat romandrà a aquest mòdul fins que la seva petició de transport sigui atesa, a més de tenir un valor vàlid en el seu atribut Entity.Station. Lo qual evita el mateix problema explicat dins del mòdul ACCESS, causar per les especificacions fetes pels mòduls BATCH i SEPARATE principalment.

<b>Especificacions del mòdul Advanced Transfer REQUEST</b>	
<b>Camp</b>	<b>Descripció</b>
<b>Name</b>	Nom del mòdul, ha de ser únic
<b>Transport name</b>	Nom o identificador del transport o flota.
<b>Selection Rule</b>	Regla de selecció de transport.
<b>Save Attribute</b>	Nom de l'atribut en el qual s'emmagatzemarà, si es desitja, el nombre de l'element transportador assignat, es a dir, el seu índex dins de la flota o grup.
<b>Unit Number</b>	Sols s'activa quan la regla de selecció és Específic member. El nombre de la unitat o element de transport que es sol·licita s'ha d'introduir dins de la flota.
<b>Priority</b>	Li assigna prioritat a la sol·licitud de transport per al mòdul especificat. Es d'utilitat quan es tenen varies sol·licituds per a un mateix transport i una o més son de més importància.
<b>Velocity</b>	Velocitat amb la qual es desplaçarà el vehicle d'allà on es troba fins l'estació que sol·licita el servei (velocitat sense càrrega).
<b>Time Units</b>	Unitats mitjançant les quals s'expressa la velocitat. S'ha de tenir cura en mantindre la concordança entre les unitats de distància i velocitat.
<b>Queue Type</b>	Tipus de cua en la qual s'emmagatzema l'entitat que espera transport dins d'aquest mòdul.
<b>Queue Name</b>	Nom de la cua d'emmagatzement de l'entitat d'espera que adopta el nom del mòdul per defecte.

El mòdul REQUEST és una combinació dels mòduls ALLOCATE seguit de MOVE. En el cas de reservar el transportador per a una entitat, al mateix temps ja s'està movent el transportador cap el lloc on se'l demana, contràriament al que realitzen aquest dos mòduls on no es mou el transportador.

**17. TRANSPORT:** Aquest mòdul transporta l'entitat controladora del transport junt al propi transport en qüestió, entre dues estacions.

El temps necessari per a moure els dos elements està basat en la velocitat del transportador i la distància entre estacions.

Si comparem TRANSPORT amb MOVE, el segon permet moviments sense càrrega i el primer sempre es mou carregat amb l'entitat que va activar-lo.

Quan una entitat entra a aquest mòdul, el seu atribut Entity.Station rep el codi o valor representatiu de l'estació de destí. L'entitat és immediatament transportada cap aquesta propera estació.

Abans que el transportador pugui ser reassignat cap a una altra entitat que l'ha sol·licitat, l'hem d'alliberar de l'entitat de control mitjançant el mòdul FREE.

<b>Especificacions del mòdul Advanced Transfer TRANSPORT</b>	
<b>Camp</b>	<b>Descripció</b>
<b>Name</b>	Nom del mòdul, ha de ser únic
<b>Transport name</b>	Nom o identificador del transport o flota.
<b>Unit Number</b>	Sols s'activa quan la regla de selecció és Específic member. El nombre de la unitat o element de transport que es sol·licita s'ha d'introduir dins de la flota.
<b>Destination Type</b>	S'especifica el lloc fins el qual es mourà l'entitat, el qual pot ser cap a una estació (Station). Es pot utilitzar una seqüència (Sequential), tanmateix la informació del seu destí pot estar emmagatzemada dins d'un atribut (Attribute) o mitjançant l'avaluació d'una expressió (Expression).
<b>Velocity</b>	Velocitat de l'element expressada en unitats de longitud, la qual serà equivalent a la distància que recorre el transport (velocitat amb càrrega).
<b>Time Units</b>	Unitats mitjançant les quals s'expressa la velocitat. S'ha de tenir cura en mantindre la concordança entre les unitats de distància i velocitat.

#### ▪ MÒDULS DE DADES.

Venen en forma de fulles de treball, no apareixen directament en el model gràfic doncs no tenen caràcter actiu dins la simulació. Son editats a través de formularis i la seva funció és la declaració de les especificacions per a cada element dels mòduls de flux, fixant els seus valors inicials i les seves propietats. Formen part del "template Advanced Transfer" i estan formats pels següents mòduls:

**18. SEQUENCE:** Dins d'aquest mòdul de dades es defineix una seqüència lògica d'estacions per al flux de les entitats a través del model. Per a cada estació de la seqüència visitada, els atributs i a les variables que prendran les entitats que hi passin es poden assignar dins d'aquest mòdul.

**19. CONVEYOR:** Aquest mòdul de dades defineix els paràmetres per a cada conveyor que es vulgui utilitzar en el sistema. Entre d'altres, es pot definir si en conveyor serà o no acumulatiu, el tamany de cel·la en unitats físiques com per exemple metres o la seva velocitat.

Un conveyor consisteix en una seqüència de segments entre estacions definits dins del mòdul de dades SEGMENT i la suma de distàncies entre elles especificades dins d'aquest mòdul poden ser dividides pel tamany de cel·la. En cas contrari tindrà lloc un error quan el model sigui verificat.

**20. SEGMENT:** En aquest mòdul de dades es definiran la seqüència de segments que formen un conveyor. Cada segment és un connexió directe entre dues estacions. L'adreça del conveyor és definida per una estació d'origen i una sèrie d'estacions següents, a cada parell de les quals hi associarem una distància.

No es poden repetir noms d'estacions dins del circuit d'estacions pels quals passarà el conveyor, a no ser que aquest sigui circular. En aquest últim cas la primera i última estació del circuit tindran el mateix nom.

**21. TRANSPORTER:** Aquest mòdul de dades permet a l'usuari definir les característiques d'un grup d'unitats transportadores. Per exemple, podem canviar el nombre d'unitats del grup, la velocitat per defecte i la seva localització inicial. Addicionalment, s'haurà d'especificar la distància per a cada grup de transport, mitjançant el mòdul de dades DISTANCE.

**22. DISTANCE:** Definirem les distàncies, per exemple en metres, per a tots els possibles moviments que realitzi un determinat grup de transport.

El seu valor no podrà ser negatiu i arrodonida si s'entren decimals (20.2 metres seran 20 metres).

Aquest mòdul bàsicament crea una matriu d'origen i destí per a les distàncies que existeixen entre cadascuna de les estacions del model. Aquestes son les distàncies de viatge per als transportadors, no per a les entitats i es poden incloure moviments dels transportadors en buit, sense entitats associades.

Per a  $n$  localitzacions d'estacions s'haurà d'especificar almenys  $n * (n-1)$  possibles distàncies i aquestes no hauran de ser necessàriament simètriques entre estacions. Addicionalment, no serà necessari afegir una distància si aquesta no es durà a terme mai, a no ser que el transportador viatgi carregat o buit. En aquest cas el sistema assumirà que aquesta serà zero automàticament i finalment si la distància és simètrica, igual en les dues direccions, només s'haurà d'especificar un parell.

**23. NETWORK:** Es defineix dins d'ell una o varies xarxes de rutes dins del model, la qual serà seguida per una sèrie de conveyors programats. Cada xarxa tindrà els seus paràmetres propis com distància, interseccions, adreces, etc., definits dins del mòdul de dades NETWORK LINK.



**24. NETWORK LINK:** Defineix les característiques per a una ruta guiada per a un conveyor entre una intersecció inicial i una final. El mòdul NETWORK farà referència a una sèrie d'enllaços per tal de definir la xarxa que seguiran els conveyors programats.

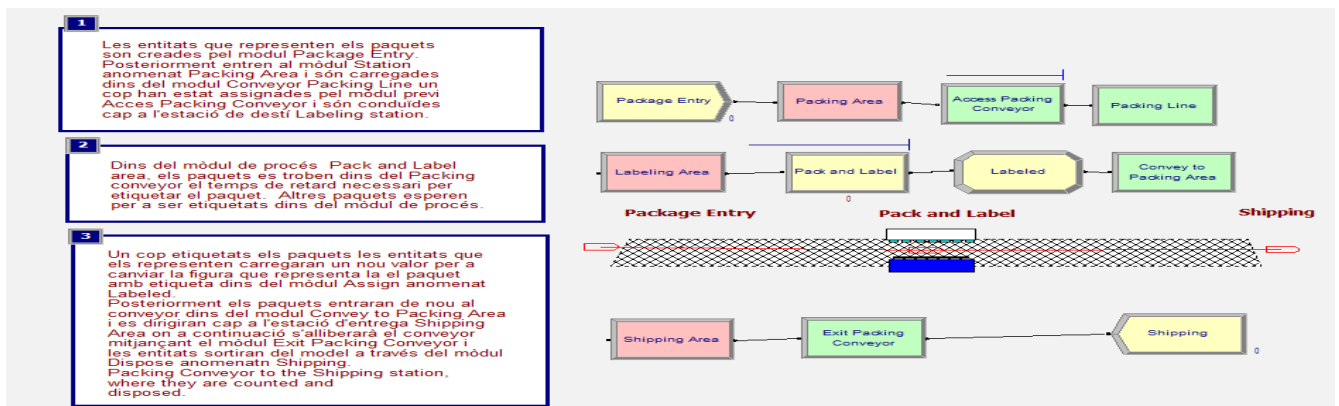
**25. ACTIVITY AREA:** Es una àrea que pot estar associada a una o més estacions, les quals seran utilitzades per a recol·lectar estadístiques associades a l'estació.

Cadascuna de les ACTIVITY AREA es pot associar a una altra ACTIVITY AREA de nivell superior, la qual és utilitzada per a definir la jerarquia del sistema que s'està modelant.

Les estadístiques seran aleshores recol·lectades segons aquesta jerarquia creada.

Com a resum d'aquest important apartat, a continuació es mostra una figura amb un dels exemples que veurem dins l'apartat pràctic del projecte i que tracten alguns dels mòduls d'aquest template *Advanced Transfer*, on es pot visualitzar tota la seqüència de blocs que seran necessaris per tal de modelar un procés productiu que simula una cinta transportadora, per la qual hi passen paquets per a ser etiquetats. S'ha de tenir molt en compte els mòduls utilitzats i el seu ordre: **CREATE -> STATION -> ACCESS -> CONVEY -> STATION -> PROCESS -> ASSIGN -> CONVEY -> STATION -> EXIT -> DISPOSE**

I les assignacions que es fan dins dels mòduls de dades **CONVEYOR** :



I **SEGMENT** (on es poden veure les dues columnes on s'indiquen les distàncies entre estacions) :

Conveyor - Advanced Transfer									
	Name	Segment Name	Type	Velocity	Units	Cell Size	Max Cells Occupied	Initial Status	Report Statistics
1	Packing_Conv	Packing_Conv.Segment	Non-Accumulating	2.5	Per Minute	1	1	Active	<input checked="" type="checkbox"/>

Segment - Advanced Transfer			
	Name	Beginning Station	Next Stations
1	Packing_Conv.Segment	Packing_St	{ 2 rows }

Double-click here to add a new row.

Next Stations		
	Next Station	Length
1	Label_St	10
2	Shipping	10

Double-click here to add a new row.

### 4.5.5.2. Funcionament d'un transport fix.

Abans d'entrar en el detall de la sol·licitud de cada tipus de transport es mostra a continuació un petit resum dels principals blocs de transferència avançada que s'utilitzen per aquest tipus de transport fix.

<b>Característiques dels principals mòduls del template Advanced Transfer per a modelar sistemes de transport fix</b>	
<b>Mòdul</b>	<b>Principal característica</b>
<b>Allocate</b>	Aquest mòdul assigna una unitat d'un transportador a una entitat, però NO mou l'entitat que provoca el transport cap a l'estació destí on es trobi l'entitat sol·licitant i tampoc inicia el moviment del transportador cap al destí assignat. El transportador viatjarà buit i en estat lliure des d'on es trobi cap a l'estació destí on es troba l'entitat sol·licitant. <b>ASSIGNA UNA ENTITAT AL TRANSPORTADOR DIFERENT A LA ENTITAT CONTROLADORA I ESPERA L'ORDRE DE MOVIMENT.</b>
<b>Move</b>	Avança un transportador des d'una estació a una altra sense moure l'entitat controladora. Aquesta romandrà dins d'aquest mòdul fins que el transport tingui assignat un destí i aleshores l'entitat estarà habilitada per a moure's cap a una altra mòdul del sistema. <b>MOU EL TRANSPORT EN BUIT.</b>
<b>Request</b>	Aquest mòdul assigna a una entitat que es vol moure, una unitat d'un transportador disponible determinat. L'entitat sol·licitant romandrà en aquest mòdul fins que la seva petició de transport sigui atesa, però el transportador ja s'està movent el transportador cap el lloc on se'l demana. <b>REQUEST té el mateix efecte que si utilitzem ALLOCATE + MOVE, SENSE MOURE LA ENTITAT CONTROLADORA.</b>
<b>Transport</b>	Aquest mòdul transporta tant l'entitat controladora com el transportador en qüestió entre dues estacions. <b>MOU EL TRANSPORT CARREGAT AMB LA ENTITAT CONTROLADORA.</b>
<b>Free</b>	Aquest mòdul allibera el transportador més recentment utilitzat, el mateix es col·locarà en estat ociós en l'estació on s'ha produït la seva alliberació, en cas que no s'hagi especificat diferent en el mòdul de dades Transporter. <b>ALLIBERA LA ENTITAT CONTROLADORA QUE HA VIAJAT CONJUNTAMENT AMB EL TRANSPORTADOR.</b>
<b>Halt</b>	En aquest mòdul canvia l'estat d'una entitat transportadora i passa aquest a inactiu (Inactive). Si en el moment d'entrar una entitat a aquest mòdul, l'element de transport que ha de canviar l'estat es troba ocupat, el seu estat serà ocupat e inactiu fins que l'entitat controladora sigui alliberada. En canvi, si l'estat del transportador és ociós (Idle) quan una entitat el defineix com a inactiu mitjançant aquest mòdul, el mateix passarà a estat inactiu immediatament. El transportador en aquest estat no podrà donar servei a cap entitat sol·licitant de transport, fins que sigui activat de nou. <b>DESACTIVA UNA DETERMINADA ENTITAT TRANSPORTADORA.</b>
<b>Activate</b>	Aquells transportadors desactivats prèviament mitjançant el mòdul HALT, seran activats de nou quan una entitat de control entri a aquest mòdul. En el moment d'activament de nou del transportador, la seva variable IT passarà de valdre 2 (inactive) a 0 indicant que passa a ociós (Idle) o 1 si es troba ocupat (busy) en el cas que existeixin entitats per a ser transportades per aquest transportador. <b>ACTIVA UNA UNITAT TRANSPORTADORA PREVIAMENT DESACTIVADA PER HALT.</b>
<b>Route</b>	Transfereix una entitat cap a una estació específica, o per a la pròxima estació dins d'una ruta seqüencial d'estacions predefinides per a l'entitat. Es pot aplicar un retràs a aquesta transferència.
<b>Transporter</b>	Permet a l'usuari definir les característiques d'un grup d'unitats transportadores. Per exemple, podem canviar el nombre d'unitats del grup, la velocitat per defecte i la seva localització inicial. Addicionalment, s'haurà d'especificar la distància per a cada grup de transport, mitjançant el mòdul de dades DISTANCE. <b>HAUREM D'ESPECIFICAR LES CARACTERÍSTIQUES PER A CADA GRUP DE TRANSPORT.</b>
<b>Distance</b>	Definirem les distàncies, per exemple en metres, per a tots els possibles moviments que realitzi un determinat grup de transport. El seu valor no podrà ser negatiu i arrodonida si s'entren decimals (20.2 metres seran 20 metres). Per a n localitzacions d'estacions s'haurà d'especificar almenys $n * (n-1)$ possibles distàncies i aquestes no hauran de ser necessàriament simètriques entre estacions. Addicionalment, no serà necessari afegir una distància si aquesta no es durà a terme mai, a no ser que el transportador viatgi carregat o buit. En aquest cas el sistema assumirà que aquesta serà zero automàticament i finalment si la distància és simètrica, igual en les dues direccions, només s'haurà d'especificar un parell. <b>PER A CADA PARELL D'ESTACIONS S'HAN D'ENTRAR LES SEVES DISTÀNCIES.</b>

Un transport fix és sol·licita mitjançant el mòdul REQUEST. El qual s'encarrega d'assignar el transport a l'entitat sol·licitant i mou l'element transportador dins a l'estació on aquesta es troba esperant.

Quan l'entitat sol·licitant ha sortit del mòdul REQUEST han tingut lloc dos events:

- Se li ha assignat un transport a l'entitat sol·licitant (la capacitat de la flota disminueix en 1).
- El transport ha arribat a l'estació on es troba l'entitat sol·licitant.

Això no vol dir que ambdós events hagin ocorregut immediatament després que s'inicia la transferència. Tenen lloc només quan l'entitat, la qual ara controla el transport, entra dins del mòdul TRANSPORT.

Es pot dir doncs, que entre un mòdul REQUEST i un TRANSPORT també poden tenir lloc diferents events o activitats que es desenvolupin sobre l'entitat que controla el transport, per exemple, la càrrega.

Quan una entitat arriba a l'estació de destí aquesta continua tenint el control del transport fins que se li doni la instrucció d'alliberament mitjançant la seva entrada a un mòdul FREE.

Anàlogament al conjunt REQUEST-TRANSPORT, entre un mòdul TRANSPORT (no es troben units) i un FREE (amb un mòdul STATION abans) poden ocórrer diferents events o activitats que es desenvolupen sobre l'activitat que té el control del transport en l'estació de destí com pot ser la càrrega.

El moviment d'un transport es troba determinat per la distància entre estacions i per la velocitat que s'especifiqui (amb càrrega o sense). Les distàncies i les característiques del transport s'especifiquen dins dels mòduls de dades **DISTANCE** i **TRANSPORTER**, respectivament.

### 4.5.5.3. Sol·licitud de transport lliure. Mitjançant Conveyors.

Un conveyor és un camió, "BELT" o qualsevol aparell que carrega elements i els transporta en moviment sobre un camí prefixat. Normalment, els conveyors son utilitzats per a transportar un gran volum de materials a través de curtes o mitjanes distàncies.

Els conveyors també es poden denominar transportadors lliures i cada conveyor es troba dividit en vèries cèl·lules que representen, cadascuna, la menor unitat d'espai necessària per a emmagatzemar una entitat.

Quan una entitat és carregada sobre o dins d'un conveyor ocupa les cèl·lules que aquesta necessita i si ja no queden cel·les lliures s'impedeix la càrrega d'altres entitats cap el conveyor, fins que vagin apareixen nous espais o cel·les buides.

Les informacions bàsiques per a definir-los son la velocitat del dispositiu i la distància entre les estacions que el connecta i la velocitat típica dels conveyors es mou en un rang d'entre 20 y 80 peus(feet) per minut fins als 500 peus per minut.

Els conveyors poden anar guiats o lliscar gravitativament i els seus diferents sistemes es poden modelar partint de la següent classificació (can be roughly classified):

- **Accumulating** : Bloquegen el progrés de les entitats que s'aproximen quan el conveyor es troba aturat a una estació, mentre les altres continuen el seu moviment normalment.
- **Nonaccumulating**: El moviment de totes les entitats s'interrompeix quan qualsevol d'elles s'atura en una estació en espera d'accedir al conveyor.
- **Fixed spacing**: Els elements són ubicats en un espai fix dins del conveyor o els elements son conduits IN A BUCKET OR BIN.
- **Random spcing**: Els elements són ubicats aleatòriament dins del conveyor AND TAKE UP espai sobre el conveyor.

Diàriament fem servir conveyors en el nostre estil de vida. Per exemple, THE BELT CONVEYOR AT A GROCERY STORE RESEMBLES AN ACCUMULATING CONVEYOR. Una escala és un "Fixed spacing nonaccumulating conveyor", doncs els esgraons són com a "BUCKET" per al peus de la única persona que hi puja. Les persones que es mouen per un aeroport RESEMBLE NONACCUMULATING RANDOM ESPACING conveyors.

Quan es dissenyen sistemes amb conveyors, un nombre de mesurables característiques s'han de tenir en compte, per exemple:

- **Throughput capacity**: Nombre d'unitats carregades per unitat de temps.
- **Delivery Time**: Temps que es triga a moure els element carregats des del seu origen.
- **Queue lenghts**: Tamany de les cues d'acumulació que es fixaran a cada conveyor quan aquest fa funcions d'acumulació.
- **Number of carriers used or space used**: Nombre d'espais ocupats dins del conveyor.

A l' igual que els transportadors fixes, els conveyors poden estar en diferents estats: en espera, quant esperen la sol·licitud d'una entitat per a ser movimentada; ocupats, quant estan transportant una o més entitats des d'una estació a una altra; inactius, quant apareix alguna fallida o el mateix es troba en manteniment.

Al contrari dels transportadors fixes, aquests es mouen com a conjunt d'entitats, inclús l'animació es fa conjunta com a una sol'entitat.

El conveyor es carrega sense un tractament especial, simplement s'utilitza un mòdul PROCESS o un DELAY amb un temps determinat per a la càrrega entre cadascun dels elements que s'hi transportaran. El procés invers serà utilitzat per a la seva descàrrega en l'estació d'arribada i el modelatge del sistema complet, serà motiu d'estudi dins la part pràctica d'aquest projecte.

Però en aquest apartat es mostra a continuació un petit resum dels principals blocs del panell de transferència avançada, usats per al modelatge de conveyors.

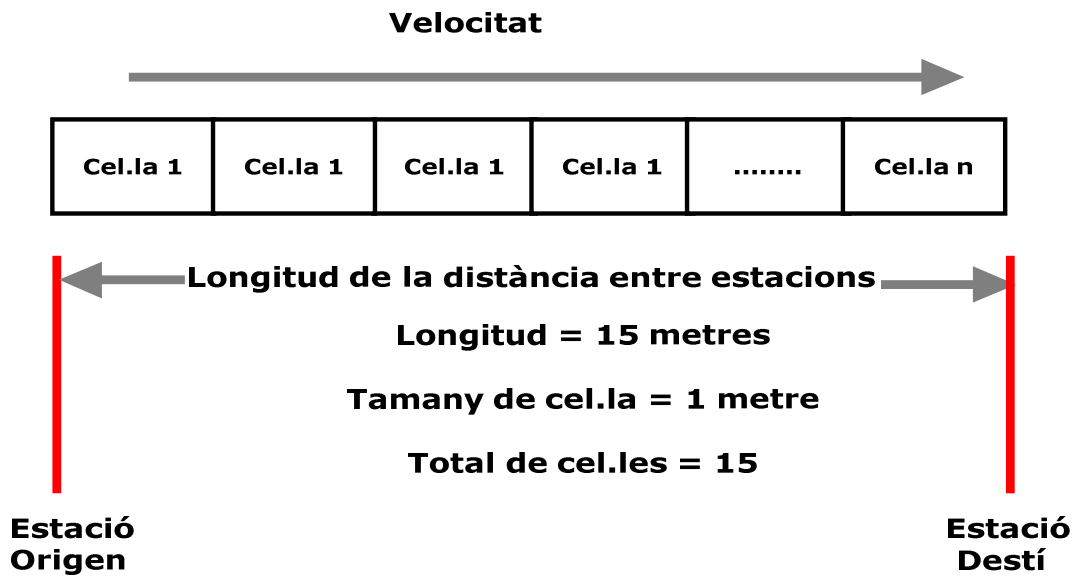
<b>Característiques dels principals mòduls del template</b>	
<b>Advanced Transfer per a modelar sistemes de transport lliure</b>	
<b>Mòdul</b>	<b>Principal característica</b>
<b>Access</b>	Aquest mòdul assigna una o més cel·les d'un conveyor a una entitat per tal de moure-la d'una estació a una altra. L'atribut Entity.Station de l'entitat ha de tenir un valor vàlid quan aquesta intenta accedir a un conveyor.
<b>Convey</b>	Aquest mòdul és el responsable del transport d'una entitat des d'una estació fins a una altra mitjançant un conveyor. El temps que trigarà el transport d'una estació a una altra està basat en la velocitat del conveyor, especificada dins del mòdul de dades CONVEYOR, i la distància entre estacions s'especificarà dins del mòdul de dades SEGMENT. Quan una entitat entra en aquest mòdul, el seu atribut Entity.Station és definit amb l'estació de destí. Si el tipus d'estació de destí és definit com a "Sequential", la propera estació vindrà predefinida pels atributs Entity.Sequence i Entity.Jobstep de l'entitat que acaba d'entrar al mòdul.
<b>Exit</b>	En aquest mòdul s'alliberen les cel·les ocupades per l'entitat dins d'un conveyor. Si una altra entitat està esperant aquestes cel·les disponibles en la mateixa estació on es produeix l'alliberament, aquesta accedirà al conveyor.
<b>Start</b>	Aquest mòdul recupera un conveyor del seu estat inactiu. En el moment d'activació de nou del conveyor, l'estat pot venir d'inactiu (3), d'espera o Idle (0), d'actiu o Moving (1) o que el mateix es trobi bloquejat o Blocked (2). En aquest moment existeix la possibilitat d'alterar la velocitat del conveyor.
<b>Stop</b>	Aquest mòdul passa a inactiu el conveyor al que fa referència. Quan una entitat entra en aquest mòdul, el conveyor indicat s'atura immediatament i la seva variable interna d'estat ICS passa a valer 3 (inactive), independentment del tipus de conveyor o el nombre d'entitats que estiguin sent conduïdes.
<b>Conveyor</b>	En aquest mòdul canvia l'estat d'una entitat transportadora i passa aquest a inactiu (Inactive). Si en el moment d'entrar una entitat a aquest mòdul, l'element de transport que ha de canviar l'estat es troba ocupat, el seu estat serà ocupat e inactiu fins que l'entitat controladora sigui alliberada. En canvi, si l'estat del transportador és ociós (Idle) quan una entitat el defineix com a inactiu mitjançant aquest mòdul, el mateix passarà a estat inactiu immediatament. El transportador en aquest estat no podrà donar servei a cap entitat sol·licitant de transport, fins que sigui activat de nou. <b>DESACTIVA UNA DETERMINADA ENTITAT TRANSPORTADORA.</b>
<b>Segment</b>	Aquells transportadors desactivat prèviament mitjançant el mòdul HALT, seran activats de nou quan una entitat de control entri a aquest mòdul. En el moment d'activament de nou del transportador, la seva variable IT passarà de valdre 2 (inactive) a 0 indicant que passa a ociós (Idle) o 1 si es troba ocupat (busy) en el cas que existeixin entitats per a ser transportades per aquest transportador. <b>ACTIVA UNA UNITAT TRANSPORTADORA PRÈVIAMENT DESACTIVADA PER HALT.</b>

Si volem simular el transport de grups d'elements entre dues estacions i l'espai per a ubicar el transport és important per a la operativa del sistema, es pot utilitzar la construcció anomenada ARENA's conveyor constructs.

Dins ARENA, un conveyor pot ser un element de transport per transferir o moure entitats a través d'un camí prefixat amb punts de càrrega i descàrrega intermitjos. Cada entitat pendent de càrrega haurà d'esperar en un espai suficient dins del conveyor abans de ser carregada i comenci el seu transport.

Bàsicament, el model construït per als conveyors divideix l'espai entre estacions en unitats denominades celes. El conveyor, aleshores es converteix en un moviment entre celes de la mateixa longitud i quan arriba una entitat, aquesta es mourà a través de cada cel·la de longitud determinada, la suma de totes les longituds donarà la velocitat de transport entre l'estació origen i destí.

En la figura següent es pot veure el fet que si cada cel·la ocupa 1 metre i el total de la longitud del conveyor és de 15 metres, aleshores tenim 15 cel·les. El tamany d'una cel·la és la mínima porció d'aquest que pot ser ocupada per a una entitat.



Quan modelem conveyors el tamany de l'entitat també es té en compte. Per exemple, si pensem en persones que pugen una escala mecànica, poden ocupar dos esgraons si duent una maleta i d'altres ocupen només un, doncs no duen equipatge.

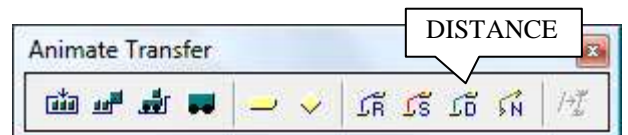
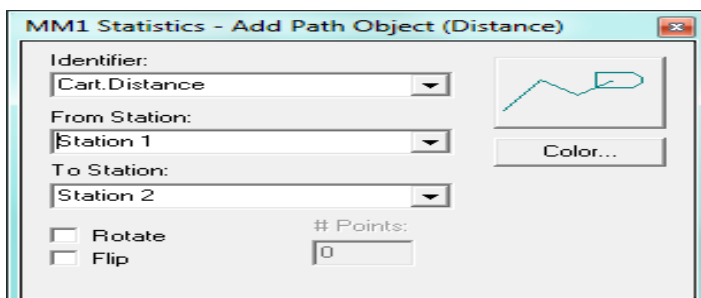
Aleshores una entitat ha de poder agafar les celes contigües que necessiti per tal de poder mantenir la cua amb la resta d'elements a transportar.

#### 4.5.5.4. Animació del camí entre estacions.

Tot transport o grup de transports ha de rebre un nom, per exemple, Carts, aleshores tots els camins (distàncies) associats a aquests rebran com a nom Carts.Distance.

Per connectar l'estació de sortida i la d'arribada utilitzarem el connector per a distàncies anomenat DISTANCE, el qual es crida mitjançant el botó del mateix nom que es troba integrat dins de la barra anomenada ANIMATE TRANSFER.

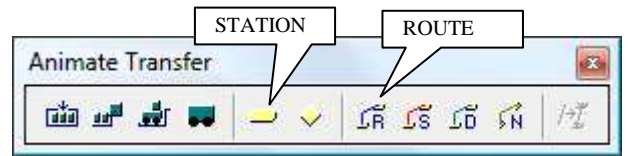
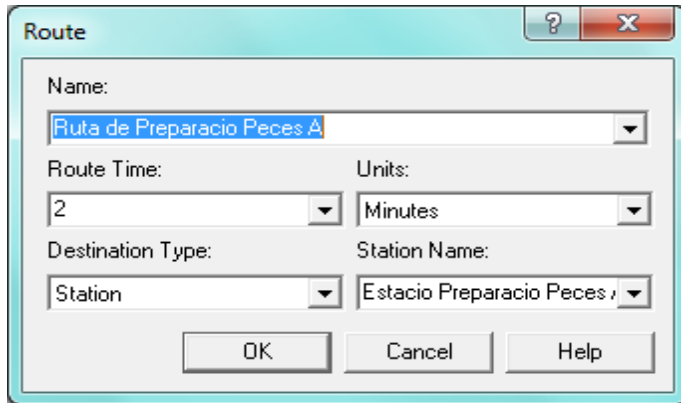
Quan es pressiona aquest botó apareix el següent



formulari, on entrarem com a identificador el nom del camí (Carts.Distance) i escollirem estació origen i destí. El punter del ratolí es transforma en una creu i ja podem procedir a la unió entre les respectives estacions. Primer

marquem l'estació d'origen, prement durant tot el recorregut un vegada el ratolí per fixar un punt de pas i finalment dos clics de ratolí per a finalitzar la ruta i que apareix-hi el gràfic de l'estació destí.

Existeix una altra mètode d'animació vist en el projecte anterior [9], dins l'apartat 9.4 sota la realització del pas 4 de l'exemple 6: Assemblatge i prova de peces electròniques (Pàgina 98). En el mateix s'utilitza un mètode més manual, assignant a cada estació origen i destí, una figura d'estació simple prement el botó STATION de la barra ANIMATE TRANSFER i unint-les mitjançant la ruta que es crea quan premem el botó ROUTE de la mateixa barra.



D'aquesta forma estem obligats a utilitzar el mòdul ROUTE dins el model lògic, dins del qual haurem d'indicar com a Destination Type el tipus Station i quin és el nom de l'estació gràfica de destí dins de camp Estation Name .

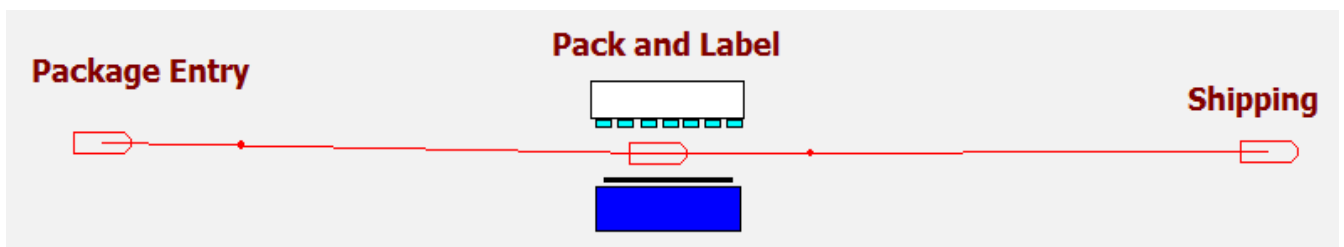
Si el que volem és connectar dues estacions connectades mitjançant un conveyor el que utilitzarem per connectar l'estació de sortida i la d'arribada serà un segment.

Utilitzarem el connector per a segments anomenat SEGMENT, el qual es crida mitjançant el botó del mateix nom que es troba integrat dins de la barra anomenada ANIMATE TRANSFER.



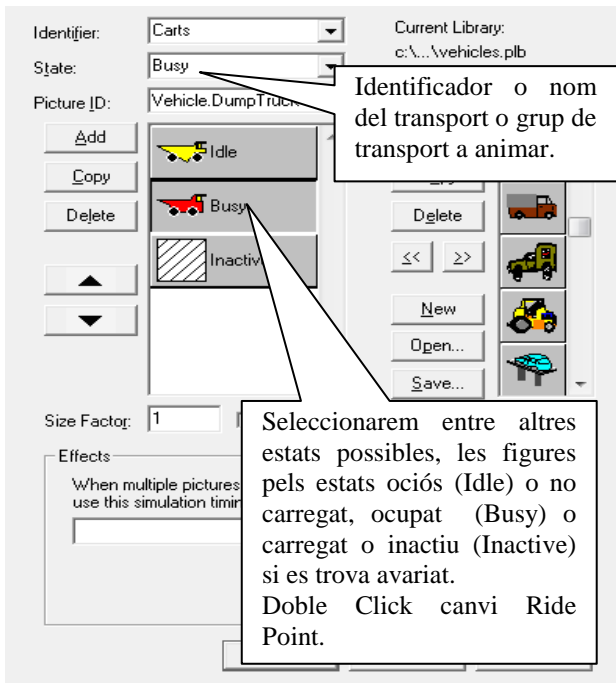
Fent clic sobre el botó SEGMENT s'obrirà un quadre de diàleg semblant al del botó DISTANCE i el punter del ratolí es transforma en una creu. Però ara ja podem procedir a la unió entre les respectives estacions i aquestes poder ser més de dues, traçant entre elles la ruta que seguirà el conveyor.

El segment en el cas de la figura, connectarà tres estacions per les quals passarà el conveyor , origen (Package Entry), intermèdia (Pack and Label) i destí (Shipping).

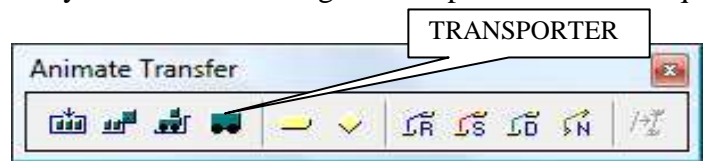


### 4.5.5. Animació de l'element transportador.

Es pot veure l'animació de les entitats que segueixen una ruta determinada mitjançant l'associació de la



seva figura a la ruta, vista en el projecte anterior [9], dins l'apartat 9.4.3.4. sota la realització del pas 4 de l'exemple 6: Assemblatge i prova de peces electròniques (Pàgina 109). Sempre es pot modificar la figura mitjançant la entrada del menú principal "Edit -> Entity Pictures..." i les figures sempre tenen el nom que



comença per Picture.XXXX.

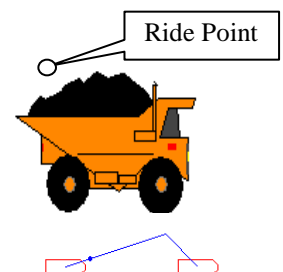
En canvi el dispositiu o mitjà de transport utilitzat pot ser representat mitjançant una figura, però la mateixa no es troba a nivell d'entitats i l'assignació de la figura es fa de forma anàloga a l'animació de recursos i també

pot venir d'una llibreria interna de l'entorn ARENA o be predefinida per un gràfic extern. L'animació de l'element transportador que representa a tot transport o grup de transports, per exemple Carts, serà associat directament a tots els camins (distàncies) associats en l'apartat anterior i que han rebut com a nom Carts.Distance.

La figura es defineix mitjançant la icona anomenada TRANSPORTER de la barra ANIMATE TRANSFER.

Quan premem el botó TRANSPORTER apareixerà el formulari per animar el recurs de transport i haurem de seleccionar la figura que representa al grup de transport, per exemple Carts, quan el mateix es trobi ocupat (Busy).

Podem variar el punt d'anclatge on es situarà l'entitat que transporta la figura si fem doble clic sobre la casella on es troba l'animació del transport ocupat (Busy). Aleshores apareixerà la figura dins l'editor de figures ARENA i mitjançant la opció de la barra de menú d'aquest: "Object -> Ride Point", es mou on es vulgui el punt d'anclatge de la figura que representa l'entitat de transport durant la trajectòria del moviment. Quant es selecciona aquesta opció, el punter del mouse es transforma en una creu per tal d'indicar en quin lloc de la figura es desitja que s'hi adhireixi la figura a la ruta especificada en l'apartat anterior.





#### 4.5.5.6. Animació de l'element conveyor.

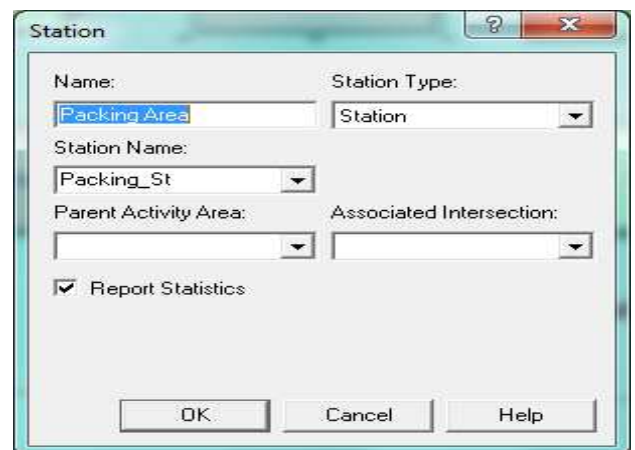
Cal dir que l'entitat associada a cada estació en el moment de l'execució és la que portarà l'atribut de la figura que seguirà la ruta del segment i aquesta es pot canviar mitjançant un mòdul Assign en temps d'execució, accedint a la figura que carregarà l'entitat per defecte mitjançant el mòdul de dades bàsic ENTITY o modificant la mateixa obrint l'editor de figures del model mitjançant l' entrada del menú principal "Edit -> Entity Pictures...".

Podem canviar la figura per exemple quant s'ha passat per una estació intermèdia per a simular un canvi en el procés.

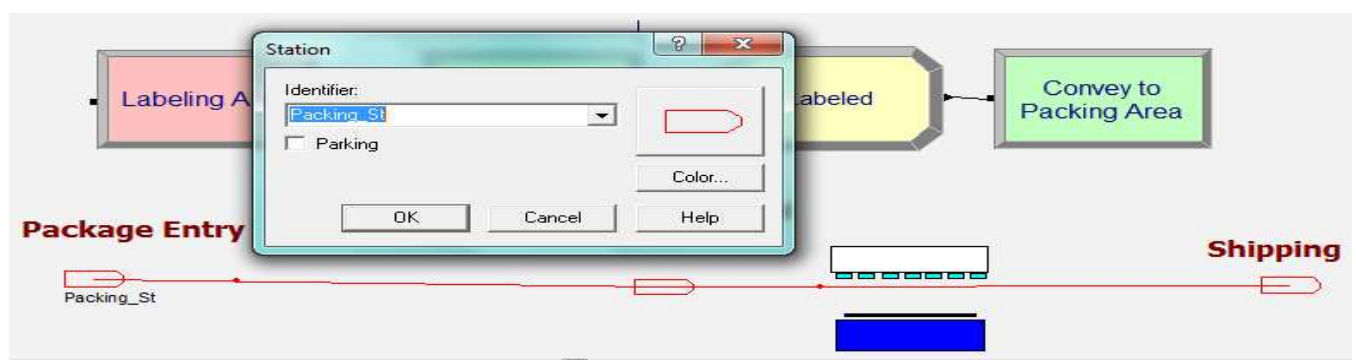
També és important destacar que el nom de cada estació, assignat dins del seu mòdul lògic STATION, no és el nom que apareixerà dins de les estacions connectades pel SEGMENT, les quals estan referenciades pel nom que hi donem a l'estació dins el mateix mòdul sota la opció Station Name.

A les dues figures següents es comprova aquest fet, el qual pot provocar errors a l'hora de graficar el model.

La figura de la dreta mostra el mòdul Station, el qual té com a nom Packing Area i en canvi el nom de l'estació per al segment es dirà Packing\_St.



Aquest últim serà el nom de l'estació dins la ruta especificada, tal i com es mostra a la figura inferior.

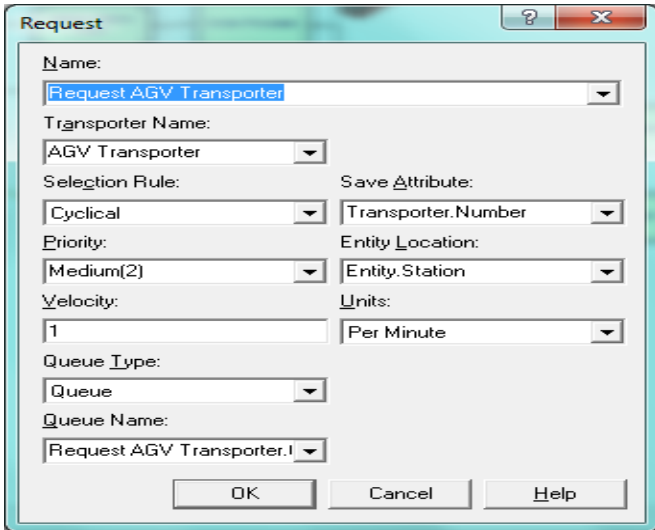


També direm que les estacions no poden animar recursos, i l'animació aquests han d'anar sempre associats al mòdul PROCESS que els utilitza. Aleshores al gràfic no es veuran en temps d'execució les figures que representen a les estacions, al seu lloc en canvi es pot veure l'animació dels estats del recurs (Idle, Busy, Inactive) i la figura associada a cadascun d'ells simularà l'arriba el procés amb l'arribada de les peces per al seu tractament.

Aquesta simulació serà tractada en detall dins dels exercicis d'aquest projecte.

### 4.5.5.7. Modelatge i Animació del transport mitjançant vehicles filoguiats. AGV's.

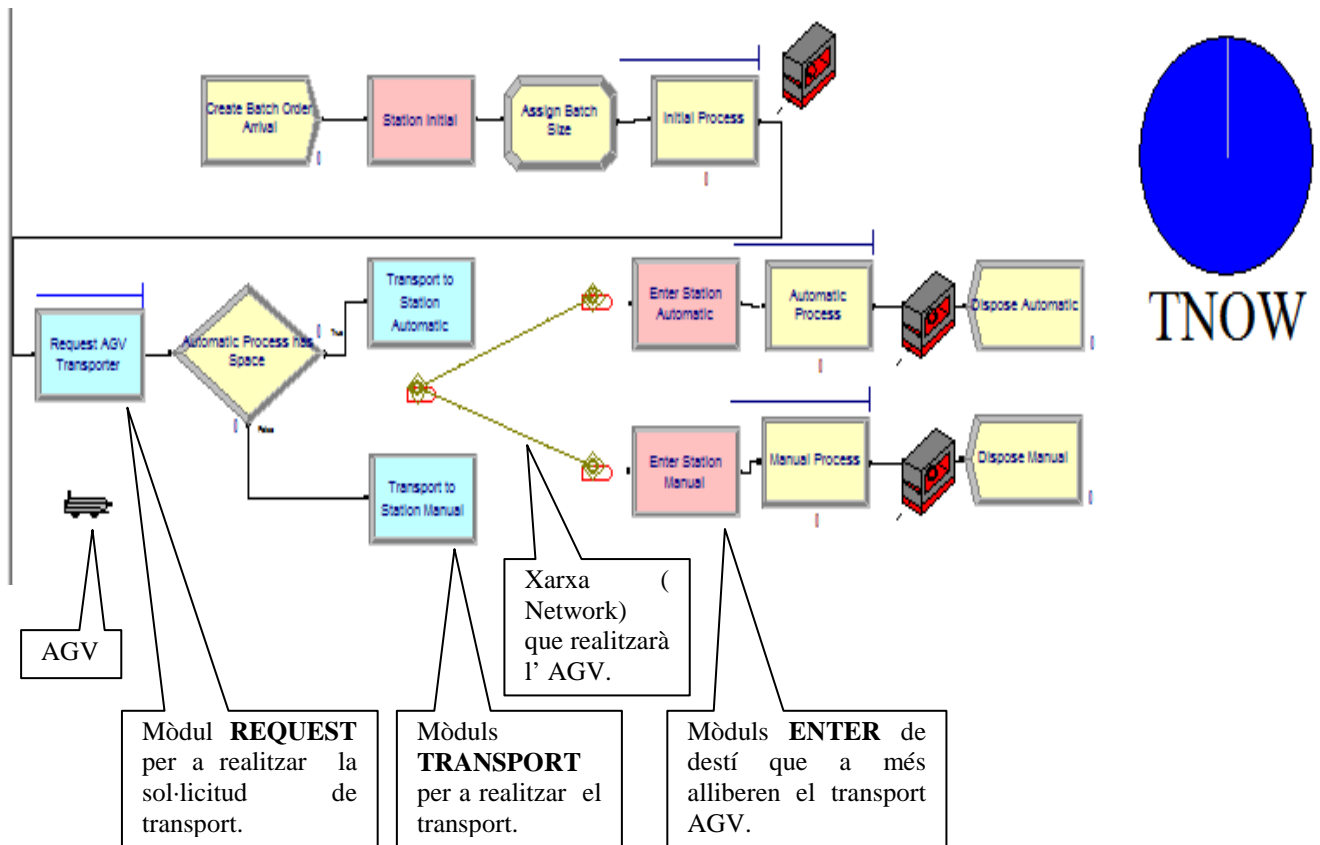
ARENA a partir de la seva versió 8.0 va introduir un tipus d transport especial, els vehicles filoguiats o AGV's (Automatic Guide Vehicle).



Aquest transport es modelitza mitjançant un mòdul REQUEST Aquest mòdul assigna a una entitat que es vol moure, una unitat d'un transportador disponible denominat AGV Transporter. Mitjançant una entitat especifica del transportador o una regla d'assignació predeterminada que en aquest cas serà cíclica (Cyclical).

L' entitat de control del transport emmagatzemarà el número de l'element transportador (atribut Transporter.Number), així com l'estació a la qual i hi vol anar (atribut en el seu atribut Entity.Station) i romandrà a aquest mòdul fins que la seva petició de transport sigui atesa i pugui iniciar el seu moviment cap a ella.

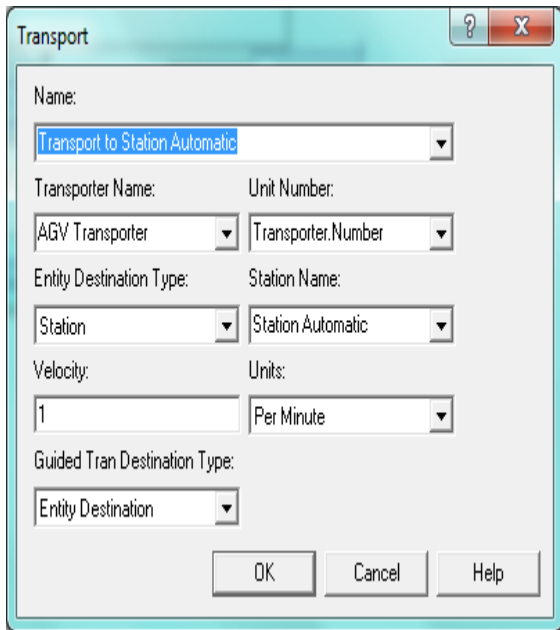
A continuació s'exposa un model d'exemple, el qual serà treballat amb més detall dins dels exercicis pràctics.



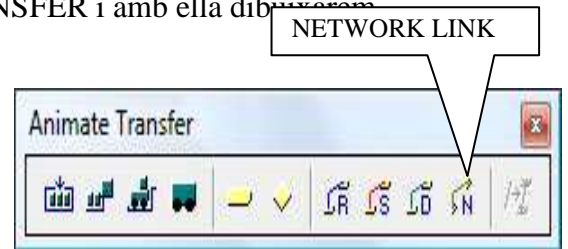
En el model es veu la seqüència de mòduls que faran falta per a realitzar el transport mitjançant un AGV:

**REQUEST → TRANSPORT → ENTER**

El mòdul TRANSPORT, defineix el transportador AGV, la seva velocitat inicial i la seva posició inicial

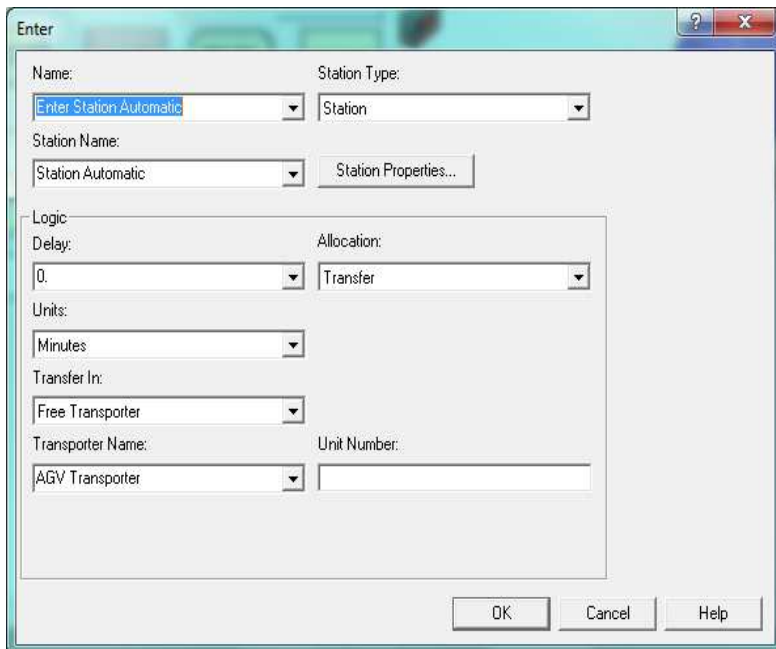
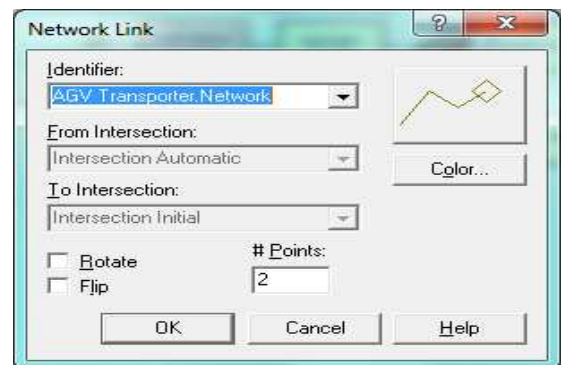


i el connecta cap a la xarxa de transport (Network). Aquesta transportador conduirà també a l'entitat de control per la xarxa de transport, la qual serà graficada mitjançant la icona NETWORK LINK que trobem a la barra ANIMATE TRANSFER i amb ella dibuixarem les rutes que seguirà el transport cap a les estacions destí.



En l'exemple anterior, l'AGV parteix sempre d'una estació central i es mourà mitjançant moviments automàtic d'anada i tornada entre aquesta i dues a cada extrem a dreta i esquerra.

En l'exemple anterior, l'AGV



Finalitzant cada extrem amb les estacions destí mitjançant un mòdul ENTER. El quals són utilitzats per alliberar els elements transportadors o AGV de l'entitat de control.

Quant una entitat arriba al mòdul ENTER, un retràs de descàrrega por ser seleccionat (0 en aquest cas) i algun equipament de transferència que s'hagi utilitzat (AGV

*Transporter*) per fer-la efectiva pot ser alliberat seleccionant la opció *Free Transporter* udins del cap *Transfer In*.

## 4.6. GENERACIÓ I ANÀLISI DE REPORTS.

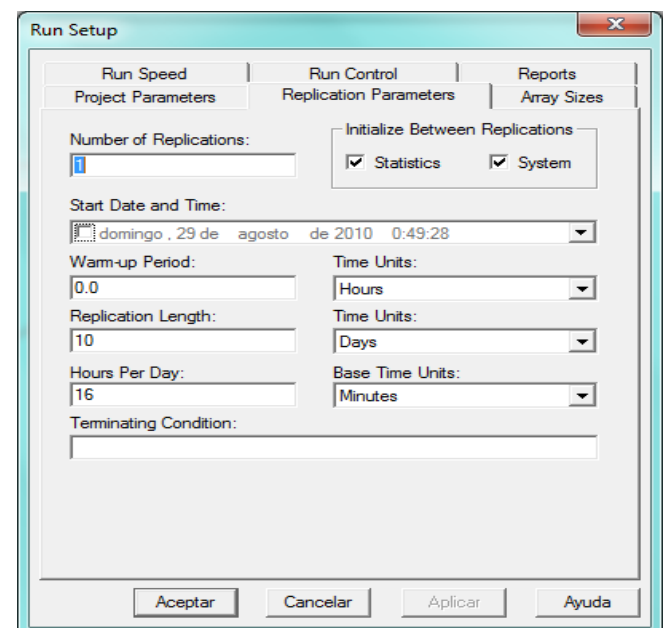
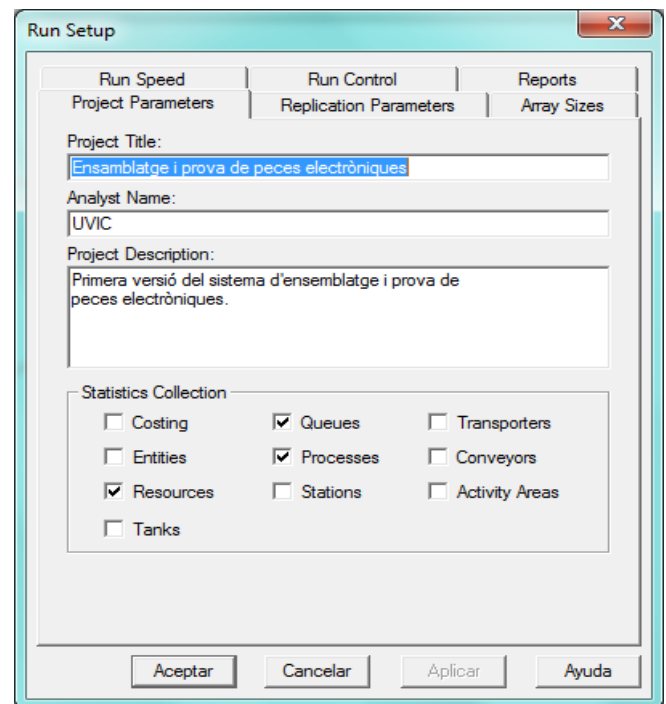
Una vegada s'ha construït el model de simulació s'han de fixar els paràmetres necessaris per al seu funcionament. Dins ARENA es poden incloure com a paràmetres del projecte i segons el model: la informació del model, el nom de l'analista i el tipus d'estadístiques que es volen recollir.

Per altre part, el nombre de rèpliques, el període d'estabilització, la longitud de la simulació i les unitats bàsiques de temps, entre d'altres aspectes, s'inclouen com a paràmetre de la simulació.

El control de simulació es fa mitjançant la barra de menús i mitjançant les instruccions "Run -> Setup". Vist dins l'apartat 3.5 el menú setup, aquí només indicarem que hem d'accedir dins del quadre de diàleg Project Parameter, entrant així dins dels paràmetres del projecte i activar les caselles de verificació corresponents als apartats sobre els quals es volen recollir les estadístiques (Cost, cues, transports, entitats, processos, conveyors i recursos). Si no es vol treballar amb costos, es recomana no activar aquesta casella si no volem recarregar l'informe, fent més difícil la seva interpretació.

Com a element important, entre d'altres, és tracta d'activar la casella *Statistics*. La mateixa es troba dins la pestanya *Replication Parameters* i sota la descripció *Initialize Between Replications*, si no volem que es barregin dades de totes les simulacions en el moment de fer el càlcul de les estadístiques. Aquí també trobem la pestanya *System*, si no s'activa cada rèplica no començarà amb 0 entitats, si no que partiria de les condicions finals de l'anterior.

També podem aturar la simulació en qualsevol moment, si afegim una expressió dins del camp *Terminating Condition*, per exemple, quan s'ha acabat un cert nombre de peces d'un tipus.

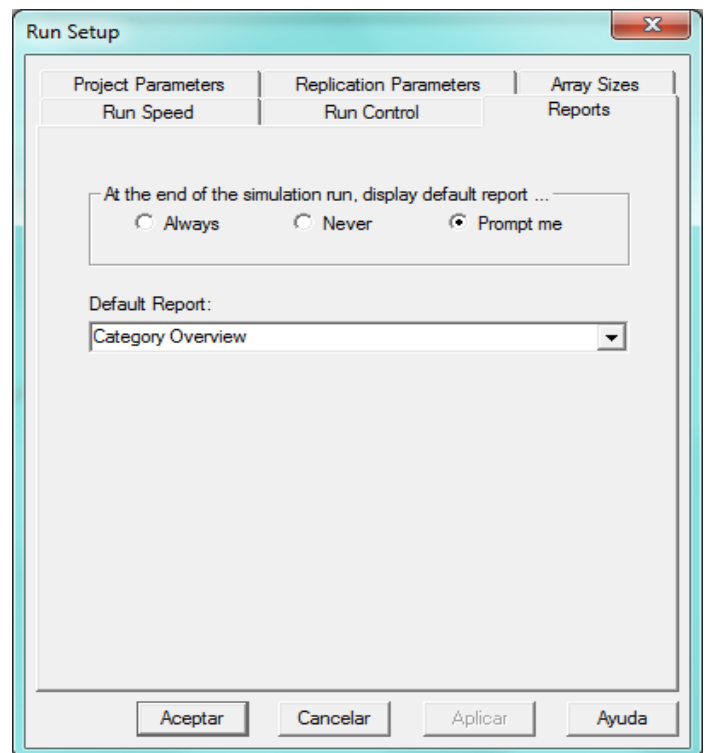


Els reports es constitueixen també com de gran interès per l'analista. Mitjançant la pestanya report de la opció Setup, accedim al seu quadre de diàleg, en el qual es defineix el report predeterminat i la manera com aquest apareixerà.

Bàsicament li diem al sistema si al final de la simulació ens ha de mostrar el report sempre (Always), mai (Never) o que ens ho preguntí (Prompt me).

A continuació es mostra un menú desplegable que permet seleccionar el tipus de report que es vol com a predeterminat, d'entre els següents:

Opcions default report
Category overview
Category by replication
Entities
Frequencies
Processes
Queues
Resources
Transfer
User specified
SIMAN summary report (.out file)



El SIMAN summary report es genera dins d'un arxiu de text i per a la seva creació no és necessita una base de dades.

Per això, quan es fixa com a predeterminat sortirà una casella en la qual podrem deshabilitar la generació de la base de dades per al report.

Base de dades en Accés que es generarà sempre al final del report i on es guardaran tots els elements que l'integren. És molt potent per l'analista, doncs pot fer servir aquesta base de dades per tal de connectar les dades amb d'altres sistemes d'informació per tal d'ampliar les funcionalitats del sistema.

Si s'escull com a report predeterminat el resum general (Category overview) tindrem un arbre desplegable a l'esquerra on es pot anar d'una categoria a una altra mitjançant un explorador de reports.

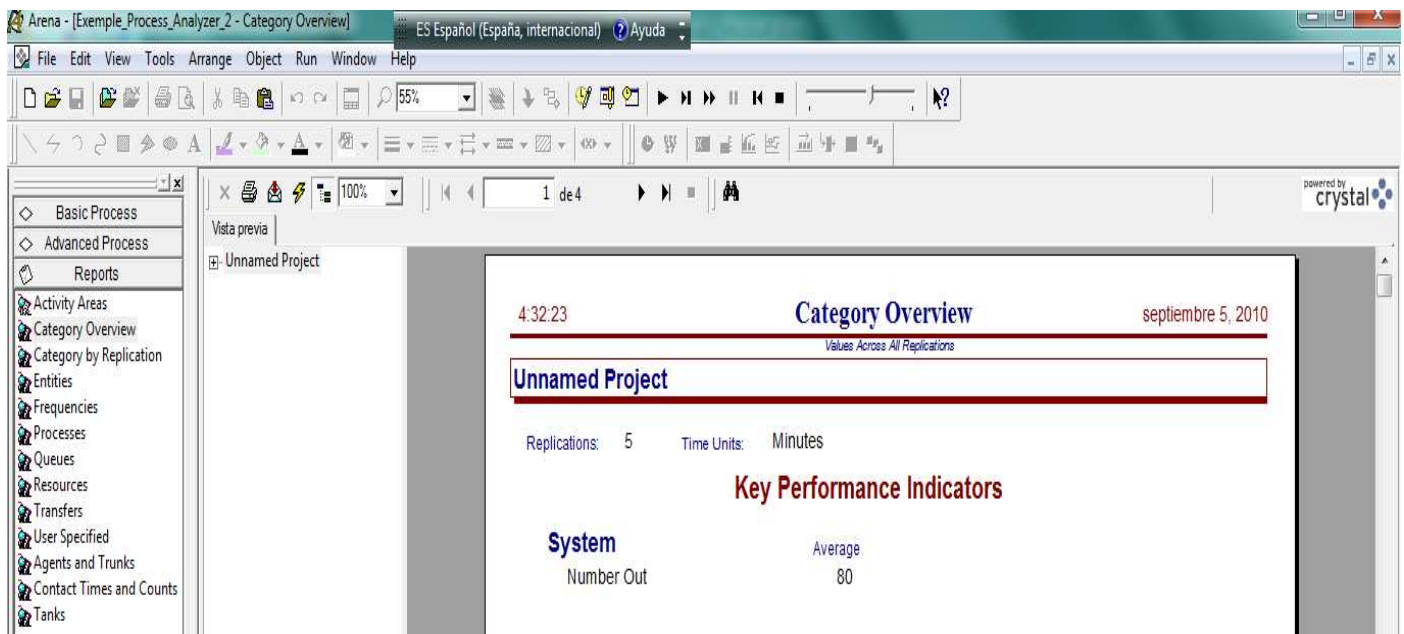
Dins de cadascuna de les categories es passa d'un ítem a una altra mitjançant aquest explorador, al igual que es faria l'explorador del propi Windows.

## 4.6.1. TRACTAMENT DE REPORTS.

El primer report que encapçala la llista anterior es el resum general (Category overview). Aquest conté tota la informació sobre les entitats, els recursos, els processos, les transferències i demés.

La informació es mostra de forma numèrica i gràfica, base d'aquest projecte i també es visualitzaran els promitjos i les desviacions dels diferents temps analitzats.

Per mitjà de l'explorador, es poden accedir a les diferents seccions del report, tal i com es mostra a la figura següent:



Per a visualitzar els informes referents a les entitats, es desplega la secció corresponent i s'escull l'ítem específic que es desitja observar o analitzar mitjançant un informe.

**Important: En els casos en els qual es presenten varies rèpliques, el resum general mostra les dades promig de totes les rèpliques i si es vol veure per a cada rèplica els reports per categoria s'ha de recórrer als reports individuals dins la plantilla de reports que el sistema incorpora per defecte.**

Si es desitja veure sense gràfics informació sobre les entitats, s'ha de seleccionar aquesta categoria dins de la plantilla de reports i apareix, aleshores, a la primera fulla del report el resum de la informació per a tots els tipus d'entitats.

Dins l'apartat 7.2.8 anomenat Interpretació dels Informes del projecte anterior <sup>[9]</sup> (pàgina 30) es pot trobar una bona guia per a extreure conclusions i aprendre a navegar a través dels informes automàtics que genera ARENA un cop ha finalitzat la simulació.

## 4.6.2. RECOL·LECCIÓ D'ESTADÍSTIQUES.

És possible que sigui necessari realitzar un recompte d'entitats en determinat lloc del model, mesurar el temps promig d'arribades d'entitats a una estació o procés en un punt mig del model o simplement mesurar el temps que triguen les unitats en anar d'un lloc a una altra mitjançant estadístics d'usuari.

Aquests es recullen amb mòduls Record que permeten recol·lectar tres tipus d'estadístics i mostren els resultats dins dels reports sota la categoria d'estadístiques definides per l'usuari (**User Defined**).

- **Estadístiques de temps discrets o Tally statistics:** basades únicament en l' observació d'algun atribut de les entitats (temps de permanència mitjà d'una entitat).
- **Estadístiques de temps continu o Time-persistent stadistics:** estadístics en que el seu valor depèn del temps (número mitjà de clients en el sistema).
- I el tercer tipus són els **comptadors**.

Especificacions del mòdul bàsic RECORD	
Camp	Descripció
<b>Name</b>	Nom del mòdul, ha de ser únic
<b>Type</b> (Tipus De l'estadístic que es recol·lectarà)	<b>Count:</b> Incrementa o disminueix el valor d'un estadístic en una certa quantitat especificada. <b>Entity Statistics:</b> Genera estadístiques generals de les entitats que entren al mòdul. <b>Time Interval:</b> Calcula i guarda la diferència entre el temps actual i el de simulació quan arriba una entitat. <b>Time Between:</b> Emmagatzema el temps entre entrades d'entitats dins del mòdul. <b>Expression:</b> Emmagatzema un valor en funció d'una expressió generada per l'Expression Builder.
<b>Count</b>	Aquesta funció permet que el mòdul realitzi un conteig d'entitats a mesura que aquestes passen al seu través.
<b>Value (Count)</b>	Activa el camp Value, quantitat en la que s'incrementa el comptador per entitat comptada, és a dir, de 1 en 1, etc.
<b>Counter name</b>	Nom que se li dóna al comptador per tal que sigui identificat en els reports.
<b>Entity Statistics</b>	Recol·lecció de les estadístiques d'una entitat fins la instància del model. Si diferents mòduls RECORD tenen aquest mateix tipus d'acció, només la informació recol·lectada en cadascun d'ells es barreja i apareix als reports.
<b>Time interval</b>	Recol·lecta d'estadístiques relacionades amb un interval de temps entre dos punts del model. L'entitat ha de contenir un atribut que inclogui el temps en el qual va passar pel primer punt, per tal que en aquest mòdul es compari amb el temps actual (Tnow) i així es determini l'interval de temps transcorregut.
<b>Attribute name</b>	Nom de l'atribut que conté el temps en el que l'entitat va passar pel punt inicial de l' interval.
<b>Tally name (Time interval)</b>	Nom de l'estadística en la qual es recol·lecten les dades obtingudes quan s'escull el tipus Time Interval. Si està en blanc agafa el nom del mòdul per defecte.
<b>Time between</b>	Recol·lecta de temps entre arribades d'entitats al mòdul. La primera entitat que arriba no es té en compte per al càlcul, donat que aquesta és la que inicia el conteig (n-1 dades de temps entre arribades).
<b>Tally name (Time between)</b>	Nom de l'estadística en la qual es recol·lecten les dades de temps entre arribades obtingudes quan s'escull el tipus Time Between. Si està en blanc agafa el nom del mòdul per defecte.
<b>Expression</b>	Calcula una expressió per tal que sigui reportada com a estadística. Un expressió pot ser, segons les necessitats de l'analista, el valor d'un atribut multiplicat per una altra o per una constant, el mínim valor entre diferents atributs, etc.
<b>Tally name (Expression)</b>	Nom de l'estadística en la qual es recol·lecten les dades calculades segons una expressió quan s'escull el tipus Expression. Si està en blanc agafa el nom del mòdul per defecte.

**Notes importants.**

- Les estadístiques observables o TALLIES llencen un promig aritmètic normal d'una variable, es a dir, la suma del total de dades dividides entre el nombre d'observacions.
- Les estadístiques de variables donen un promig ponderat en base al temps, es a dir, la suma dels valors que adopta la variable multiplicats pel temps en el qual aquesta va estar en aquest valor (pes) i dividit pel temps total.
- Les estadístiques de les entitats (Entity Statistics) corresponen a tots els temps segons la seva categoria (temps total, valor afegit, espera, etc.). Les estadístiques d'utilització de recursos, transports, cues, entre d'altres, es recol·lecten normalment pel fet que son variables globals.
- La funció expression permet emmagatzemar dins d'una estadística observable (Tally) el valor que té una determinada expressió quan una entitat passa per aquest mòdul RECORD. S'ha de tenir en compte que, al final, el valor promig resulta de la suma del valor de cada observació més el nombre d'entitats (observacions) que es tenen.

Una expressió pot ser una relació matemàtica entre valors d'atributs d'una entitat o de variables del model. Per exemple, si les entitats en un cert punt, representen una làmina en forma triangular i tenen dos atributs anomenats base i alçada, l'expressió que es calcularà podria ser l'àrea mitjançant la fórmula  $\text{base} * \text{altura} / 2$ .

La recol·lecció d'estadístiques mitjançant el mòdul RECORD pren importància quan es desitgen analitzar dades amb les eines que incorpora l'ambient de treball ARENA, com el Process Analyzer o el Output Analyzer.

#### 4.6.2.1. Estadístiques recol·lectores del conteig d'entitats.

Per tal d'analitzar l'ús del mòdul RECORD començarem modelitzant un sistema en el que hi han quatre processos en sèrie: A, B, C i D. Els mateixos requereixen de quatre recursos també anomenats A, B, C i D, respectivament.

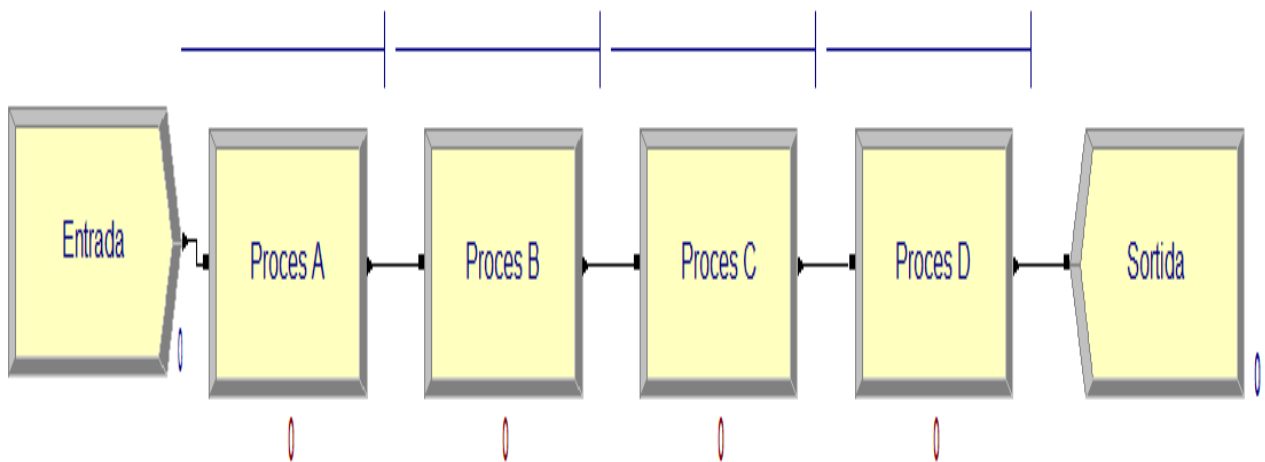
Els diferents temps de procés en minuts es mostren en la taula següent:

Procés / Recurs	Temps
A	Normal (8,0.89)
B	Uniforme (4,12)
C	Triangular (5,10,15)
D	Exponencial (5)

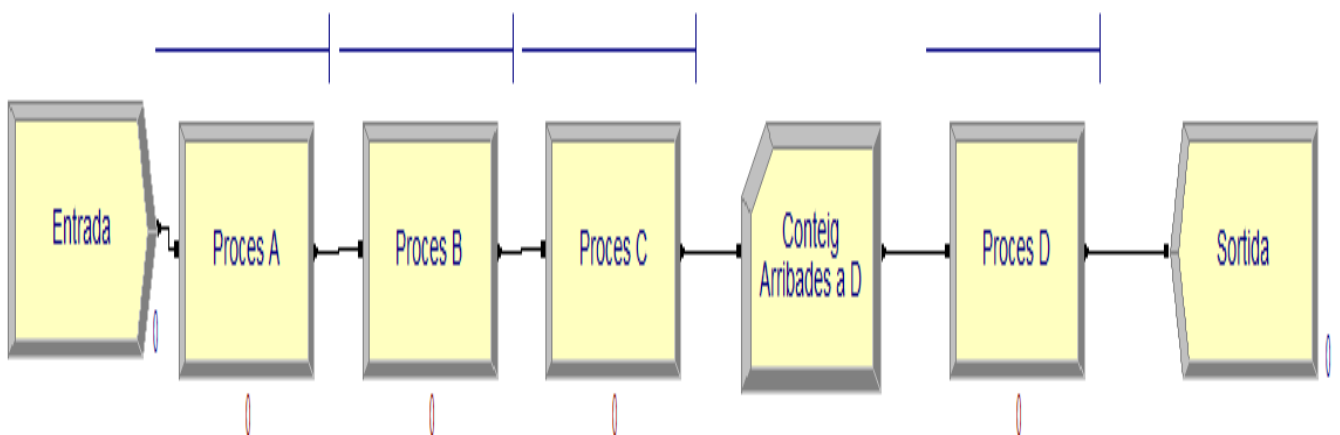


El temps entre arribades de les entitats al sistema es referència segueix una distribució uniforme entre 1 i 20 minuts. El sistema serà simulat en 2.400 minuts i les unitats de temps seran minuts.

A continuació es mostra la seqüència de mòduls en ARENA i en els següents subapartats veure'm com fer a través d'ells la recollida d'estadístiques d'usuari més comuns, les quals es realitzen mitjançant el mòdul RECORD.

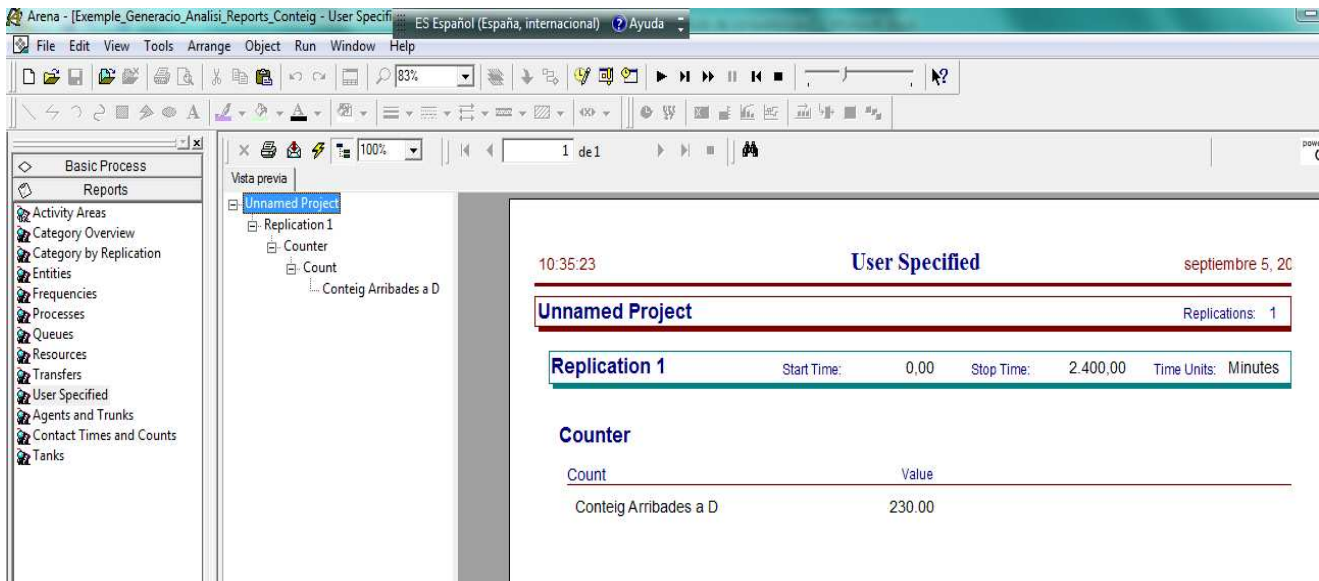


El model base es troba guardat dins del fitxer anomenat Exemple\_Generacio\_Analisi\_Reports.doe. Suposem que es desitja saber el nombre d'unitats que varen arribar al procés D. A tal efecte s'afegeix un mòdul RECORD abans del procés D i el mateix es configura com a tipus Count amb un increment de comptador de 1 en 1, tal i com es mostra a la figura de la dreta. A continuació s'exposa el model



modificat.

Aquesta variació del model base es troba guardada dins del fitxer anomenat **Exemple\_Generacio\_Analisi\_Reports\_Conteig.doe**



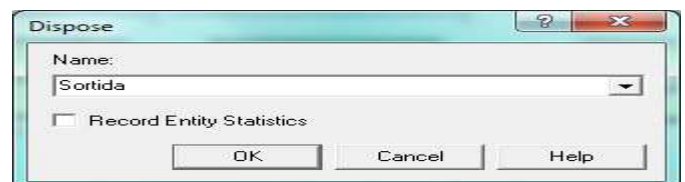
El model s'executa en el temps especificat i a continuació s'exposen els resultats:

Buscant el report sota la categoria User Specified es mostra la pàgina del resultat del conteig del mòdul RECORD, el qual dóna que entren 230 entitats dins del Procés D que venen del Procés C.

#### 4.6.2.2. Recol·lecció d'estadístiques de l'entitat.

Durant la simulació es van incrementant automàticament els diferents comptadors pels mòduls Create, Decide i Dispose, cada vegada que una entitat surt del mòdul. En canvi en el cas dels mòduls Process, el comptador mostra el número d'entitats que hi ha en cada moment en el mòdul, tenint en compte les entitats que s'estan esperant a la cua per utilitzar el recurs i les entitats que estan essent processades.

Ara bé suposem que volem recol·lectar estadístiques de l'entitat sense tenir en compte un mòdul Process, com per exemple el procés D. Per tal cosa s'afegeix un mòdul RECORD abans del mòdul de Procés D i es configura per a recol·lectar estadístiques de l'entitat (Entity Statistics). Tal i com es mostra a



continuació:

També i amb la finalitat d'evitar confusions, es desactiva la casella Record Entity Statistics del mòdul DISPOSE anomenat SORTIDA.

Aquesta variació del model base es troba guardada dins del fitxer anomenat **Exemple\_Generacio\_Analisi\_Reports\_Entitat.doe**

El model s'executa amb el temps especificat i a continuació s'exposen els resultats de la recollida d'aquest exemple, el qual a efectes pràctics significa que el Procés D no existeix, doncs les estadístiques de l'entitat es recullen abans que aquestes entrin dins d'aquest procés, i l'original on les estadístiques de les entitats es recullen al final dins del mòdul DISPOSE anomenat SORTIDA.

10:52:00

**Entities**

septiembre 5, 20

**Unnamed Project**

Replications: 1

**Replication 1**

Start Time: 0.00 Stop Time: 2.400,00 Time Units: Minutes

**Entity Detail Summary****Time**

	NVA Time	Other Time	Total Time	Transfer Time	VA Time
Part	0.00	0.00	52.81	0.00	23.92
<b>Total</b>	<b>0.00</b>	<b>0.00</b>	<b>52.81</b>	<b>0.00</b>	<b>23.92</b>

**Other**

	Number In	Number Out
Part	232	230
<b>Total</b>	<b>232</b>	<b>230</b>

Es pot veure que en l'exemple actual les entitats tenen un temps menor d'atenció (VA Time) sense incloure el Procés D, passen de 28,61 en el model base a 23,92 minuts.

En la següent figura del procés anterior es pot veure que el temps total (Total Time) també ha disminuït, de 58,78 a 52,81 minuts vist abans, doncs les estadístiques totals de les entitats es recullen abans de la sortida.

10:55:38

**Entities**

septiembre 5, 20

**Unnamed Project**

Replications: 1

**Replication 1**

Start Time: 0.00 Stop Time: 2.400,00 Time Units: Minutes

**Entity Detail Summary****Time**

	NVA Time	Other Time	Total Time	Transfer Time	VA Time
Part	0.00	0.00	58.78	0.00	28.61
<b>Total</b>	<b>0.00</b>	<b>0.00</b>	<b>58.78</b>	<b>0.00</b>	<b>28.61</b>

**Other**

	Number In	Number Out
Part	232	230
<b>Total</b>	<b>232</b>	<b>230</b>

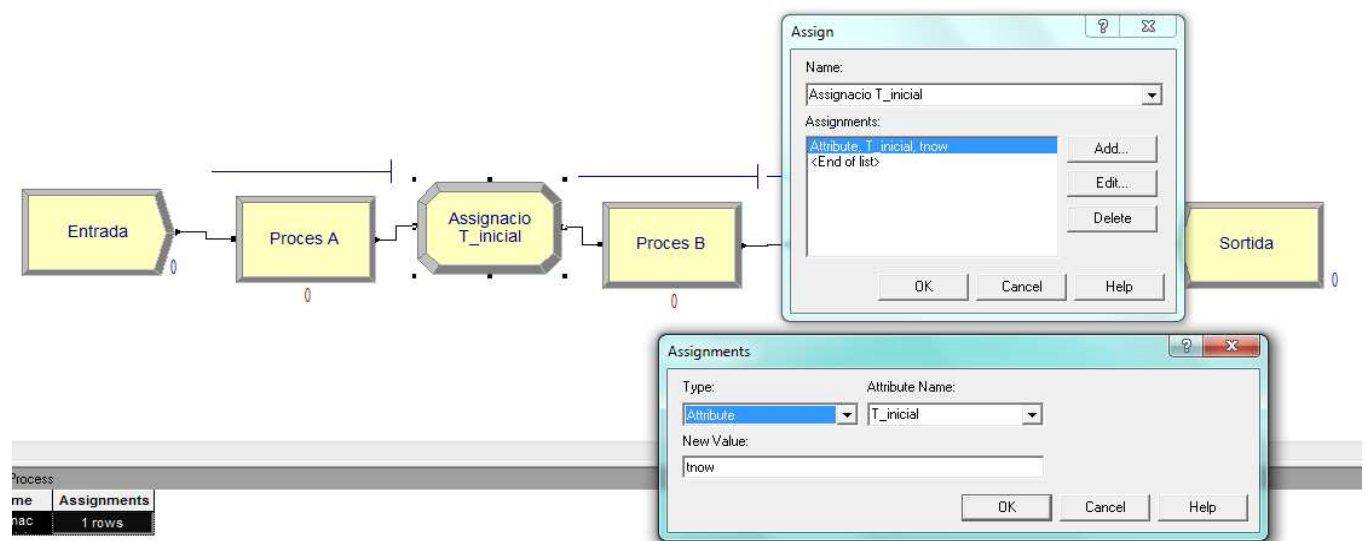
### 4.6.2.3. Estadístiques de l'interval de temps entre dos punts del model.

Suposem que ens interessa mesurar el temps que triga una entitat en anar entre la sortida del Procés A fins a la sortida del Procés C. Per això hem de seguir el següent procediment:

- Es pren el temps (temps simulat) amb el que una peça surt del Procés A,  $T_{inicial}$ .
- Es pren el temps (temps simulat) amb el que la mateixa peça surt del Procés B,  $T_{final}$ .
- La longitud de l'interval ve donada per la diferència  $T_{final} - T_{inicial}$  per a cada entitat.
- La longitud de l'interval s'acumula i es divideix entre el total d'entitats observades, posteriorment s'obté el promig del temps entre la sortida A i la sortida C (estadística observable o Tally)

Per a realitzar aquest procediment, afegirem un mòdul ASSIGN entre els processos A i B amb el qual s'ha de crear un atribut associat a l'entitat. Dins del qual s'emmagatzemarà el temps en el qual l'entitat passa pel primer punt de sortida del Procés A. En aquest cas en particular, aquest atribut rep en aquest instant el nom de  $t_{inicial}$  i el valor del temps de simulació (Tnow).

A continuació es mostra aquesta operació:



Aquesta variació del model base es troba guardada dins del fitxer anomenat Exemple\_Generacio\_Analisi\_Reports\_Punts.doe

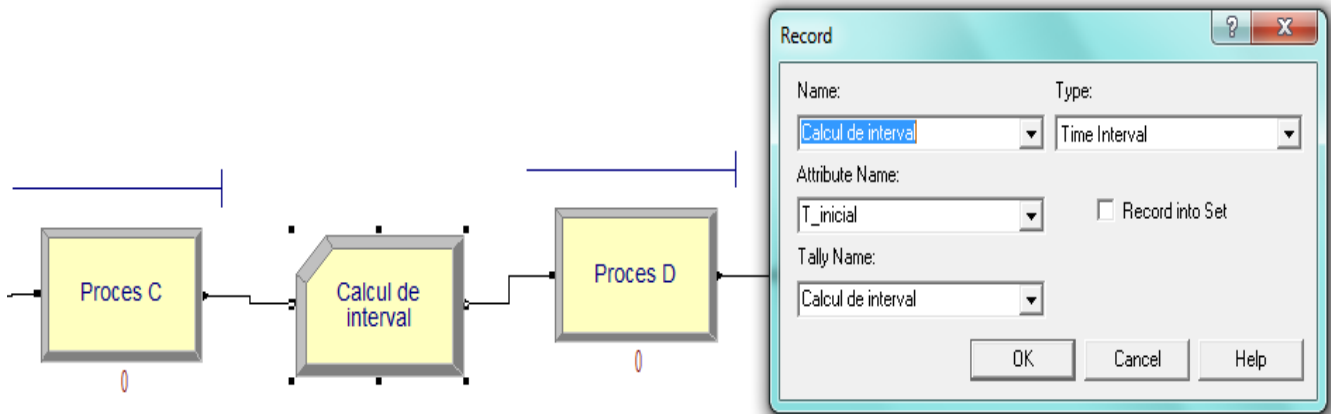
El valor de la variable (tnow) es pot obtenir si es fa clic amb el botó dret dins del camp New Value i s'entra dins del Expression Builder i es busca la variable mitjançant la següent ruta:

**Simulation control variables -> Current simulation time**

Per tal d'avaluar l' interval també s'afegeix un mòdul RECORD entre el Procés C i el Procés D, configurat com a tipus Time Interval. Aquesta acció permet avaluar un interval de temps tenint en compte el temps actual i un temps anterior prèviament guardat dins l'atribut T\_inicial.

També es crea una estadística observable o Tally, en la qual s'emmagatzenaran els intervals de temps de les diferents entitats, amb la finalitat que facin promig i es mostrin els reports.

L'edició del mòdul RECORD és la següent:



El model s'executa amb el temps especificat i a continuació es mostren els resultats (User defined):

11:33:02	<b>User Specified</b>		septiembre 5, 20
<b>Unnamed Project</b>			Replications: 1
<b>Replication 1</b>	Start Time:	0,00	Stop Time: 2.400,00 Time Units: Minutes

**Tally**

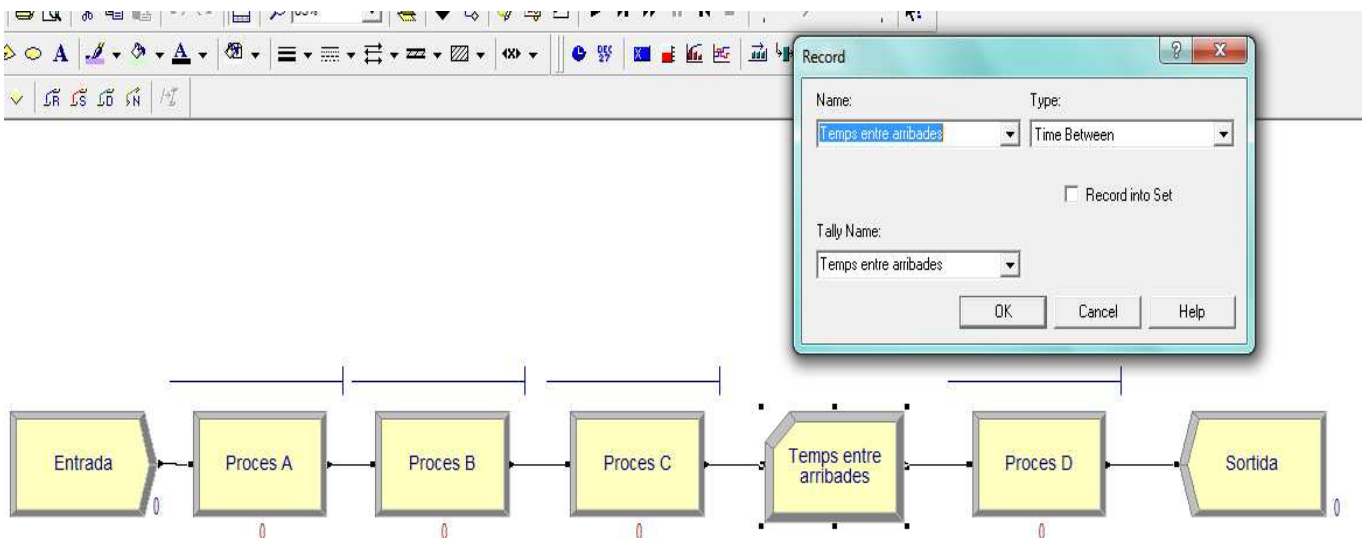
Interval	Average	Half Width	Minimum	Maximum
Calcul de interval	39.3214	(Insufficient)	11.6565	91.4576

Es pot veure que el temps promig que triga una entitat en anar de la sortida del Procés A la sortida del Procés C és de 39,3214 minuts i que el tamany de la mostra recollida amb respecte a la variabilitat de les dades, no és suficient per a calcular un interval de confiança ( Half With = Insufficient), , doncs van des d'un mínim de 11,6565 fins a un màxim de 91,4576 minuts.

#### 4.6.2.4. Estadístiques del temps entre arribades.

Per últim suposarem que es desitja conèixer la taxa d'arribades d'entitats al Procés D. Aleshores, s'haurà d'afegir un mòdul RECORD abans del Procés D i configurar la seva acció per a mesurar el temps entre arribades (Time Between).

Les n-1 dades (temps entre arribades de n entitats) s'emmagatzema dins d'una estadística del tipus Tally per al seu processament i report. L'edició del nou mòdul RECORD es mostra a continuació:



Aquesta variació del model base es troba guardada dins del fitxer anomenat Exemple\_Generacio\_Analisi\_Reports\_Arribades.doe

El model s'executa en el temps especificat i a continuació es mostren els resultats sota la categoria User defined en els reports Category overview (amb una sola rèplica s'obtenen els mateixos resultats):

11:48:12

User Specified

septiembre 5, 20

Unnamed Project

Replications: 1

Replication 1

Start Time:

0,00

Stop Time:

2.400,00

Time Units:

Minutes

#### Tally

Between	Average	Half Width	Minimum	Maximum
Temps entre arribades	10.3494	(Insufficient)	5.3935	25.2266

El promig de temps entre arribades d'entitats al Procés D és de 10.3494 minuts i el tamany de la mostra recollida amb respecte a la variabilitat de les dades, no és suficient per a calcular un interval de confiança (Half Width = Insufficient), doncs van des d'un mínim de 5,3935 fins a un màxim de 25,2266 minuts.

Per obtenir un interval de confiança s'hauria d'obtenir un tamany de mostra més gran mitjançant l'augment del temps d'execució, per exemple pujar-lo de 2400 a 10000 minuts de simulació.

Es mostren els resultats d'aquesta nova simulació:

11:56:33

**User Specified**

septiembre 5, 20

**Unnamed Project**

Replications: 1

**Replication 1**

Start Time:

0,00

Stop Time:

10.000,00

Time Units:

Minutes

**Tally**

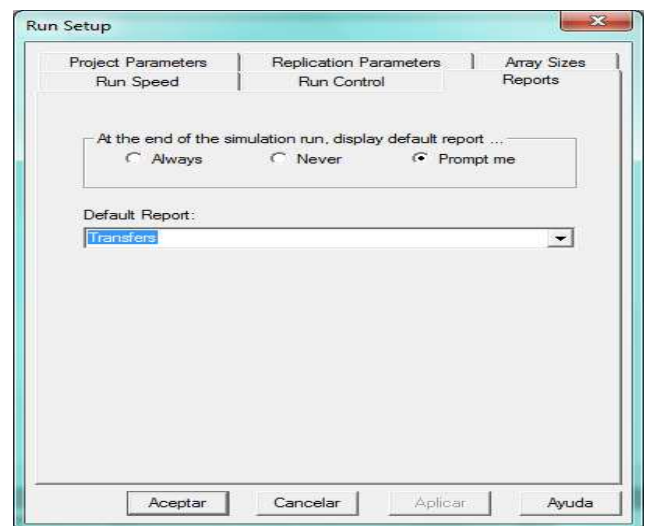
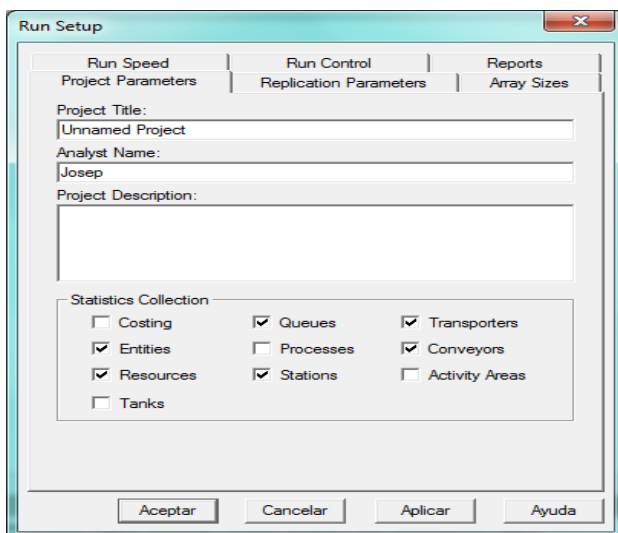
Between	Group #3 Name (Cadena)	Average	Half Width	Minimum	Maximum
	Temps entre arribades	10.6240	0,241519254	5.3531	26.1391

Els nous resultats, tot i que mostren que el promig entre arribades no ha variat significativament, ha passat de 10,4502 a 10,6240 minuts, ja permet obtenir un interval de confiança de 0,241519254 que correspon a la meitat d'un interval de confiança amb  $\alpha = 0,05$ . El que significa que la mitja de temps entre arribades es certa en un  $100 - 0,241519254 = 99,758\%$ .

### 4.6.3. ESTADÍSTIQUES DINS ELS MODELS DE TRANSPORT.

Estadísticament es diu que un model que corre sota la versió estudiantil de l'entorn ARENA es recomana que corri en no més de 1700 minuts, aquest temps ja es suficient per extreure estadístiques dels models més complexes, els que contenen transports.

Per tal que les estadístiques dels transports es mostrin en els reports, s'han d'activar les caselles STATIONS, TRANSPORTERS i CONVEYORS dins de la opció de menú: **Run -> Setup -> Project Parameters**, així com la casella de verificació corresponent a la recollida d'estadístiques de transport en el mòdul que ho requereixin i seleccionar com a report per defecte l'anomenat TRANSFER mitjançant la opció també de menú: **Run -> Setup -> Reports**.



#### 4.6.3.1. EXEMPLE DE MODEL DE TRANSPORT AMB ESTADÍSTIQUES.

A continuació s'exposa un model combinat de transport per tal d'estudiar els reports que d'ell es generen.

##### 1. DESCRIPCIÓ DEL MODEL DE TRANSPORT.

Suposarem un procés de manufactura mitjançant el qual es fabrica dos determinats tipus de peces per a l'industria de l'automòbil.

Les peces tipus A i tipus B arriben ambdues amb una distribució exponencial, però de mitjana 5 minuts i 10 minuts respectivament.

Les peces A i B son processades en un torn amb un temps uniforme entre 3 i 6 minuts i un cop acabades son emmagatzemades formant grups de 20, dins dels quals es poden barrejar peces tipus A i tipus B.



Posteriorment passen a un forn, dins d'aquest es realitza la cocció i les peces es mouen a una velocitat de 3 metres per minut amb un temps total contant de 20 minuts.

Una vegada fora del forn, les peces tipus A son polides una a una en un temps constant de 2 minuts per element i emmagatzemades en caixes de 10 unitats. Les peces tipus B no son polides i s'emmagatzemen en caixes de 50 unitats.

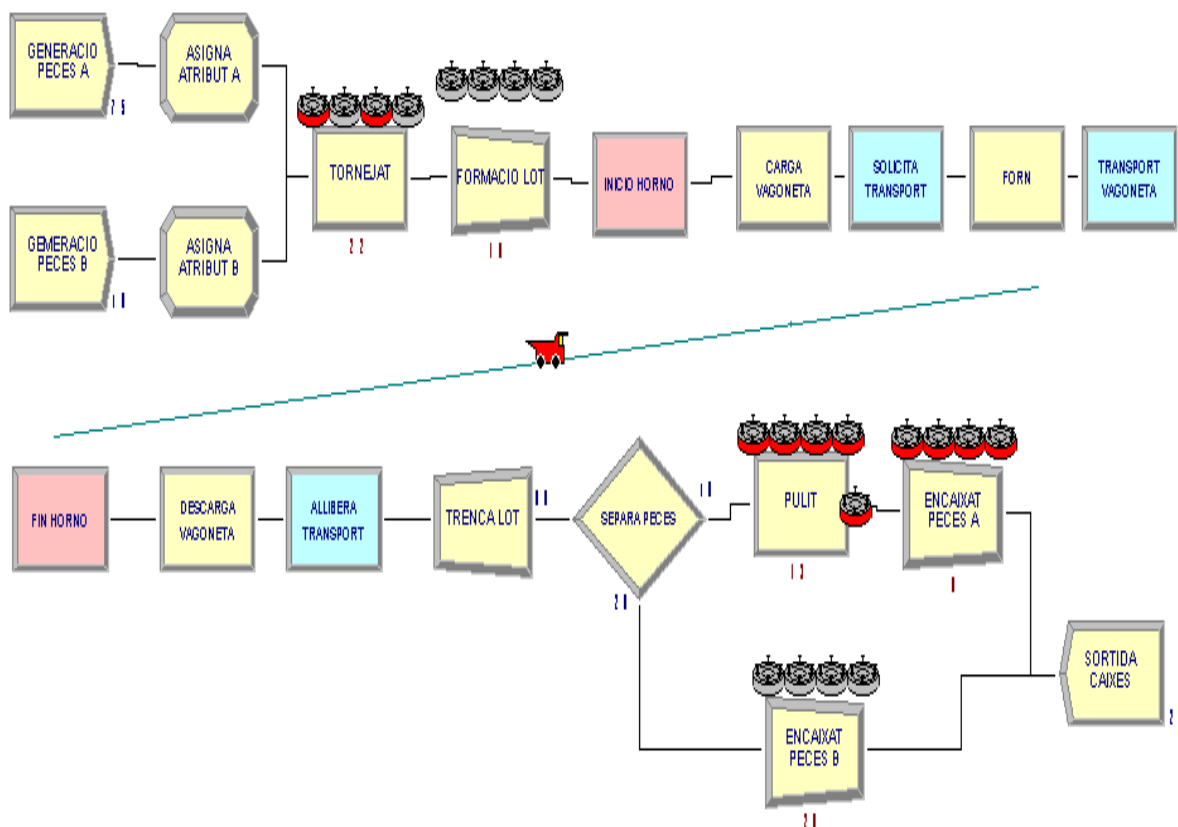
## 2. DESENVOLUPAMENT DEL MODEL.

Inicialment es generen les entrades d'entitats mitjançant dos mòduls DECIDE, un per les tipus A i una altra per a les tipus B.

A continuació per a diferenciar les entitats dins del model, se'ls hi afegixen valors diferents a l'atribut TIPUS, mitjançant un mòdul ASSIGN, 1 per les tipus A i 2 per les tipus B.

L'operació del torn es desenvolupa mitjançant un mòdul PROCESS anomenat TORNEJAT i a la sortida d'aquest trobem un mòdul BATCH, amb el propòsit de formar els lots de transferència de 20 unitats cap al forn, les quals entraran juntes dins d'una mateixa vagoneta.

En la figura següent podem veure el model generat per il·lustrar aquest exemple:



Aquest model es troba guardat dins del fitxer anomenat Exemple\_Reports\_Transport.doe

El forn es representa mitjançant un mòdul DELAY, doncs el procés de cocció es realitza en continu sense aturar els blocs d'entitats i el mateix dura 20 minuts.

A la sortida del forn es desmunten els lots de peces mitjançant el mòdul SEPARATE. Amb la qual cosa procedim a realitzar la descarrega de les peces de la vagoneta.

Per tal de realitzar l'operació de polit només a les peces tipus A, abans les haurem de seleccionar mitjançant un mòdul DECIDE i s'enviaran a un mòdul PROCESS per a realitzar la tasca de polit i s'agruparan en blocs de 10 mitjançant un bloc BATCH final.

Les peces tipus B, també són seleccionades a la sortida del forn mitjançant el mateix mòdul DECIDE que selecciona les tipus A i s'enviaran sense polir a un mòdul BATCH final per a ser agrupades en blocs de 50.

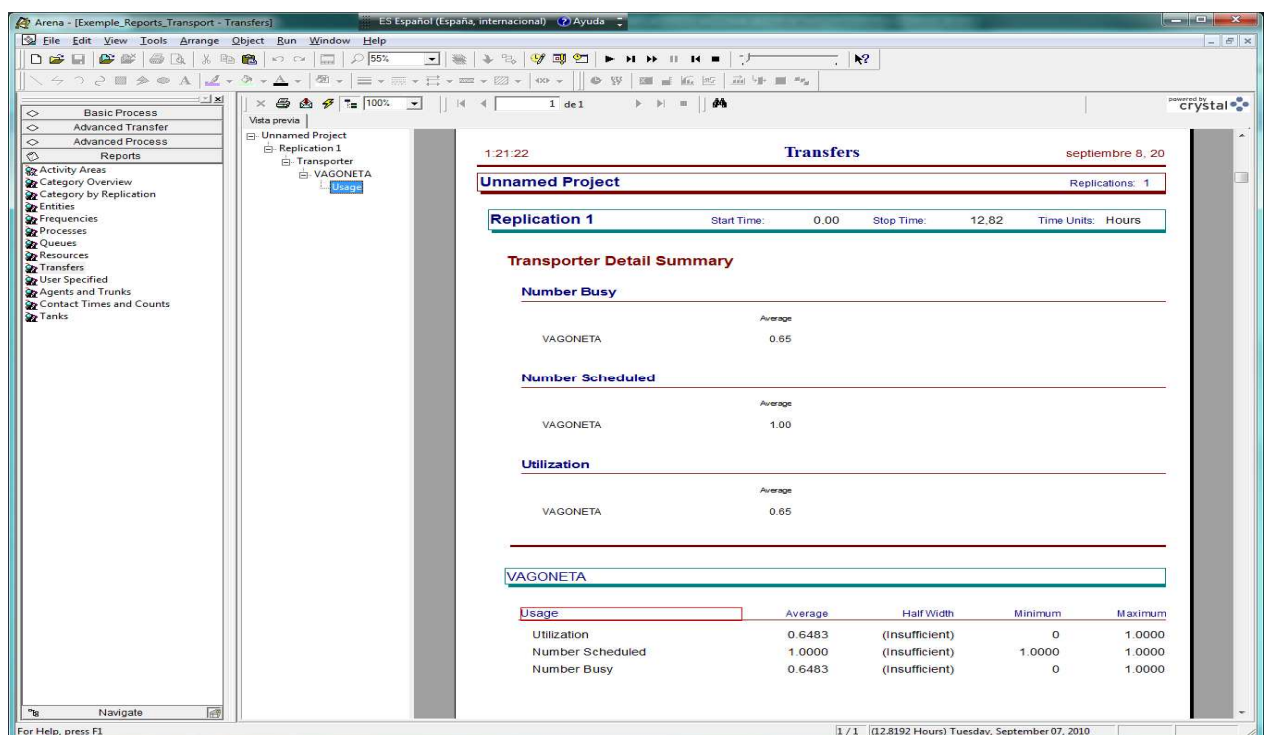
Una vegada empaquetades, les entitats llestes per a deixar el sistema, ho faran mitjançant un mòdul DISPOSE.

### 3. EXECUCIÓ DE LA SIMULACIÓ.

El màxim d'entitats que es generaran si no volem que la versió estudiant del sistema ARENA ens doni error han de ser 75 tipus A i 75 tipus B, les quals definirem dins dels mòduls CREATE.

### 4. ANÀLISI DE RESULTATS.

Una vegada hem executat el model li diem al sistema que volem veure les estadístiques finals i mitjançant la barra de navegació de l'esquerra obrirem la opció Transfers.



En l'informe es pot veure el número promig d'elements de transports que s'utilitzen i en aquest cas és només 1 (Number Scheduled), el qual està sent utilitzat un 64,83 % (Utilitzation), el que vol dir que les entitats no es demoren pel transport. A continuació es mostra el temps total de les peces dins del sistema:

Total Time	Average	Half Width	Minimum Value	Maximum Value
PECES_A	4.8709	(Insufficient)	3.1049	6.2178
PECES_B	11.8948	(Insufficient)	11.8948	11.8948
<b>Other</b>				
Number In	Value			
PECES_A	87.0000			
PECES_B	76.0000			

Si ara fem que els lots de transferència cap al forn, en comptes de 20 unitats només en carreguin 10. Veurem que l'únic transport que tenim es troba saturat, doncs la seva utilització és del 93,3 %, tal i com es mostra a la figura següent:

VAGONETA				
Usage	Average	Half Width	Minimum	Maximum
Number Busy	0.9333	(Insufficient)	0	1.0000
Utilization	0.9333	(Insufficient)	0	1.0000
Number Scheduled	1.0000	(Insufficient)	1.0000	1.0000

Aleshores el temps total de les peces al sistema ha augmentat. Inicialment era de 4,8709 minuts per les peces tipus A i de 11,8948 minuts per les tipus B i ara amb el transport saturat és de 6,7745 minuts per les peces tipus A i de 18,6173 minuts per les tipus B.

Total Time	Average	Half Width	Minimum Value	Maximum Value
PECES_A	6.7745	(Insufficient)	2.5870	10.5146
PECES_B	18.6173	(Insufficient)	18.6173	18.6173
<b>Other</b>				
Number In	Value			
PECES_A	96.0000			
PECES_B	76.0000			

Aleshores mitjançant les estadístiques dels transport podem respondre fàcilment a la següent pregunta:

Quants elements de transport haurem de posar com a mínim per transportar les peces en grups de 10 unitats dins del forn, sense perdre temps total ?

El que fem és augmentar la quantitat d'elements de transport, per exemple a 10, mitjançant el mòdul de dades TRANSPORTER (posant el camp Number of Units a 10) tal i com es mostra a la figura següent.

Transporter - Advanced Transfer								
	Name	Number of Units	Type	Distance Set	Velocity	Units	Initial Position Status	Report Statistics
1	VAGONETA	10	Free Path	VAGONETA.Distance	1.0	Per Hour	0 rows	<input checked="" type="checkbox"/>

Double-click here to add a new row.

Aleshores si executem de nou el model, veurem que si consultem les estadístiques Transfers el sistema ARENA ens ha ajustat el transports a dos com a màxim, tot i que el segon està molt insaturat.

VAGONETA				
Usage	Average	Half Width	Minimum	Maximum
Number Busy	1.1800	(Insufficient)	0	2.0000
Utilization	0.1180	(Insufficient)	0	0.2000
Number Scheduled	10.0000	(Insufficient)	10.0000	10.0000

Ara tenim que el nombre de transports ocupats és 1,18 i que la utilització dels 10 transports és únicament del 11,8 %.

Encara que es deixin només dos transports, la decisió de si aquest model és correcte, l'ha de prendre l'analista, mirant els temps totals de les entitats al sistema amb aquest increment del transport.

Total Time	Average	Half Width	Minimum Value	Maximum Value
PECES_A	3.3949	(Insufficient)	2.5870	3.9934
PECES_B	11.0560	(Insufficient)	11.0560	11.0560
<b>Other</b>				
<b>Number In</b>	<b>Value</b>			
PECES_A	96.0000			
PECES_B	76.0000			

Pel que s'observa a la figura, evidentment amb més transport realitzant la transferència, el temps total ha baixat de 6,7745 a 3,3949 minuts per a les peces tipus A i de 18,6173 a 11,0560 minuts per les tipus B.

Només quedaria doncs per avaluar, si el cost d'un element de transport més (augment del 50%) compensa la reducció observada del 50 % de temps per les peces A i el 40 % de temps per a les peces tipus B, en el seu temps total de permanència en el sistema.

#### 4.6.4. EMMAGATZENAMENT D'ESTADÍSTIQUES DINS D'UN ARXIU EXTERN.

Moltes vegades es fa necessari recollir estadístiques sobre una determinada variable amb la finalitat de realitzar proves estadístiques en d'altre programes externs al sistema ARENA.

Aleshores es requereix emmagatzemar estadístiques dins d'un fitxer extern per tal que posteriorment es puguin processar. Amb ARENA és possible tant llegir arxius com escriure o generar-los mitjançant el mòdul READ-WRITE o bé directament amb el mòdul de dades avançat STATISTIC.

##### 4.6.4.1. Emmagatzemament mitjançant el mòdul ReadWrite.

El programa ofereix la possibilitat de generar arxius, però té les seves limitacions pel que respecta a la quantitat d'observacions que suporta.

En general, dins d'un arxiu es poden escriure variable o atributs i el nombre d'observacions dins del mateix correspon al nombre d'entitats que passen pel mòdul **ReadWrite**.

Per tal de seguir aquesta tasca, es planteja la realització d'un exercici pràctic, el qual s'exposa a continuació.

##### 1. DESCRIPCIÓ DEL MODEL DE TRANSPORT.

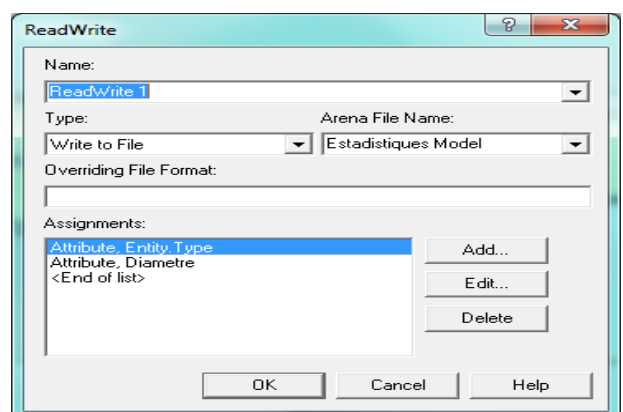
Modificarem el model de l'apartat 4.6.3.1. Afegirem un atribut que ens definirà el diàmetre de les peces Tipus A i Tipus B anomenat Diametre, el qual seguirà una distribució Normal amb una mitjana 10 per les primers i 20 per les segones amb una desviació estàndard de 0,2.

Abans que les entitats abandonin el sistema guardarem el valor del diàmetre de cadascuna de les peces dins d'un arxiu extern.

##### 2. DESENVOLUPAMENT DEL MODEL.

Primer afegirem un element Assignments dins de cada mòdul ASSIGN, un per les peces tipus A, el qual assignarà valors seguint una distribució NORM (10,0.2). i una altra per les tipus B, amb una distribució NORM (20,0.2).

A continuació adicionarem un mòdul lògic **ReadWrite**, el qual es troba incorporat dins



del *template Advanced Process* abans de la sortida de l'últim mòdul **DISPOSE**, anomenat Sortida Caixes.

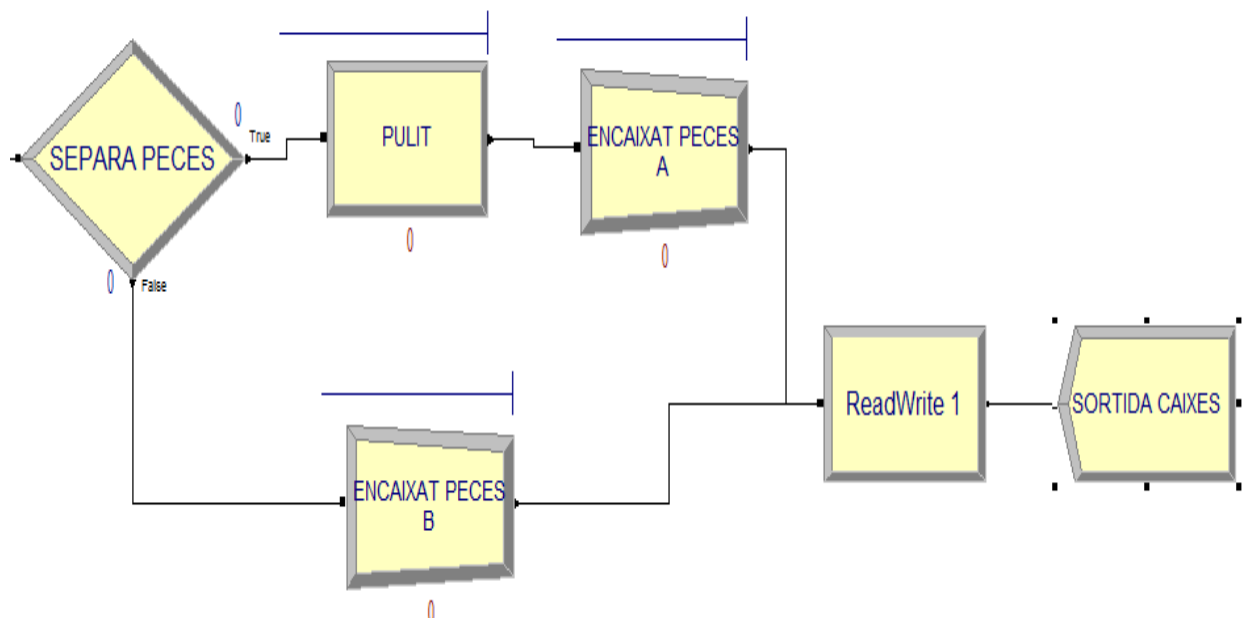
Dins del mòdul **ReadWrite** es farà referència al tipus d'acció que realitzarem sobre el fitxer, al seu nom sense extensió i assignarem l'acció a realitzar, la qual guardarà el nom de l'atribut (Entity.Type) i el seu diàmetre. En aquest cas escriurem sobre el fitxer anomenat Estadistiques Model l'atribut Diametre.

Cal dir que l'arxiu amb el qual es treballarà, primer ha d'estar identificat dins del camp *Operating System File Name* del mòdul de dades **FILE**, el qual es troba també dins del *template Advanced Process*. L'extensió del fitxer de text on es guardaran les dades serà .dat i el camp Estructura s'aconsella deixar-lo amb l'opció per defecte (Free Format) i en la següent figura es mostra la seva configuració:

File - Advanced Process							
	Name	Access Type	Operating System File Name	Structure	End of File Action	Initialize Option	Comment Character
1	Estadistiques Model	Sequential File	Estadistique Model.dat	Free Format	Dispose	Hold	No

Double-click here to add a new row.

Així doncs el model afegint el mòdul d'escriptura quedarà de la forma següent:



El model es guardarà amb el nom Exemple\_Reports\_Transport\_FILE.doe

#### 4.6.4.2. Emmagatzament mitjançant el mòdul STATISTIC.

El mòdul avançat de dades **STATISTICS**, el qual es troba incorporat dins del *template Advanced Process* s'utilitza per definir noves estadístiques que es recullen durant la simulació, així com per especificar les dades d'arxius de sortida.

Encara que les estadístiques de resum (per exemple, mitjana i màxima) es generen automàticament per a cada estadística, si s'especifica un arxiu de sortida es pot realitzar l'anàlisi de resultats de la simulació mitjançant una aplicació externa, com ara la comparació dels resultats de dues configuracions de sistema per avaluar si hi ha una diferència significativa en el funcionament.

El mòdul **STATISTIC** pot guardar cada valor individual observat d'una estadística model a un arxiu de sortida de dades amb qualsevol nom de fitxer vàlid per aquests, però mitjançant una opció de configuració de menú: **Eines Tools -> Options -> Run Control -> Write Statistics Output Files as Text**, es determina si els arxius de dades de sortida han de ser escrits en text o binari. Si l'opció de text és seleccionada, es generen arxius separats per comes (\*.csv) que es poden llegir en els editors de text, Excel, i molts altres programes populars. Si l'opció no es selecciona text, els arxius es generen en forma binària (\*.dat) per al seu ús amb l'analitzador de dades de sortida del **Output Analyzer** que incorpora el paquet ARENA.

### Tipus d'estadístiques.

Les dades de sortida contenen un registre dins del fitxer per a cada valor observat (per a les estadístiques del tipus observació i comptador) o per a cada canvi d'un valor (per a les estadístiques de tipus temps, persistent i estadístiques de les freqüències). Els registres consisteixen en un parell de valors: el valor i el moment en què es va produir.

Els tipus d'estadístiques es seleccionen mitjançant el camp **Type** del mòdul **STATISTIC** i les característiques principals per a cadascuna de les seves opcions son les següents:

#### 1. Time-persistent (Persistents).

Statistic - Advanced Process					
	Name	Type	Expression	Report Label	Output File
1	Clients	Time-Persistent	NQ(botiga.Queue)	Clients	Clients_Cua_Botiga.dat

Double-click here to add a new row.

Aquest exemple també recull estadístiques sobre la longitud d'una cua. No obstant això, en lloc d'informar de la proporció de temps que la longitud de la cua es trobava en un conjunt de categories, l'estadística de temps persistents (*Time-Persistent*) proporciona la mostra ponderada

en el temps de la mitjana de la longitud de la cua **botiga.Queue**. Per exemple, si la cua estava buida durant una quarta part de la rèplica de simulació i contenia una sol'entitat per a les altres tres quartes parts de la rèplica, l'informe contindrà una mitjana de 0,75 i aquesta dada serà guardada dins del fitxer *Clients\_Cua\_Botiga.dat*.

## 2. Tally (Observable).

Statistic - Advanced Process				
	Name	Type	Tally Name	Tally Output File
1	Components vermells	Tally	Temps Components Vermells	Vermells.dat ...

Double-click here to add a new row.

En aquest exemple, la informació estadística addicional és necessària per a una dada observada concreta (**Tally**). La dada observada es guarda dins l'estadístic anomenat **Temps Components Vermells**, el qual descriu el temps de flux del sistema per a les entitats anomenades **Components Vermells**. Una informació resumida de l'estadístic es mostra a l'usuari en l'informe específic. No obstant això, en especificar un arxiu de sortida, la informació de persona física es registra en l'arxiu **Vermells.dat**.

Cada vegada que una entitat flueix a través del mòdul **STATISTIC**, el seu temps de flux pel sistema i el seu valor en temps de simulació (TNOW), en els qual es produeix l'observació (**Tally**), es registren en l'arxiu de dades. Aquest fitxer pot ser utilitzat amb l'analitzador de sortida **Output Analyzer** per a l'anàlisi estadístic.

## 3. Counter (Comptador).

Statistic - Advanced Process						
	Name	Type	Counter Name	Limit	Initialization Option	Counter Output File
1	Comandes	Counter	Comandes 2	100	Replicate	...

Double-click here to add a new row.

En aquest exemple, el comptador anomenat **Comandes 2** té especificat com a límit 100. Una vegada que el valor de **Comandes 2** arribi a 100, la simulació del model s'aturarà. El límit no indica el nombre d'entitats que passaran a través del comptador, però si el valor d'aquest.

Si deixem en blanc el camp **Counter Output File**, el valor del comptador persistirà només durant l'execució de la simulació però no es guardarà dins d'un fitxer.

El comptador es pot inicialitzar o no en funció del que s'indiqui dins l'opció **Initialization Option (Yes, No)**. Aquesta té una tercera opció, **Initialization Option**, si la escollim indica que el comptador s'esborra, o retorna el seu valor a zero, cada vegada que les estadístiques que les



estadístiques de simulació s'esborren entre repeticions o durant el període d'escalfament.

#### 4. Output (Sortida).

Statistic - Advanced Process					
	Name	Type	Expression	Report Label	Output File
1	Comandes	Output	NQ(botiga.Queue)	Comandes	Clients_Cua_Botiga.dat

Double-click here to add a new row.

Si escollim aquesta opció en aquest exemple es recullen les estadístiques sobre la longitud d'una cua cada cop que una entitat arriba al mòdul **STATISTIC** i es guarden dins del fitxer *Clients\_Cua\_Botida.dat*.

#### 5. Frequency (Basades en freqüències o intervals)

	Constant or Range	Value	High Value	Category Name	Category Option
1	Constant	0		Buit	include
2	Range	0	6	Mig ple	include
3	Range	6	15	Ple	include

Advanced Process							
Name	Type	Frequency Type	Expression	Resource Name	Report Label	Output File	Categories
Ascensor estat	Frequency	Value	NQ(Ascensor.Queue)		Ascensor estat		1 rows

Aquest exemple il·lustra l'ús d'una freqüència en funció del valor d'una expressió. L'expressió que s'avalua és el número a la cua Ascensor.Queue o NQ (Ascensor.Queue). L'etiqueta que s'imprimeix en l'informe és l'estat de l'Ascensor per a cascuna de les tres categories, buit (la cua té 0 entitats en ella), Mig ple (la cua te entre 1 i 6 entitats) i Ple (hi han més de 6, però menys de o exactament 15 entitats a la cua).

Sota la opció Category Option a part d'escollir **Include**, una o més categories poden ser excloses d'alguns càlculs de freqüències per tal de recollir estadístiques en un subconjunt restringit. Això s'indica mitjançant l'especificació **Exclude**.

L'informe resum exclusiu per aquesta opció, es pot consultar dins l'informe final si seleccionem el tipus de report per defecte (Default Report) **Frequencies** sota el menú desplegable de configuració de l'execució, el qual es crida mitjançant l'opció **Run ->Setup -> Reports** des de la pantalla principal. Dins del mateix report es mostraran les estadístiques separades per a cadascuna de les dues categories, estàndard i restringida, per a totes les freqüències especificades.

## 4.7. ALTRES EINES D'ANÀLISI I PROCESSOS AVANÇATS.

La simulació és una eina que permet l'anàlisi de sistemes mitjançant la captura de part de la complexitat i variabilitat inherents a la realitat per tal de constituir un suport robust en el moment de la presa de decisions.

Mitjançant un model de simulació es poden desenvolupar els següents processos:

1. **OPTIMITZACIÓ:** Permet determinar la configuració de certes variables que maximitzen o minimitzen una mesura del desenvolupament segons sigui el cas.
2. **PREDICCIÓ:** Permet estimar l'efecte tant de variacions en el sistema com en l'entorn.
3. **CONEIXEMENT:** Permet:
  - Confirmar relacions causa-efecte de certs events.
  - Confirmar que totes les variables rellevants són conegudes.
  - Avaluar la sensibilitat del model davant variacions en els paràmetres més importants.
4. **JUSTIFICACIÓ:**
  - Permet decidir entre dues alternatives en relació a una mesura de rendiment.
  - Un model de simulació amb una bona animació pot ser generador de noves idees.

Dins d'un sistema es troben factors controlables e incontrolables que produeixen algun efecte en una determinada variable de resposta. Sota aquest punt de vista, es pot fer servir la simulació per tal d'avaluar l'efecte que tindrien certs valors d'aquest factors o variables, tant controlables com incontrolables, sobre la o les variables de resposta que s'escullin. D'aquesta manera, cada conjunt de valors per als factors anomenats abans rep el nom d'escenari. Cal dir que dins de la realitat els factors incontrolables no es poden variar a criteri de l'analista, però, mitjançant la simulació es poden estimar els efectes que les variacions d'aquests tindrien sobre el sistema.

Començarem aquest apartat amb dues de les primers eines avançades d'anàlisi, l'anomenada **ARENA Process Analyzer** i l'**ARENA Output Analyzer**.

A continuació veurem i analitzarem com es comunica ARENA externament amb les principals eines de la suite Microsoft Office per al tractament de dades, Microsoft Excel i Microsoft Access. La primera té una interacció directa amb ARENA i dedicarem un apartat, mentre que a les dades que emmagatzema la segona ARENA hi accedeix mitjançant la programació de mòduls específics dins l'entorn **VBA** (Visual Basic For Applications).

L'últim sub apartat de la part teòrica tractarà doncs d'introduir la programació de mòduls **VBA**, mitjançant els quals podem connectar ARENA amb el món exterior sense cap tipus de limitació.

### 4.7.1. ARENA Process Analyzer.

ARENA compte amb un eina anomenada *Process Analyzer* (analitzador de processos), la qual permet a l'analista generar, avaluar i completar diversos escenaris d'un sistema, prèvia construcció, verificació i validació del model en relació a una o vàries variables de resposta específica. A continuació s'exposen dos exemples per tal que el lector interpreti quins son els casos en els quals seria convenient utilitzar aquesta eina, la qual treballa independentment del mòdul principal d'ARENA, com a programa extern.

- **EXEMPLE 1.**

En un sistema de cues d'un banc es pot establir l'efecte que tindria incloure un caixer més (factor controlable) dins del paràmetres d'atenció o variables de resposta (temps d'espera, temps en el sistema, nombre de clients atesos per hora, entre d'altres). Inclús, també es poden estimar l'efecte que tindria l'increment en la taxa d'arribada de clients (factor incontrolable) tenint en compte la capacitat actual.

- **EXEMPLE 2.**

En un sistema d'inventari es poden estimar l'efecte que tindria un canvi en les polítiques (R, Q o S) sobre els costos totals d'operació. També es poden determinar que tan robusta seria la política actual davant d'increments en les quantitats demanades (factor incontrolable).

En realitat existeixen molts més factors que controlar i situacions en diferents escenaris. El Process Analyzer es converteix en una eina bàsica, però no es aconsellable carregar grans models, és millor fer particions dels model i fer servir les conclusions d'aquest anàlisi parcialment.

Aleshores podrem crear simulacions robustes i eficients, evidentment buscant una relació servei-cost equilibrada, al igual que en la vida real.

Per tal d'analitzar l'ús d'aquesta eina s'estudia el cas d'un sistema de cues d'un banc al qual arriben clients amb temps entre arribades, el qual segueix una distribució normal de mitja 5,89 i desviació estàndard de 0,65. Mentre que els clients son atesos per un únic caixer en un temps que segueix una distribució exponencial de mitjana 4,5.

Després de ser atesos, els clients es retiren del sistema (banc) i la simulació total del sistema ha de ser de 480 minuts.

En aquest cas es tenen les següents possibles variables de resposta del sistema:

6. El temps que espera un client a la cua (Wait time).
7. El temps que passa un client en el sistema (Total time).
8. La longitud promig de la cua (Number in queue).
9. El número de clients en el sistema (WIP).

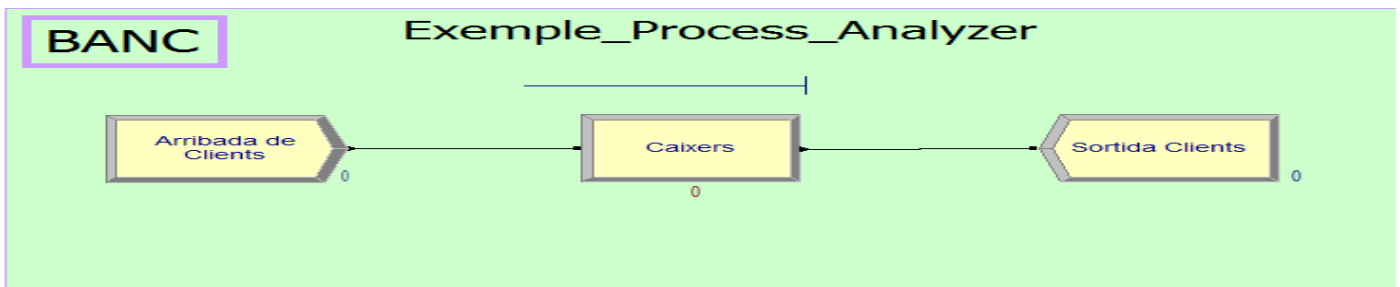
10. El número promig de caixers ocupats (Number busy).

11. La utilització dels caixers (Utilization).

La variable de control és el nombre de caixers. Aquesta variable es troba representada per la capacitat del recurs caixer, la qual es pot veure dins la fulla de dades del recurs (Resources) del panell de procés bàsic.

L'objectiu és el d'explorar diferents escenaris per tal d'observar l'efecte que té, en les variables de resposta, treballar amb 1,2,3 o 4 caixers.

A continuació es mostra el model en ARENA corresponent al sistema bancari sota anàlisi , el mateix



serà guardat sota el nom de fitxer següent, Exemple\_Process\_Analyzer.dot

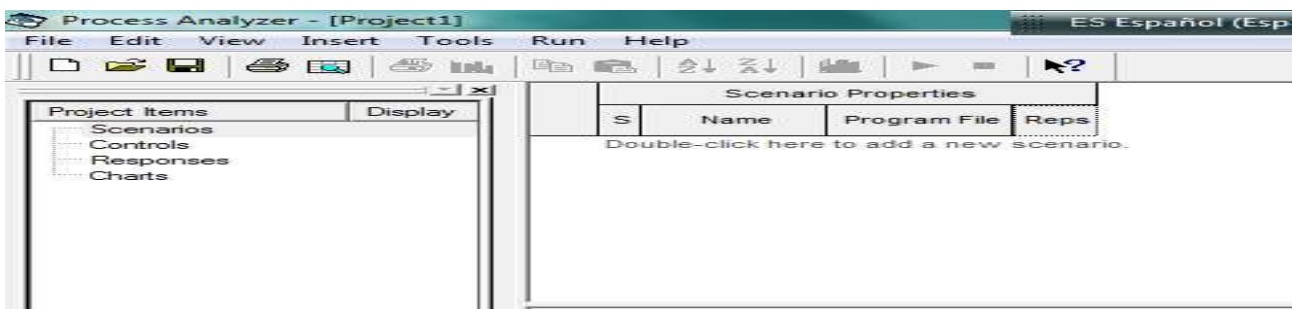
Per aconseguir un millor anàlisi del model es faran 5 rèpliques de 480 minuts cadascuna. Una vegada es construeixi i verifiqui el model (s'assumeix que aquest ha estat validat), entrarem dins del Process Analyzer mitjançant l'entrada de menú Tools -> Process Analyzer o mitjançant la ruta Rockwell/Software/Arena/Process Analyzer de del menú d'inici de Windows, doncs al programa pot córrer independent, sense la interfície ARENA.

Important: El model motiu d'anàlisi ha d'estar lliure d'errors i s'ha d'haver generat el fitxer .p corresponent,

mitjançant la tecla F4 apareixerà l'avís mostrat a la figura indicant el model e troba lliure d'errors, abans de guardar el model en disc.



Una vegada obert el Process Analyzer s'ha de crear un nou projecte mitjançant la ruta File -> New o a



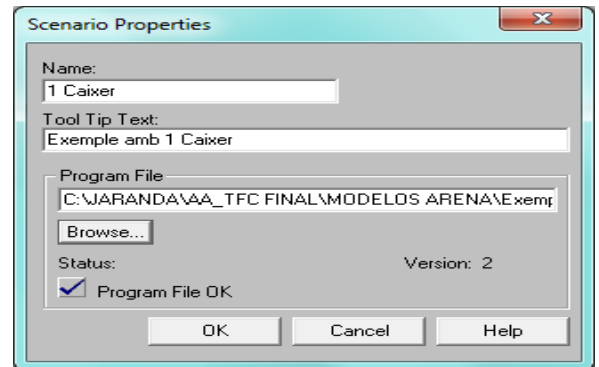
través de la icona de nou projecte.

A partir d'aquí, exposarem els nou passos que haurem de seguir en tot anàlisi de processos en arena.

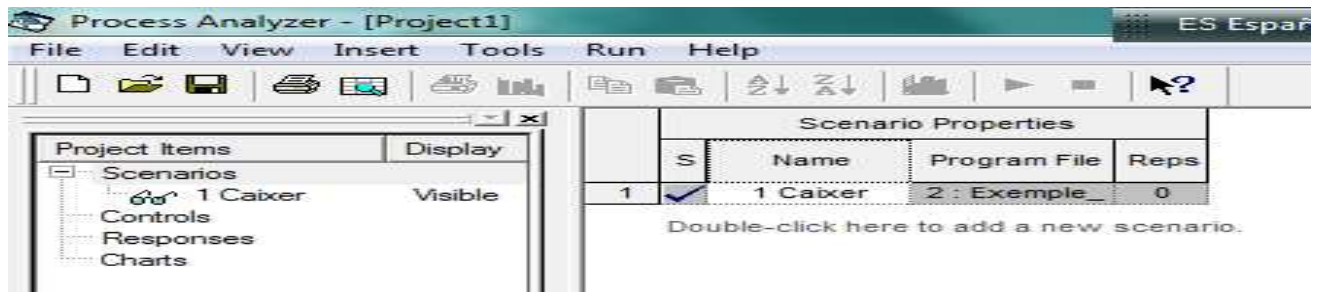
### 1. CREACIÓ D'UN ESCENARI.

Dins del nou projecte es troba la fulla de dades d'escenaris.

A continuació es mostra el quadre de diàleg amb les propietats de l'escenari que s'ha de crear. Omplirem el nom i una descripció de l'escenari i a continuació mitjançant el botó Browse buscarem dins del directori on tenim guardat el model, el fitxer Exemple\_Process\_Analyzer.p que ens servirà per a realitzar l'anàlisi.



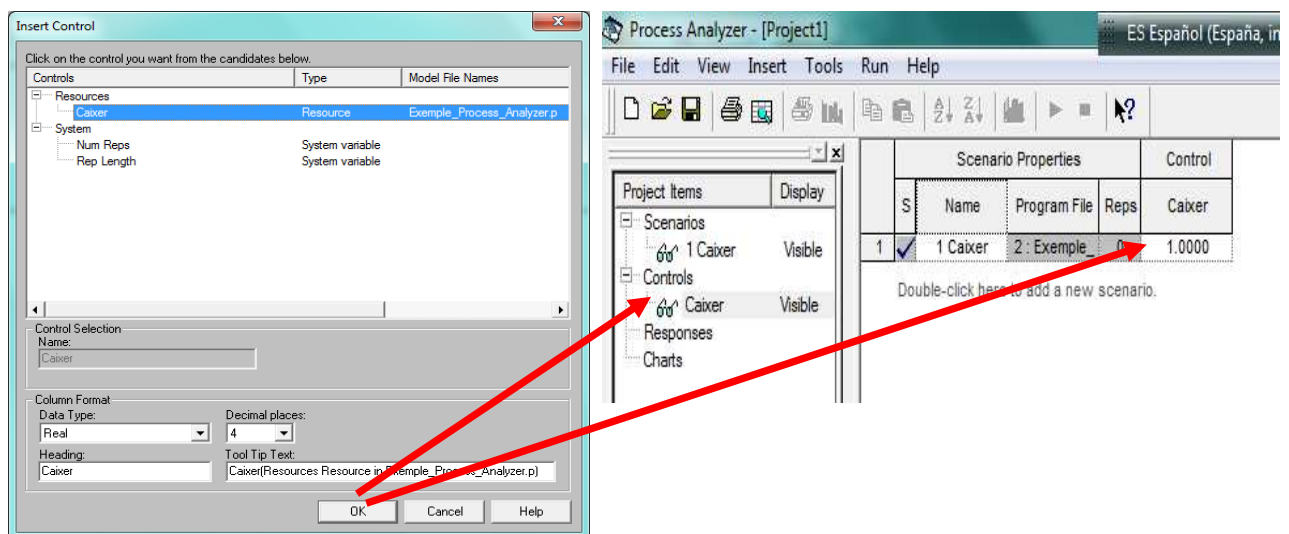
Una vegada es torna a la finestra del Process Analyzer ja es té creat el model per a l'escenari 1.



Si es volen crear els següents escenaris per a 2,3 i 4 caixers sobre el mateix model, es convenient crear primer un escenari especificant totes les variables de resposta i els actors de control, per passar després a duplicar-lo i d'aquesta manera, evitar repetir tots els passos enunciats a continuació.

### 2. DEFINICIÓ DE CONTROLS.

Mitjançant la opció de menú *Insert -> Control...* s'obre el següent formulari:



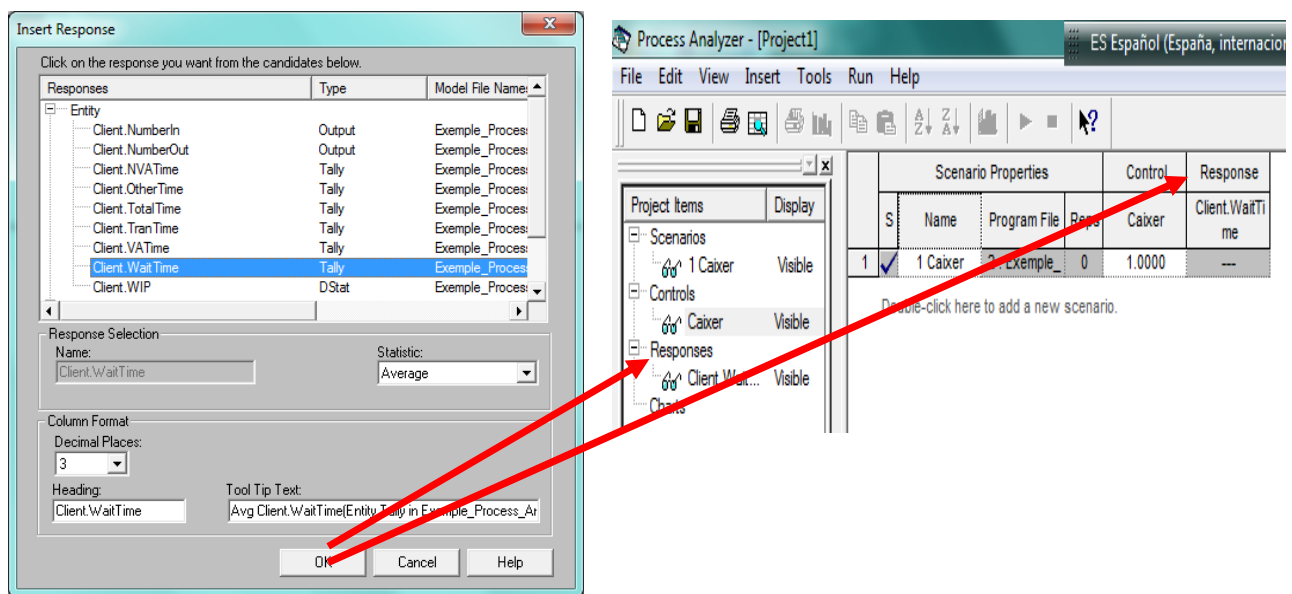
Mitjançant el mateix obrirem els Resources disponibles i seleccionarem el recurs Caixer.

Una vegada seleccionat tanquem mitjançant OK i apareixerà sota la columna *Project Items* el *Control* Caixer *Visible* i afegit a l'escenari.

### 3. DEFINICIÓ DE LES VARIABLES DE RESPOSTA.

En el model existeixen quatre tipus de categories, de les quals es poden seleccionar les variables de resposta, a saber: **entitats, cues, recursos i sistema**. El temps d'espera, els temps de sistema i el nombre de clients en el sistema es prenen de la categoria entitats.

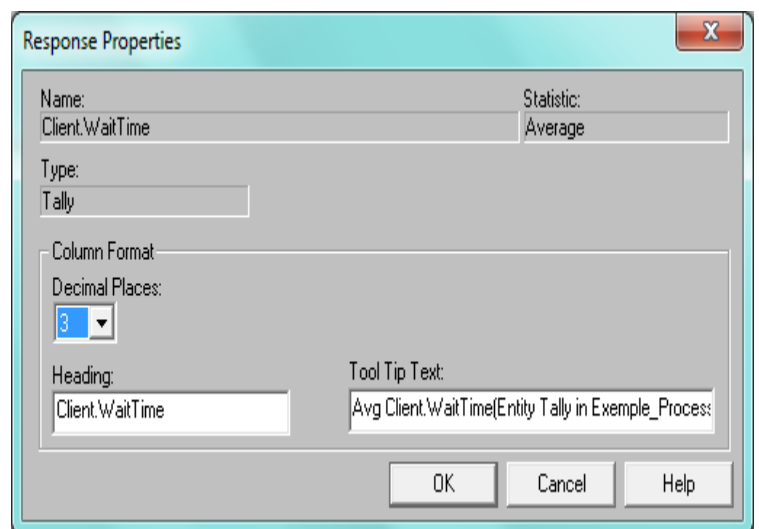
Mitjançant la opció de menú *Insert -> Response...* s'obre el següent formulari:



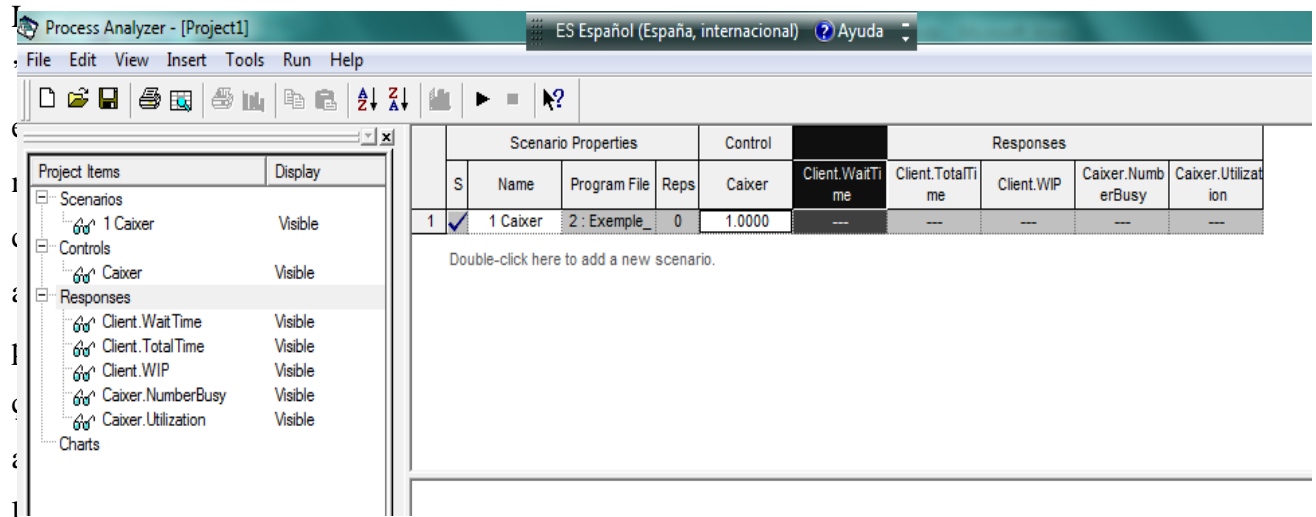
Segons el qual inclourem la primera resposta per a les entitats que es vol obtenir de l'anàlisi del procés, el temps d'espera de cada entitat anomenada client. Es busca aquest tribut, Client.WaitTime, de la llista que es desplega sota la variable de resposta Entity on Client és el nom que se li dona a l'atribut Entity.Type de l'entitat seleccionada.

Quan es pressiona OK es torna a la finestra del projecte, on es veu actualitzada la fulla de dades dels escenaris amb les dades de la variable de resposta seleccionada.

La variable de resposta seleccionada,



inicialment no te cap valor, donat que encara no s'ha executat cap simulació.



L'element es pot canviar dins del camp Heading del formulari Insert Response o bé fent clic al botó dret del ratolí sobre la columna Response a la finestra del projecte, amb la qual cosa s'obrirà un desplegable. On escollirem l'opció Response Properties... i apareixerà el següent formulari per canviar l'encapçalament i d'altres opcions com el nombre de posicions decimals de la variable de resposta o el text mostra.

Posteriorment adicionarem pel mateix mètode el temps que passen el clients en el sistema (Client.Totaltime) i el nombre de clients en el sistema (Client.WIP).

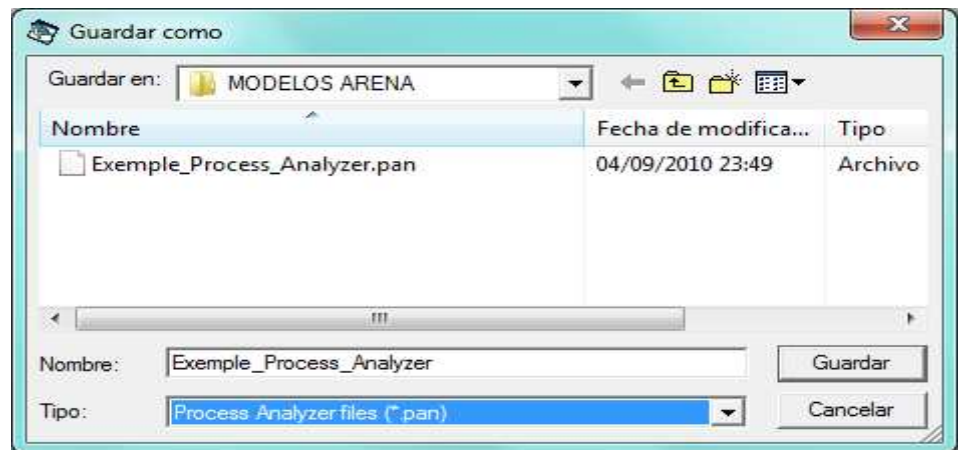
En aquest model no tenim cues com a variables de resposta i a continuació afegirem les relacionades amb el recursos, seleccionades sota la opció Resource de la mateixa forma que ho hem fet per a les entitats.

Seleccionarem el nombre promig Caixer.NumberBusy i l'utilització complerta del recurs s'addiciona mitjançant l'atribut Caixer.Utilization.

Amb la qual cosa podrem veure l'estat final de les variables de control a la figura següent de la finestra del projecte.

#### 4. GUARDAR EL PROJECTE AL DISC.

A continuació abans de crear diferents escenaris a partir d'aquest, procedirem a desar-lo en el directori del projecte amb el mateix nom que aquest, però amb l'extensió .pan que correspon als fitxers que utilitza el Process Analyzer.



#### 5. CREACIÓ DE VARIS ESCENARIS A PARTIR D'UN ESCENARI.

L'edició de la fulla de dades d'escenaris un cop finalitzada es veu de la següent manera:

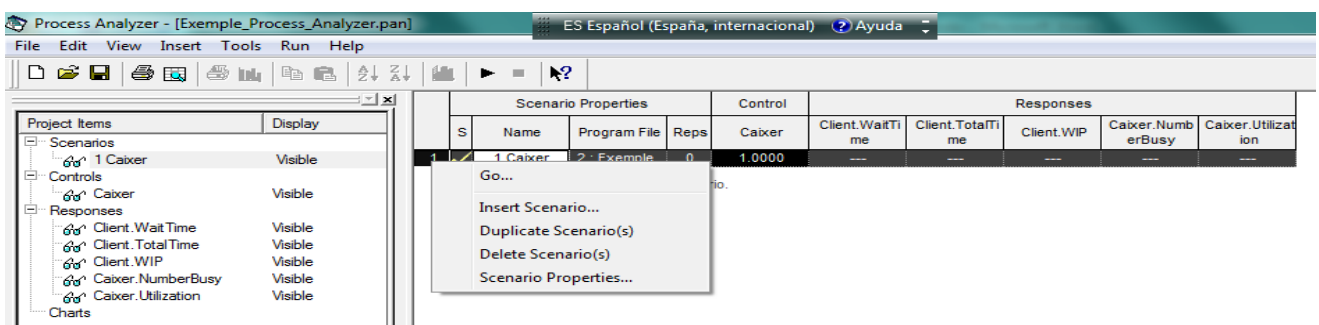
	Scenario Properties				Control	Responses				
	S	Name	Program File	Reps	Caixer	Client.WaitTime	Client.TotalTime	Client.WIP	Caixer.NumberBusy	Caixer.Utilization
1	<input checked="" type="checkbox"/>	1 Caixer	2 : Exemple_	0	1.0000	---	---	---	---	---

Double-click here to add a new scenario.

A la mateixa es pot veure el nombre de rèpliques executades (Reps = 0), la capacitat d'atenció del recurs caixer (Control/Caixaer = 1.0000) amb quatre decimals i les variables de resposta (Responses).

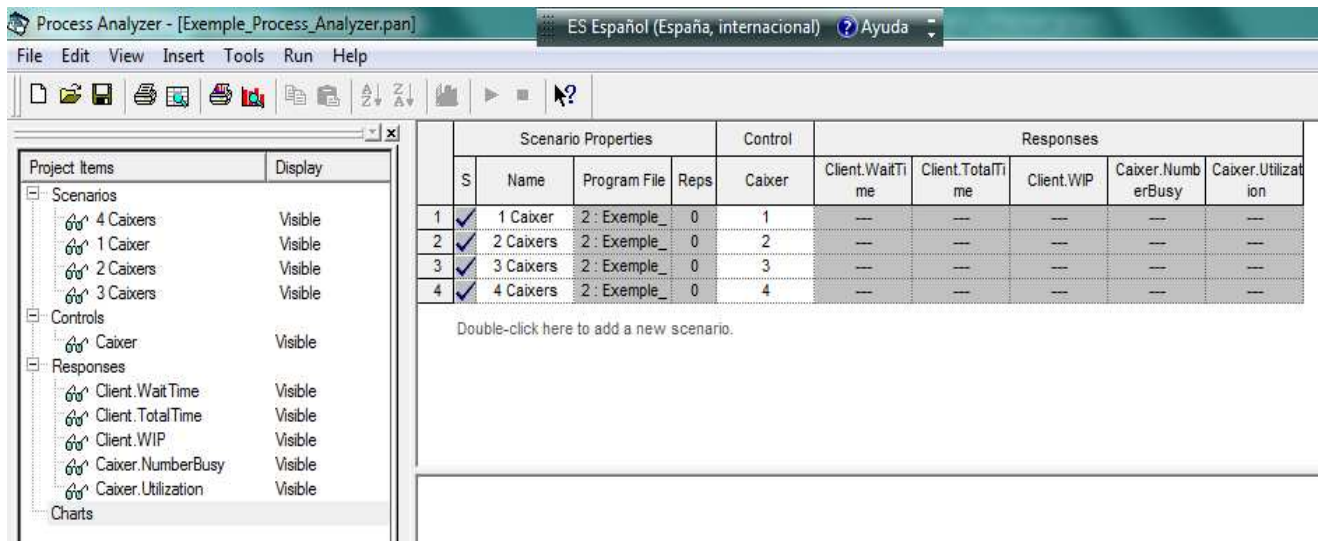
Per tal de crear diferents escenaris, es duplica l'escenari creat i es canvia el valor del control que correspongui.

Per tal de duplicar l'escenari el prem el botó dret del ratolí sobre el número del primer escenari i de la llista d'opcions que es desplega haurem d'escollir la opció Duplicate Scenario(s) tal i com s'observa a la figura següent:





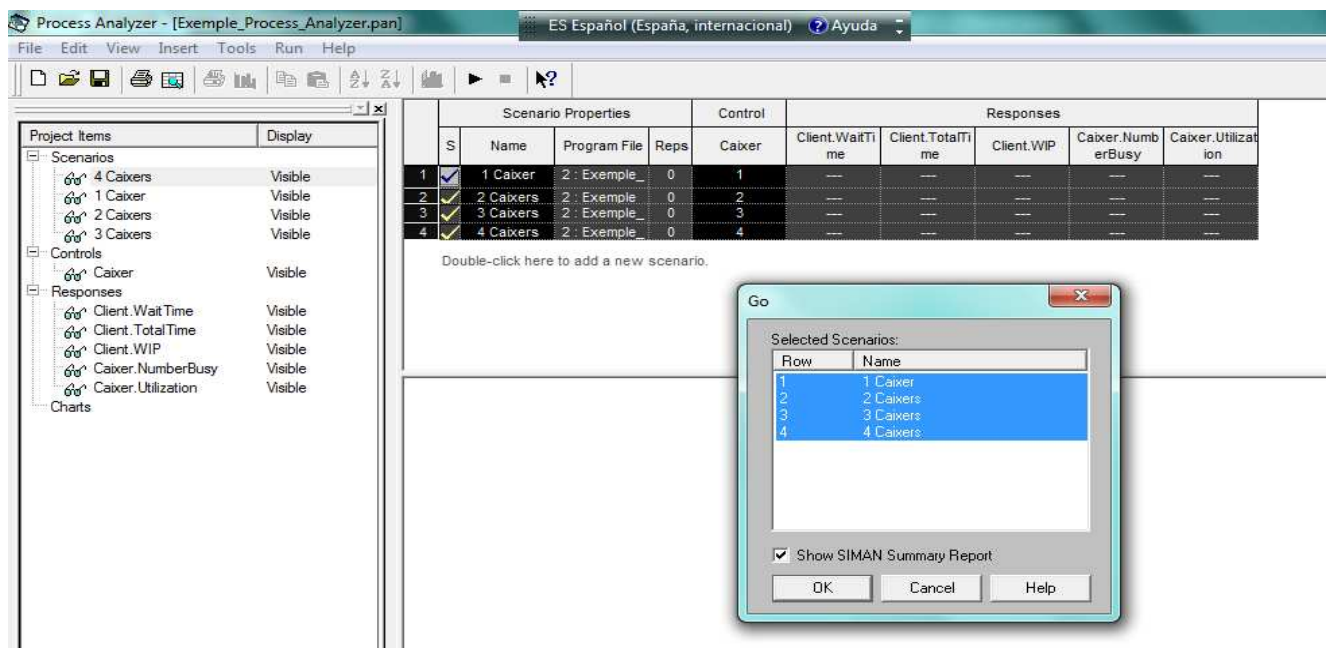
Procedirem a la creació de tres escenaris més, iguals al primer i canviarem el nombre de la capacitat d'atenció del recurs caixer (Control/Caixaer = 2, 3 i 4) i canviarem el nom de cada escenari. Quedant finalment la finestra del projecte tal i com es veu a la figura següent:



## 6. EXECUCIÓ DELS ESCENARIS.

Una vegada definits tots els escenaris que es volen avaluar, es selecciona primer mantenint polsada la tecla majúscules els quatre escenaris que es volen executar i es pressiona sobre el botó d'execució PLAY.

S'obrirà un nou quadre de diàleg anomenat GO, que ens preguntarà quins del quatre volem simular, els tornarem a seleccionar tots i si volem veure l'informe en llenguatge SIMAN.



Una vegada seleccionat OK el Process Analyzer apareixerà el símbol d'un corredor sota la columna S de Scenario Properties per a indicar quins escenaris es troben en execució i una banderola quan aquest finalitzi.

Els valors que es mostren a cada escenari es el promig del valors obtinguts en les rèpliques executades dins del sistema ARENA.

## 7. AVALUACIÓ DELS RESULTATS.

En els resultats s'aprecia que quan tenim una capacitat de 2, es redueix considerablement el temps d'espera en el sistema, passa de 6.210 minuts en l'escenari 1 a 0.059 minuts en l'escenari 2.

El nombre de caixers ocupats en l'escenari 1, indica que només un caixer pot estar capacitat per a les condicions del sistema, doncs la seva ocupació és més petita que la unitat, concretament 0.761.

En l'escenari 2 s'aprecia que la inclusió d'un caixer disminueix la utilització del recurs, passa de 0.761 en l'escenari 1 a 0.373 en l'escenari 2.

Project Items		Display	Scenario Properties				Responses				
S	Name	Program File	Reps	Caixer	Client Wait Time	Client WIP	Caixer Number Busy	Caixer Utilization	Client VATime	Client Total Time	
1	1 Caixer	12 Exemple	5	1	6.210	1.820	0.761	0.761	4.465	10.675	
2	1 Caixer	12 Exemple	5	2	0.059	0.756	0.746	0.373	4.382	4.442	
3	1 Caixer	12 Exemple	5	3	0.000	0.746	0.746	0.249	4.379	4.379	
4	1 Caixer	12 Exemple	5	4	0.000	0.746	0.746	0.186	4.379	4.379	

Així doncs si incrementem la capacitat d'atenció del temps per a les condicions donades per tal de fer que disminueixi el temps d'espera dels clients a la cua d'entrada, el temps ociós del recurs caixer s'incrementa. Aquest fet es reforça si observem el nombre de recursos ocupats, que és de 0.746, la qual cosa indica que tot i tenir dos caixers, no arribem a ocupar-ne un al cent per cent.

En els escenaris 3 i 4 els resultats són pràcticament els mateixos donat que per a les condicions donades, un increment en la capacitat no generarà millores significatives. Pel contrari, incrementarà els costos d'operació. En el cas dels recursos, només es necessiten 0.746 caixers per a efectuar les operacions i en tenim 3 i 4.

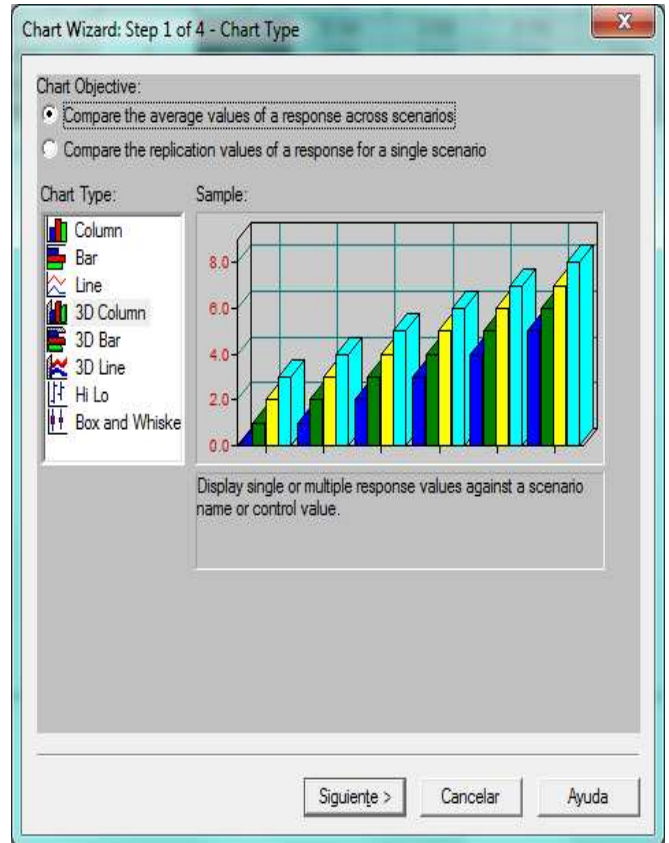
Resulta obvi que la utilització del servidor decreix a mesura que se l' incrementa la capacitat d'atenció i que l'excés de capacitat amb respecte als nivells de servei demandats disminueixen la utilització.

### 8. CREACIÓ DE GRÀFICS.

Sols es poden generar gràfics quan es tenen més d'una rèplica per a cada escenari (Reps > 1), la qual cosa es pot comprovar sota la columna Reps del Scenario Properties.

Aleshores es pot crear un gràfic que mostri el valor d'una o més variables de resposta, ja sigui per cada rèplica o per cada escenari.

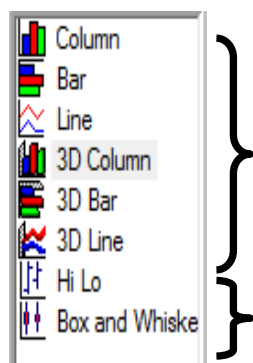
Per fer que s'activi la opció gràfic, primer hem de prémer amb el botó dret qualsevol columna de les variable de resposta (Responses). Mitjançant la opció de menú **Insert -> Chart...** o pel botó d'accés directe en forma de gràfic, s'obre el formulari de la dreta.



Com a primer pas podrem seleccionar entre dues opcions:

- I. Comparar el promig dels valors de totes les rèpliques d'una variable de resposta a través de diferents escenaris.
- II. Comparar el valors de la rèplica per a la variable de resposta en un sol escenari a escollir entre els millors.

També podrem seleccionar graficar intervals de confiança (almenys tres rèpliques per escenari) i diagrames de caixes, a més dels gràfics estàndard.

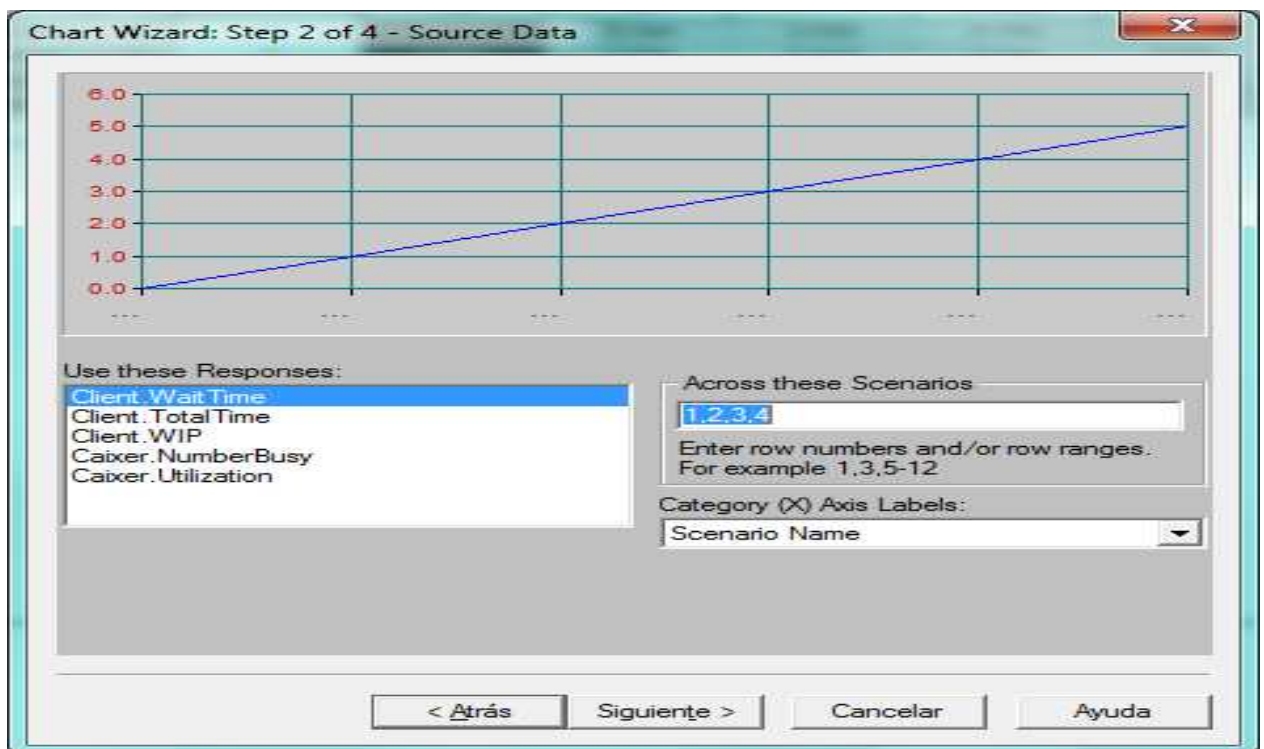


**Opcions de gràfics estàndard.**

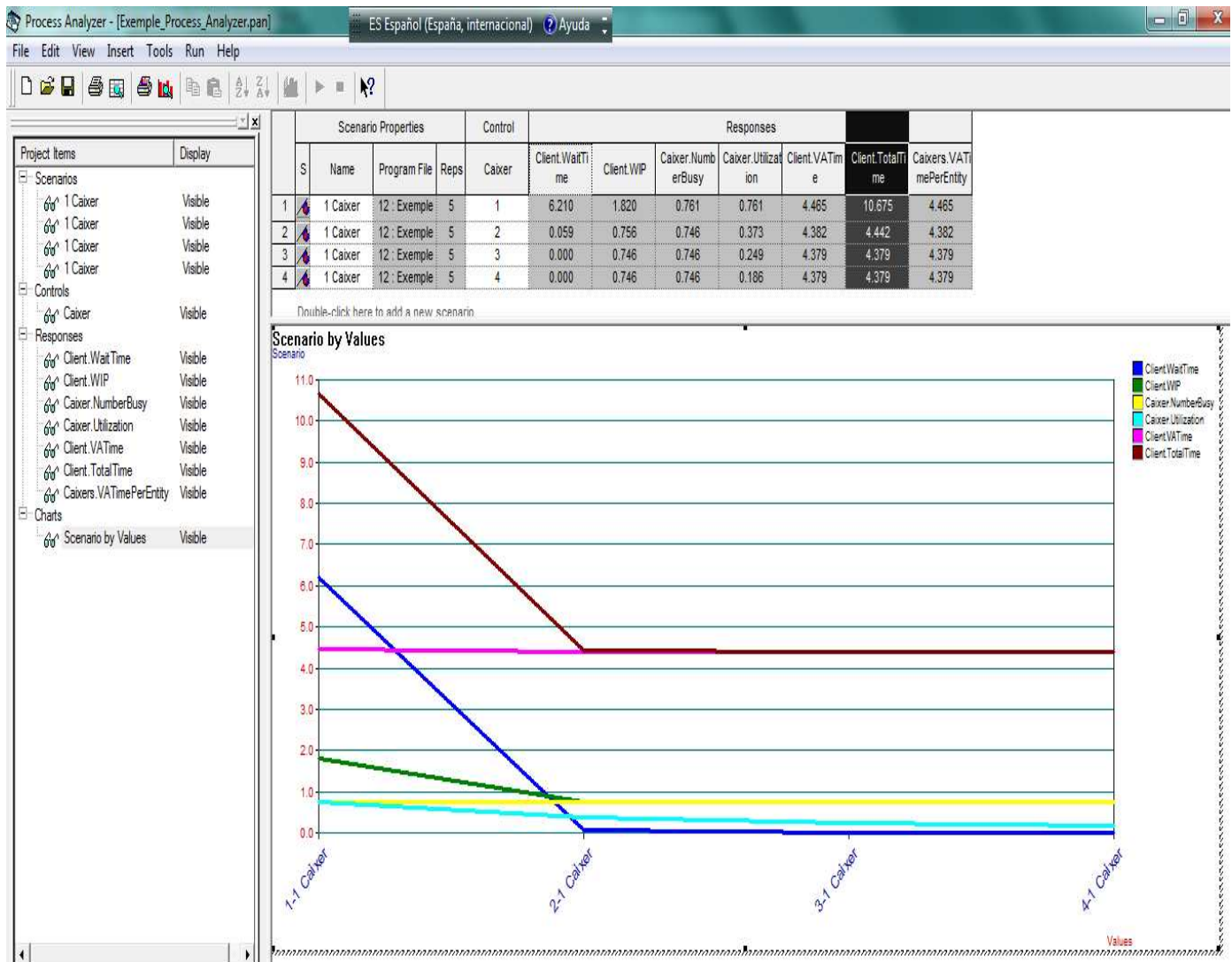
**Opcions per a graficar intervals de confiança. Es pot accedir a elles marcant la fila de la variable de resposta que es desitja. Posteriorment es pressiona la icona de l'assistent per a gràfics.**

Suposi que es desitja graficar el valor del temps d'espera en el sistema mitjançant tots els escenaris:

- Inicialment es fa clic en qualsevol cel·la de les variables de resposta, per habilitar la icona de l'assistent per a gràfics.
- A continuació s'inicia l'assistent.
- En l'assistent es selecciona una variable de resposta a través de diferents escenaris. Posteriorment s'escull el tipus de gràfic, en aquest cas es tracta d'un de línies.
- Posteriorment es prem el botó següent per a continuar en el següent pas per escollir les variables de resposta que volem graficar. També podem seleccionar una o més variables de resposta a través de qualsevol o tots els escenaris i podem canviar la descripció de la categoria sota l'eix X.
- El pas següent consistirà en especificar el nom per als gràfics.
- L'últim pas serà útil quan es grafica una variable de resposta, doncs permet especificar a l'assistent que assenyali o ressalti el millor escenari segons sigui el cas (minimització o maximització de la variable de resposta). Si no ho necessitem simplement premem següent.
- És possible canviar opcions de format del gràfic, com el gruix de les línies o l'alineació dels rètols de les categories sota l'eix X.



El gràfic generat es mostra a continuació:



En el gràfic es corrobora l'anàlisi fet dins el pas 7 i el mateix ajuda notablement a veure els resultats del procés.

Les millores notables es donen en l'escenari 2, una millora poc significativa es mostra en 3 i cap millora es presenta en l'escenari 4.

Així doncs l'òptim ja és l'escenari 2. En el mateix la línia blau fosc, que correspon a l'espera (Client.WaitTime), arriba pràcticament a zero i els clients no esperen. Per la qual cosa el temps promig d'estada de cada client al sistema, graficat per la línia marrón (atribut Client.TotalTime),

tendeix a ser el promig del temps d'atenció per a cada client (Caixers.VATimePerEntity). El qual val 4,40, doncs prové d'una distribució exponencial amb mitja 4,5.

## 9. UTILITZACIÓ D'UNA VARIABLE COM A RESPOSTA.

Fins ara s'ha trobat que amb dos caixers el temps d'espera dels clients es quasi 0, per lo tant, per a un banc seria convenient decidir-se per aquesta opció.

Un altre anàlisi que es podria realitzar és el de comprovar els efectes de canvis en els factors externs, els quals formen la realitat incontrolable, per a les condicions actuals o proposades.

Suposem que en el cas d'estudi actual s'ha fixat un temps d'espera promig de 3 minuts com estat d'atenció a un client. En les condicions proposades, que passaria si l'arribada de clients s'incrementa?.

Aquest increment s'assumirà mitjançant una disminució de la mitja d'arribades per unitat, es a dir, per al supòsit de treballar amb dos caixers, es tenen els següents escenaris:

Escenari	Valor de la mitja d'arribades
Escenari 1	5.89
Escenari 2	4.89
Escenari 3	3.89
Escenari 4	2.89
Escenari 5	1.89

Per tal de resoldre aquest cas, s'ha de modificar el model mitjançant el Process Analyzer, de tal manera que la mitjana de la distribució utilitzada per a generar les arribades sigui una variable.

L'expressió que s'utilitza en el mòdul CREATE per a generar les arribades dins del model original és:

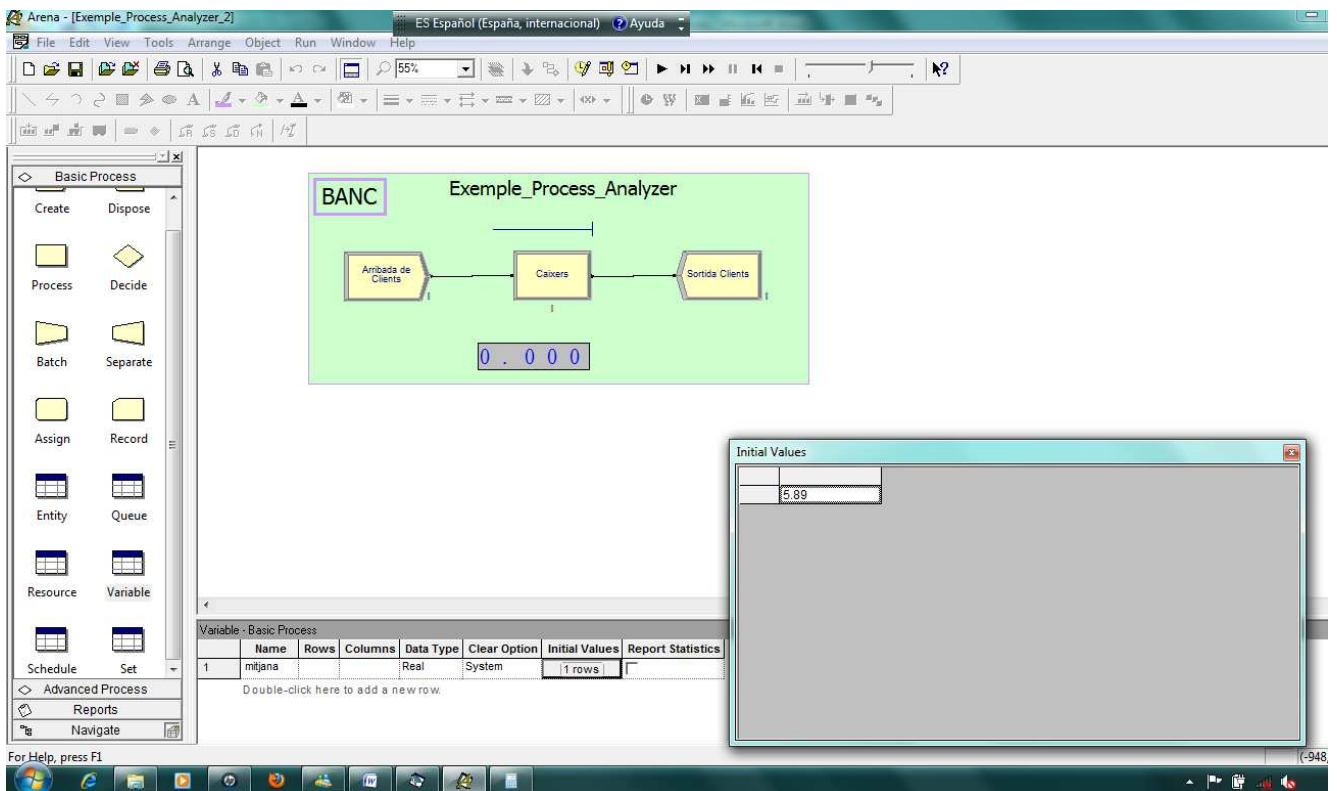
**NORM (5.89,0.65)**

Per a realitzar l'anàlisi aquesta expressió s'ha de canviar per la següent:

**NORM (mitjana,0.65)**

En la qual, “mitjana” és una variable el valor inicial de la qual és 5.89 i romandrà constant a través de tota la simulació, doncs no existeix cap mòdul lògic dins del sistema, que posteriorment a la creació de l'entitat incideixi sobre el valor d'aquesta variable.

Per la qual cosa hem de definir dins del model ARENA una variable mitjançant el panell de processos bàsics i el mòdul de dades VARIABLE amb valor inicial 5.89 i el següents segons la taula anterior, tal i com es mostra a la figura següent:



El nou model es guardarà amb un nom diferent, Exemple\_Process\_Analyzer\_2.doe, i a continuació revisarem el model mitjançant F4.

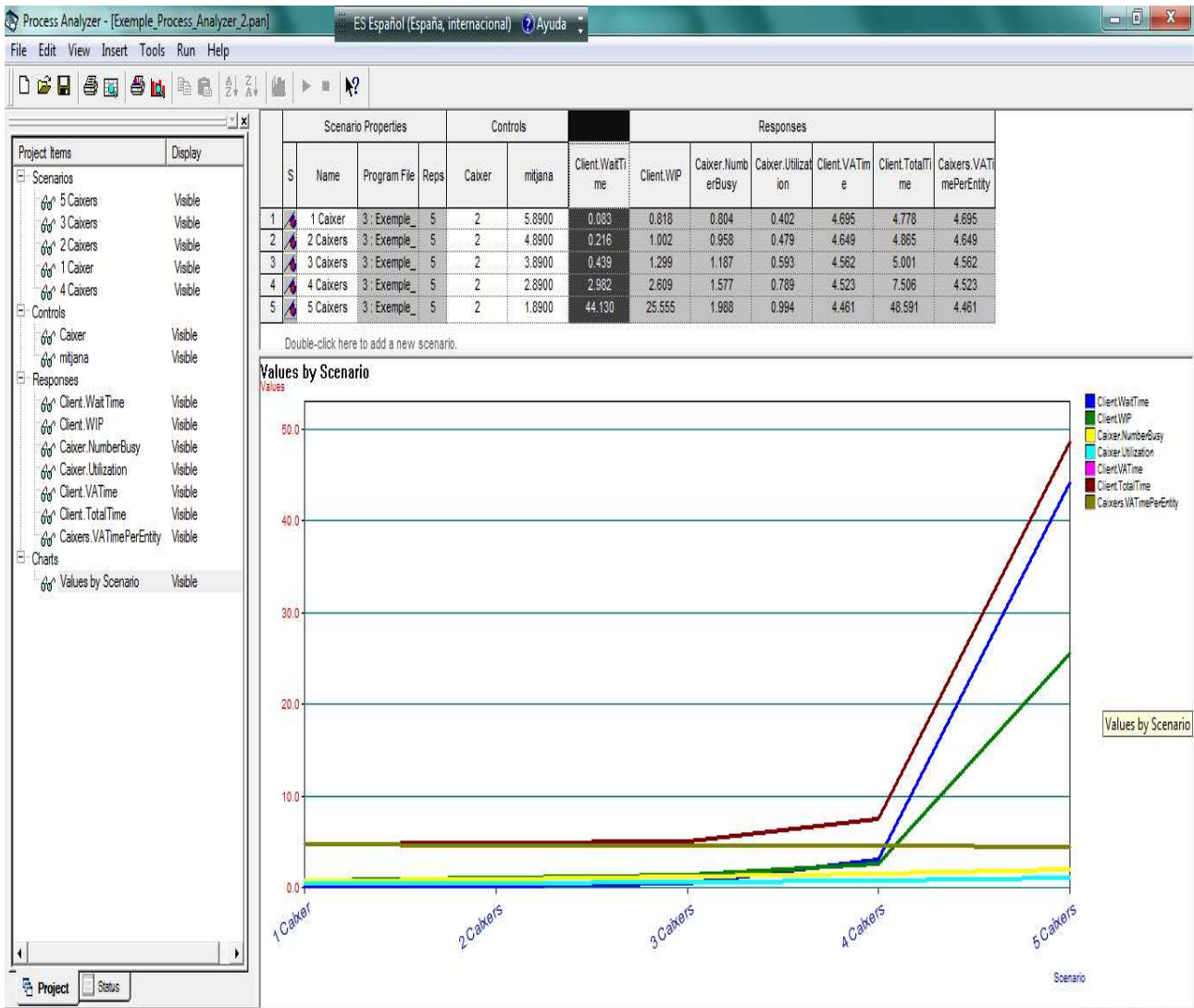
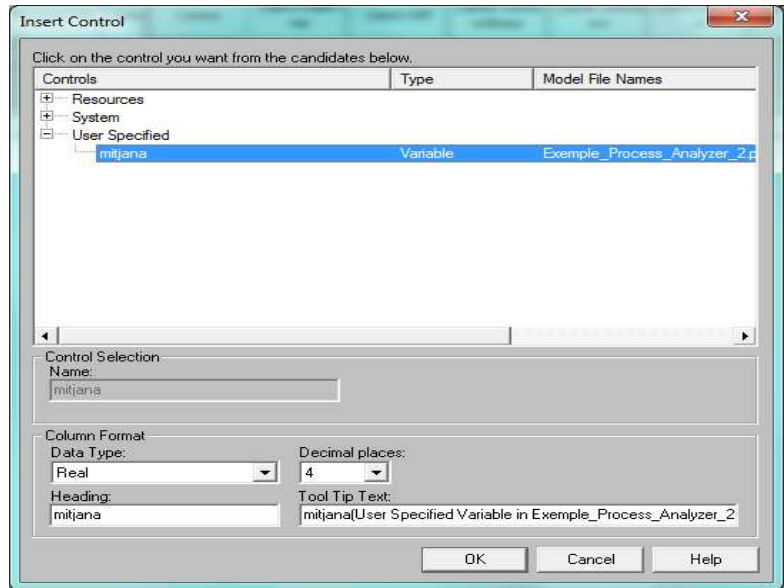
En el Process Analyzer es preparen els escenaris de la mateixa manera des del pas 1, afegint un control més que correspon a la variable que representa el temps entre arribades.

La mateixa apareixerà sota una nova categoria dins les variables de control, User Specified. La qual ha sorgit un cop mitjançant el botó Browse hem buscat dins del directori on tenim guardat el model, el nou fitxer Exemple\_Process\_Analyzer\_2.p que ens servirà per a realitzar l'anàlisi amb la variable mitjana.

Una vegada s'ha afegit la variable de control mitjana, s'han de duplicar els escenaris fins tenir-ne un total de cinc.

A continuació modificarem els valors de les variables de control, caixer i mitjana, per a cada escenari i es faran córrer els diferents escenaris. Es deixarà el nombre de caixers sempre a 2 i la mitjana segons els valors de la taula anterior per a cada escenari.

Ara s'obtenen els següents resultats, segons es pot veure a la següent figura :



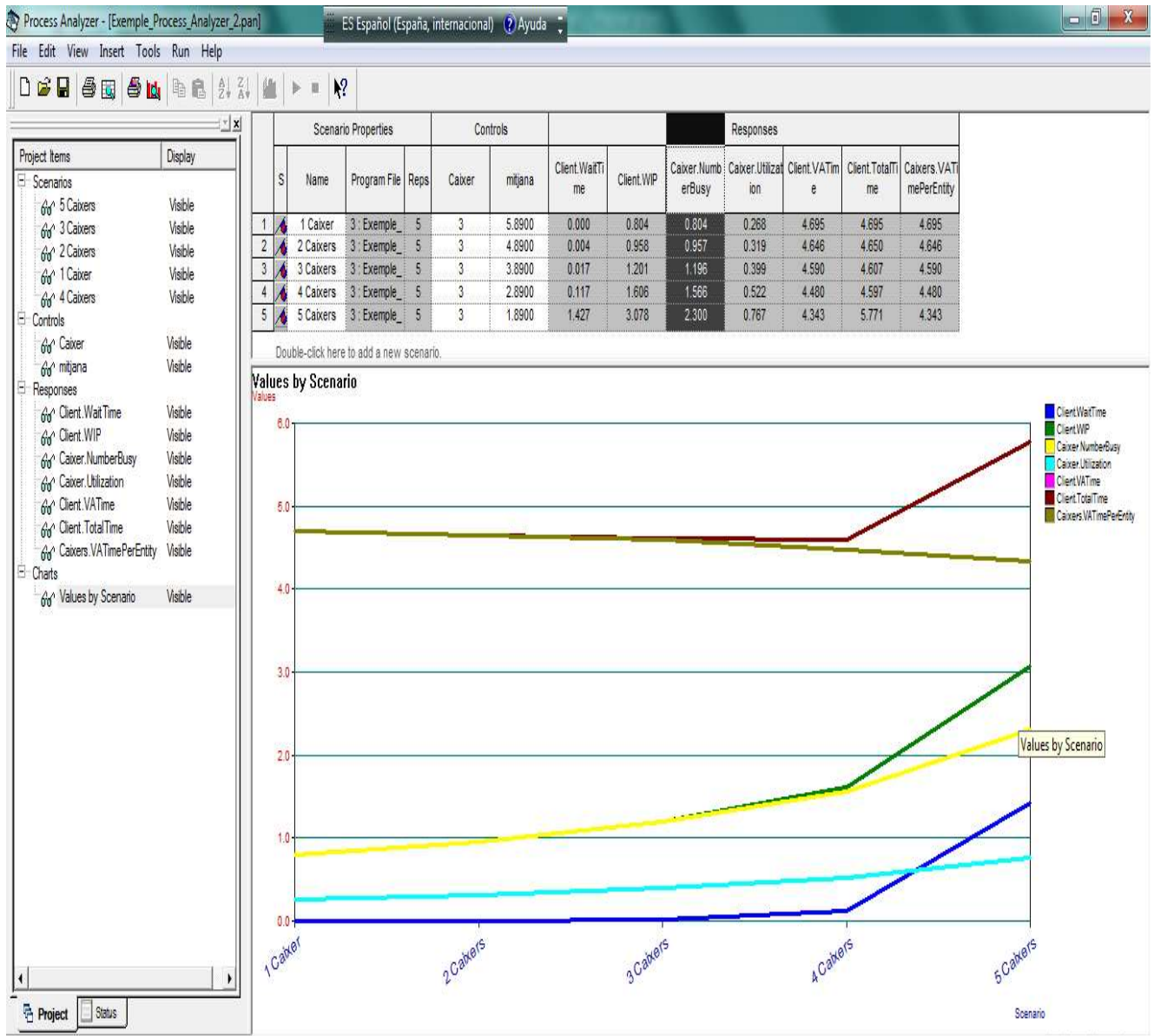
Com es pot comprovar en els resultats, aquesta política només suporta fins a una variació de 2.982 minuts en l'escenari 4 per a la mitja de temps entre arribades de clients, per sota del temps d'espera



promig de 3 minuts com estat d'atenció a un client. En les condicions proposades a més també compleix amb els estàndards d'atenció prèviament establerts, doncs no serà fins l'escenari 5, quan els clients arriben per sota la mitjana d'arribada molt per sota dels 3 minuts, concretament 1,89 minuts, s'hauran d'esperar quasi 44 minuts de mitjana per a ser atesos.

Però el cost és alt sempre, donat que mai s'ocupa més de 1 caixer en qualsevol escenari.

De totes maneres, encara es podria realitzar una última simulació ampliant el nombre de caixers a 3 i veure com es comporta el sistema. Aquest resultat es mostra a la figura següent:



Amb tres caixers ja no es dona el cas d'alta espera en l'escenari 5, però de nou el cost és encara més alt sempre, donat que mai s'ocupa més de 1 caixer en qualsevol escenari i ara ja en tenim tres.

Com es pot veure en l'anterior gràfica, la situació més crítica (mitjana d'arribades 1.89 minuts) ocasiona un temps d'espera de menys de 2 minuts i si no fos pel cost seria ideal.

## 4.7.2. ARENA Output Analyzer.

La simulació genera una història artificial d'un sistema i tracta d'aproximar el comportament de la realitat mitjançant nombres aleatoris per a la generació de les variables d'entrada com el temps entre arribades o el temps entre processos.

Com a resultat de la simulació mitjançant aleatorietat en les variables d'entrada s'obtidran unes variables o dades de sortida, les quals es comportaran també d'una forma completament aleatòria. Es a dir, mostren certa variabilitat, la qual s'haurà de tenir en compte en el moment d'interpretar els resultats que dóna el model.

D'aquesta manera, es crea la necessitat de realitzar un anàlisi estadístic de les dades de sortida mitjançant histogrames, gràfics de procés acumulats, intervals de confiança, correlogrames, entre d'altres. També quan es comparen dos dissenys o situacions en funció d'una variable de resposta, es recorre a probes de diferències de mitjanes i anàlisi de variància (un factor).

Aquest tipus d'anàlisi es pot realitzar d'una forma ràpida mitjançant una altra eina inclosa dins de l'entorn **ARENA** anomenada **Output Analyzer** (analitzador de sortida).

Per tal de desenvolupar anàlisi amb aquesta eina, es fa necessari disposar d'un arxiu amb les dades de sortida del model generats mitjançant una o vàries simulacions del model.

Els arxius de sortida de dades es creen quan quant al programa se li especifica el nom i l'extensió del fitxer en el qual es volen emmagatzemar les observacions individuals d'una estadística en particular.

Per exemple, si es vol prendre el temps de flux o temps en el sistema d'una entitat com a mesura del rendiment del sistema, aleshores les entitats es guardaran dins d'un arxiu de sortida. Aquest fitxer contindrà per a cada entitat dades gravades del temps del flux, les quals es poden graficar, trobar el seu promig acumulat, construir un histograma, etc.

El **Output Analyzer** no es troba dins de la instal·lació estàndard del programa ARENA, si no l'hem instal·lat mitjançant l'opció personalitzada i s'activa la casella de verificació corresponent no hi podrem accedir mitjançant el menú principal. Aleshores, el programa s'ha d'executar mitjançant l'executable anomenat OUTPUT, el qual es troba dins de la carpeta arrel d'instal·lació del programa ARENA en el nostre entorn WINDOWS.

Per tal de realitzar un anàlisi de les dades de sortida mitjançant el **Output Analyzer** haurem de seguir els següents passos:

### 1. CREACIÓ DELS ARXIUS DE SORTIDA.

Els arxius han de sortir del model per a ser analitzats per l' **Output Analyzer**, han de tenir un format especial i una extensió .dat per tal de poder ser reconeguts.

Aquests arxius .dat es creen mitjançant el mòdul de dades incorporat dins del *template Advanced Process* denominat **STATISTIC**, els qual es van omplint d'informació a mesura que avança la simulació i les rèpliques que es facin dins de cadascuna d'elles.

<b>Especificacions del mòdul avançat STATISTIC</b>	
<b>Camp</b>	<b>Descripció</b>
<b>Name</b>	Nom del mòdul, ha de ser únic
<b>Type</b>	Tipus d'estadística. Pot ser observable (Tally), freqüència d'ocurrències (Frecuencies), etc.
<b>Expression/Name</b>	Expressió o nom de l'estadística que es genera. Es possible que apareixin altres camps addicionals segons el tipus d'informació recol·lectada.
<b>Report label</b>	Rètol que apareix en els reports per tal d'identificar l'estadística. Pren automàticament el valor del camp Name, però pot ser modificable.
<b>Output file</b>	Nom de l'arxiu de sortida on s'emmagatzemaran les dades per al seu anàlisi posterior. Aquest camp s'ha d'omplir amb el nom, seguit de l'extensió .dat. Per exemple, Info.dat

Una vegada entrem dins del **Output Analyzer** tenim la següent barra estàndard. La mateixa conté quasi totes les instruccions dels anàlisi que es poden realitzar amb aquesta eina, la qual es mostra a continuació:



A partir de la sisena icona, trobem les opcions per tal de generar: 7- Gràfics de barres, 8 - Histogrames, 9 - Promig mòbil, acumulat i exponencial, 10 – Plot simple (variable de resposta en el temps), 11- Sub agrupaments de dades, 12- Correlograma i 13- Interval de confiança.

## 2. CREACIÓ DE L'ARXIU DE SORTIDA \*.DAT

Per mostrar la creació d'aquest arxiu farem servir el model utilitzat dins l'apartat **4.6.2.4. Estadístiques del temps entre arribades**. Exemple\_Generacio\_Analisi\_Reports\_Arribades

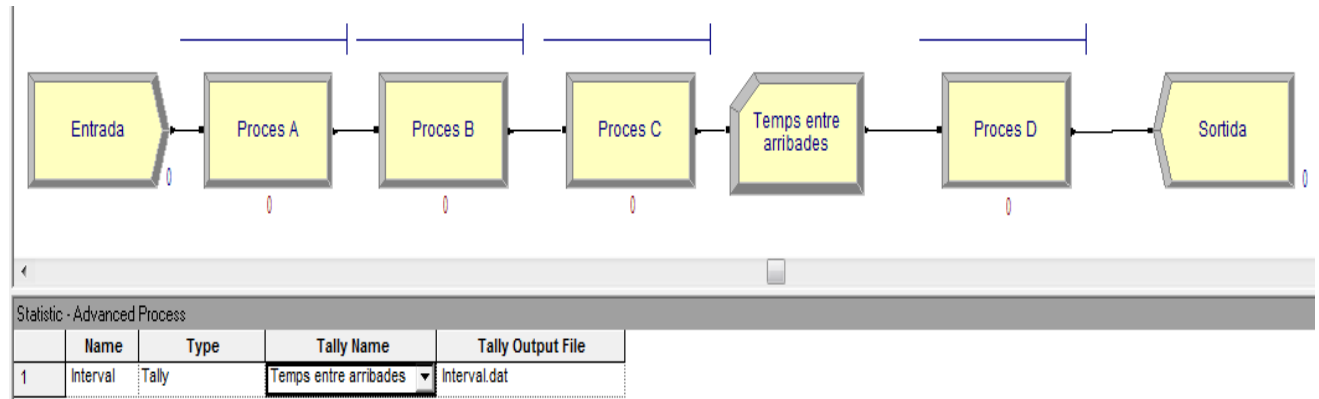
En el mateix mitjançant el mòdul **RECORD** anomenat **Temps entre arribades** s'ha creat una estadística observable o *Tally* del tipus *Time Between*, la qual anomenem **Temps entre arribades** i recol·lectarà la diferència de temps amb la que arriben les entitats dins del mòdul Procés D.

Procedirem a modificar aquest exemple, afegint el mòdul de dades **STATISTIC** del template *Advanced Process*, en el qual s'ha d'afegir una fila amb un nou estadístic anomenat **Interval** del tipus *Tally*. El qual emmagatzemarà dins el fitxer binari **Interval.dat** les dades observades pel mòdul

**RECORD** mitjançant la variable **Temps entres arribades**, a la qual fem referència també dins del mòdul **STATISTIC** seleccionant-la sota el desplegable **Tally Name** on ja hi apareix per defecte.

Totes les variables o atributs generats en qualsevol mòdul del model, seran accessibles de forma automàtica en els demás mòduls del mateix model en el qual siguin susceptibles de ser utilitzades.

A continuació es mostra el model modificat, el qual queda igual a excepció de l'afegit mòdul de dades **STATISTIC**:



El model es guardarà amb el nom **Exemple\_Output\_Analyzer.doe**

A continuació executarem el model amb un temps especificat de 2400 minuts per tal d'omplir el fitxer **Interval.dat** amb les dades de cada rèplica.

Com es pot veure observant el menú del **Output Analyzer** l'eina aporta moltes opcions d'anàlisi per a les dades de sortida dels models simulats. Degut a la complexitat que comportaria realitzar un estudi de cadascuna d'aquestes opcions, dins l'àmbit d'aquest projecte s'han escollit cinc opcions: Generació de l'histograma, Interval de confiança per a la mitjana, Promig acumulat, Comparació de mitjanes i Anàlisis de variància per a un sol factor. Els quals seran analitzats en detall dins dels propers apartats.

### 4.7.2.1. Generació de l'histograma.

L'arxiu de sortida **Interval.dat** serà la font de les dades per a construir l'histograma mitjançant el simulador **Output Analyzer**.

Si consultem el report d'entitats es pot veure que han entrat 232 (Number in) i surten 230 (Number out).

Time		NVA Time	Other Time	Total Time	Transfer Time	VA Time
Part		0.00	0.00	58.78	0.00	28.61
<b>Total</b>		<b>0.00</b>	<b>0.00</b>	<b>58.78</b>	<b>0.00</b>	<b>28.61</b>

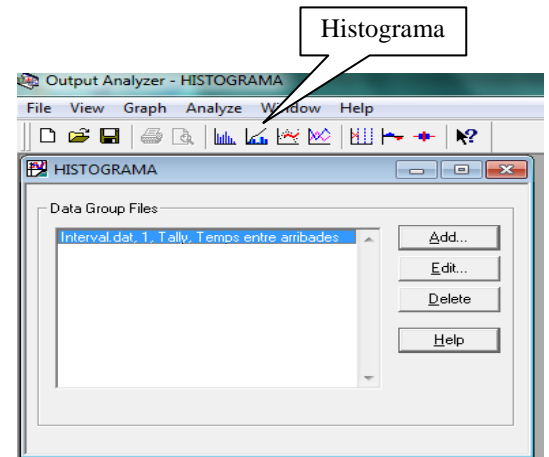
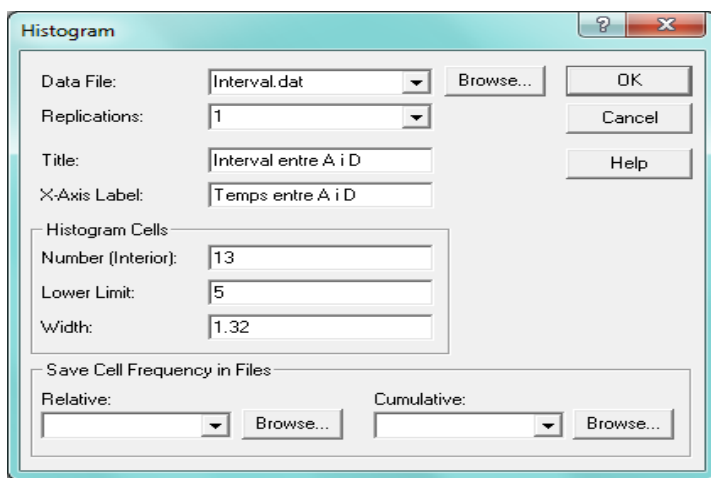
Other		Number In	Number Out
Part		232	230
<b>Total</b>		<b>232</b>	<b>230</b>

Es pot dir que almenys l'estadística conté en total 230 observacions que corresponen a les entitats que varen ser retirades del sistema en el moment de finalitzar la simulació, a les quals anomenem  $n$ .

El nombre d'interval·s de l'histograma sortirà del resultat de la fórmula  $\sqrt{n}$  i en aquest cas en concret serà de  $\sqrt{230} = 15$  interval·s.

Per a generar l'histograma obrirem el simulador **Output Analyzer** i carregarem el fitxer anomenat **HISTOGRAMA.dgr** on per defecte es carregarà el fitxer de dades creat en el pas anterior **Interval.dat**.

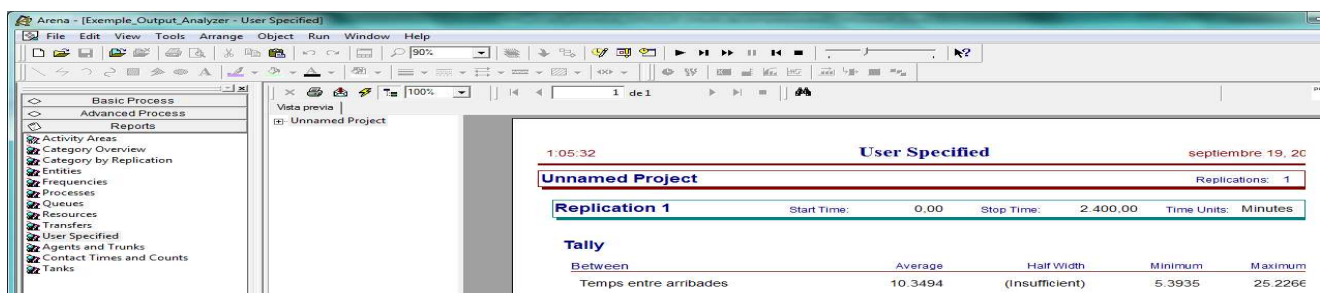
A continuació seleccionarem la icona corresponent a l'histograma de la barra estàndard i apareixerà el següent formulari:



Dins del mateix donarem un títol al gràfic i a l'eix X i a continuació mostrarem el mètode per a realitzar el càlcul de les dades que hem de col·locar dins de l'apartat **Histogram Cells**.

Inicialment s'ha de tornar al report generat per ARENA quan s'ha executat el model **Exemple\_Output\_Analyzer.doe**.

Dins del report s'ha d'obrir els estadístics generats mitjançant el mòdul **RECORD** i que es troben dins la opció **User Specified**, tal i com es mostra a continuació:



Agafarem les dades **Maximum = 25,22** i **Minimum = 5,3935**.

Tornant al formulari de l'histograma col·locarem com a **Number (interior)** s'hauria de posar el nombre d'interval·s calculat anteriorment, el qual és 15. Però quan es construeixen histogrames mitjançant aquesta eina s'han de tenir en compte que es creen dos interval·s o classes predeterminats, un des de **-Infinity** a **Minimum Classe** i una altra de **Maximum Classe** a **+Infinity**.

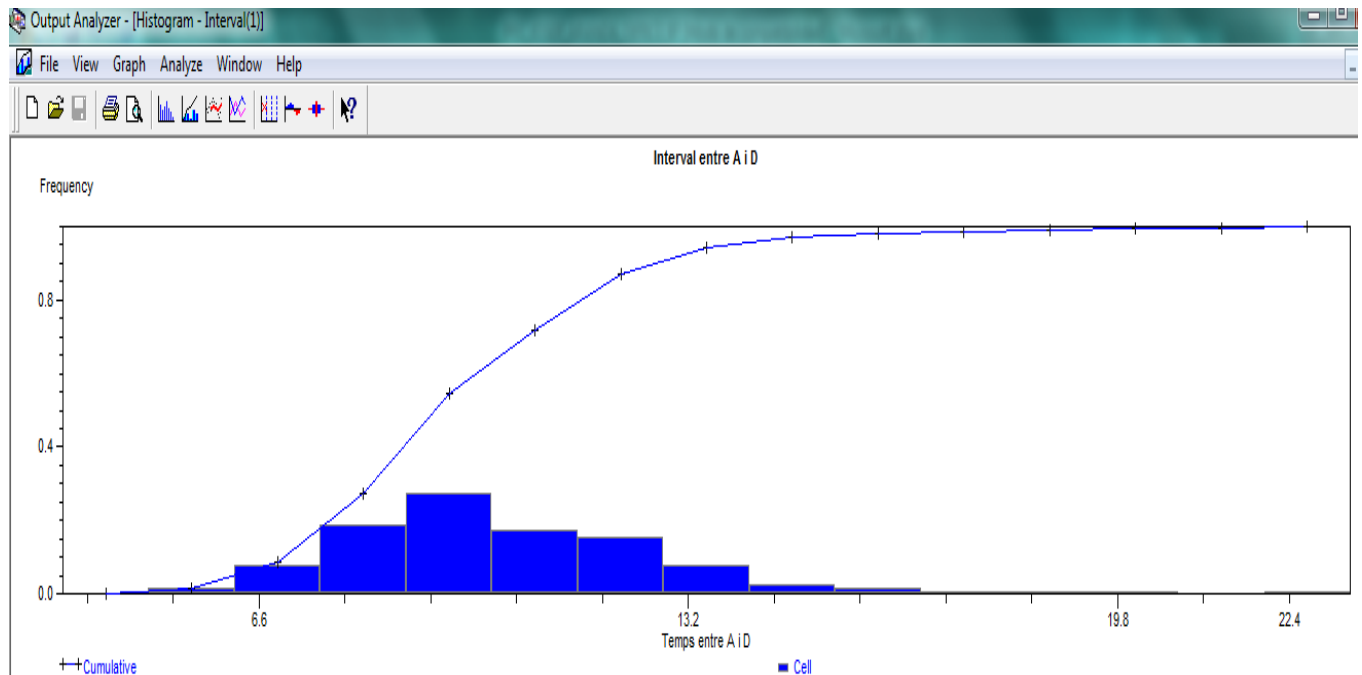
Aleshores si volem crear un histograma de m classes o intervals, s'ha de posar com a **Number(interior)** m-2 classes. En aquest cas en particular es vol construir un histograma de 15 classes, aleshores, el camp mencionat d'ha d'omplir amb el valor 13.

Com a **Lower Limit** el valor enter inferior de la dada **Minimum**, o sigui 5.

La dada que posarem dins la opció **Width**, amplada de l' interval o classe, sortirà del següent càlcul :

$$(\text{Maximum} - \text{Minimum}) / \text{Interval} = (25,22-5,39) / 15 = 1,32$$

A continuació fem clic a OK i apareixerà el següent histograma:



Histogram Summary							
Interval entre A i D							
Cell	Cell Limits		Abs. Freq.		Rel. Freq.		Cumul.
	From	To	Cell	Cumul.	Cell	Cumul.	
1	-Infinity	4.9	0	0	0	0	0
2	4.9	6.22	3	3	0.0131	0.0131	
3	6.22	7.54	17	20	0.07424	0.08734	
4	7.54	8.86	43	63	0.1878	0.2751	
5	8.86	10.18	62	125	0.2707	0.5459	
6	10.18	11.5	39	164	0.1703	0.7162	
7	11.5	12.82	35	199	0.1528	0.869	
8	12.82	14.14	17	216	0.07424	0.9432	
9	14.14	15.46	6	222	0.0262	0.9694	
10	15.46	16.78	3	225	0.0131	0.9825	
11	16.78	18.1	1	226	0.004367	0.9869	
12	18.1	19.42	1	227	0.004367	0.9913	
13	19.42	20.74	1	228	0.004367	0.9956	
14	20.74	22.06	0	228	0	0.9956	
15	22.06	+Infinity	1	229	0.004367	1	

Adicionalment, es crea un resum amb la taula de freqüències, la qual es mostra a continuació:

En aquesta taula es presenten els resultats de freqüències observades, relatives per

classe i acumulades. També s'aprecien les dues classes predeterminades i les 13 que s'han inclòs entre aquestes per a fer un total de 15.

Les freqüències acumulades (**columna Abs. Freq.**) i relatives (**columna Rel. Freq.**) es poden guardar dins de dos arxius externs, el quals només es podran obrir des de **Output Analyzer**, si omplim les dades del formulari de l' histograma dins de l'apartat **Save Cell Frequency in Files** amb el nom que volem donar a cada fitxer.

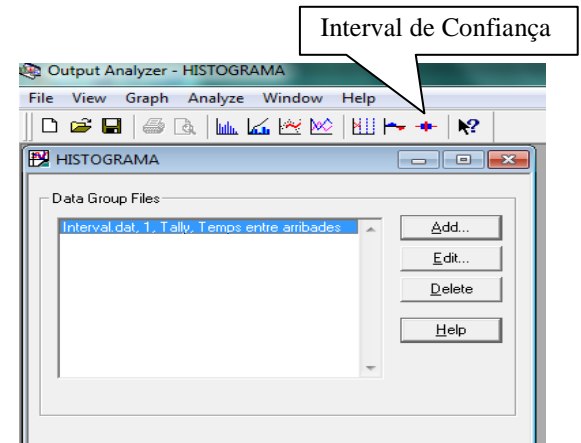
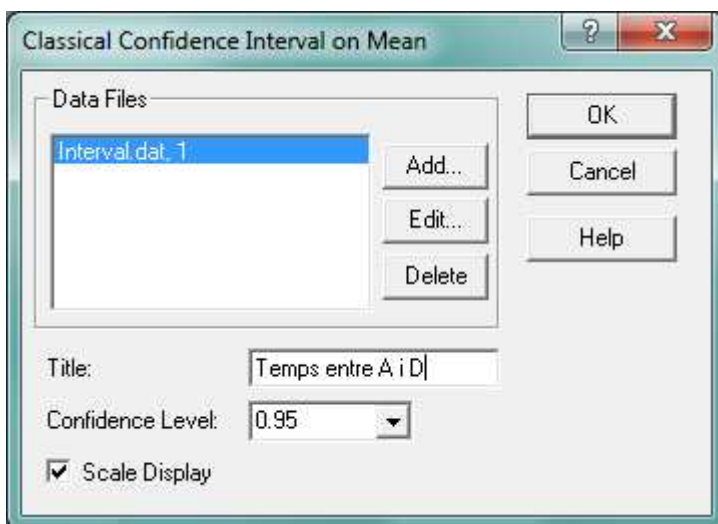
### 4.7.2.2. Interval de confiança per a la mitjana.

L'analitzador de dades de sortida **Output Analyzer** també permet calcular, a un nivell de confiança específic, l'interval de confiança per a la mitjana d'un estadístic determinat.

Seguint l'exemple, l'arxiu de sortida **Interval.dat** també serà la font de les dades per a calcular amb un nivell de confiança del 95% un interval de confiança per a la mitjana del **Temps entre arribades**.

Per a calcular l'interval de confiança obrirem el simulador **Output Analyzer** i carregarem el fitxer anomenat **HISTOGRAMA.dgr** on per defecte es carregarà el fitxer de dades creat en el pas anterior **Interval.dat**.

A continuació seleccionarem la icona corresponent a **Conf. Int. (mean)** de la barra estàndard i apareixerà el següent formulari:



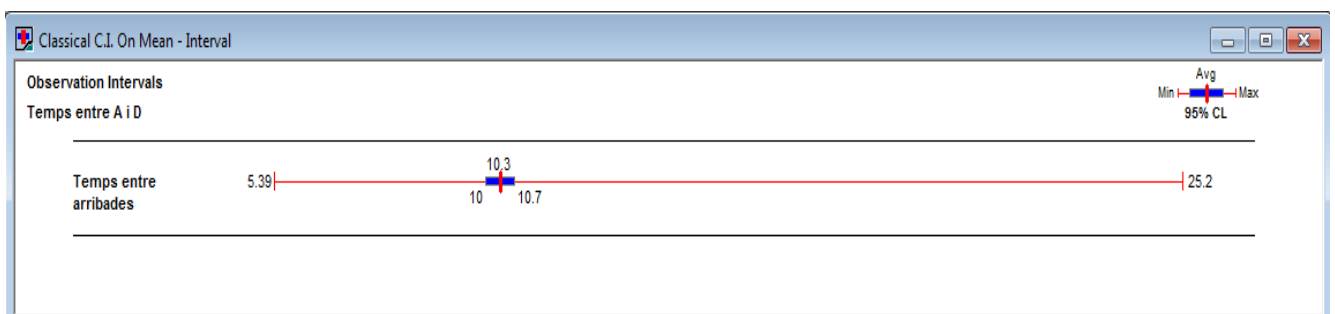
Afegirem el fitxer de dades dins l'apartat **Data Files**, tenint en compte que si es vol també podem afegir més fitxers alhora amb d'altres dades per a calcular diferents intervals de confiança.

A continuació posarem com a títol del gràfic que mostrarà l'interval de confiança el de **Temps entre A i D** i deixarem el nivell de confiança que surt per defecte, 0.95 que

correspon al 95%.

Si activem la opció **Scale Display** en el cas que s'hagi optat per mostrar diferents intervals de confiança de cop, els gràfics es mostraran sota la mateixa escala.

A continuació fem clic a OK i apareixerà el següent gràfic mostrant l'interval de confiança generat:



El qual va de 10 a 10,7 per a la mitjana de 10,3 amb un 95% de fiabilitat e indica els valors extrems.

Adicionalment es genera un report com el que es mostra a continuació:

Classical C.I. Intervals Summary						
Temps entre A i D						
IDENTIFIER	AVERAGE	STANDARD DEVIATION	0.950 C.I. HALF-WIDTH	MINIMUM VALUE	MAXIMUM VALUE	NUMBER OF OBS.
Temps entre arribades	10.3	2.52	0.328	5.39	25.2	229

En l'anterior informe es mostren per ordre d'esquerra a dreta, l'identificador de l'estadístic observat (Temps entre arribades), el promig (10,3), la desviació estàndard (2,52), la meitat de l'interval de confiança per a la mitjana (0,328), el valor mínim (5,39) i el valor màxim (25,2), de les observacions recollides (229).

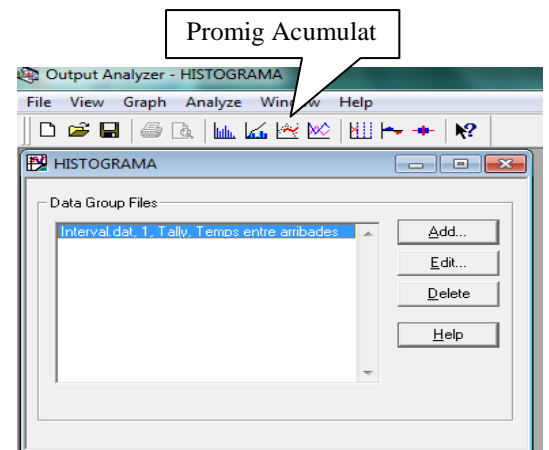
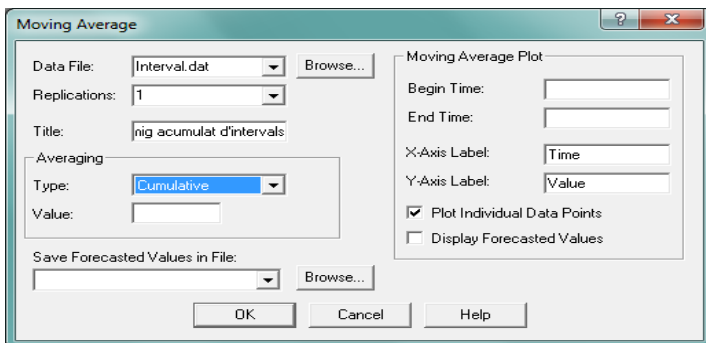
### 4.7.2.3. Promig acumulat.

El promig acumulat resulta de gran utilitat per a determinar el període d'estabilització del valor promig d'una variable de resposta en el model, el que permet trobar la durada de la simulació.

Seguint l'exemple, l'arxiu de sortida **Interval.dat** també serà la font de les dades per a calcular el promig acumulat. Mitjançant el qual es calcularà el temps de simulació necessari per trobar un valor per al **Temps entre arribades** estabilitzat, un cop passat el temps d'estabilització també anomenat de preescalfament.

Per a veure la gràfica per al promig acumulat obrirem el simulador **Output Analyzer** i carregarem el fitxer anomenat **HISTOGRAMA.dgr** on per defecte es carregarà el fitxer de dades creat en el pas anterior **Interval.dat**.

A continuació seleccionarem la icona corresponent a **Moving Average** de la barra estàndard i apareixerà el següent formulari:



Escollirem el fitxer de dades **Interval.dat**.

Sota el camp **Replications** seleccionem la rèplica a la qual s'ha realitzat l'anàlisi mitjançant el nombre de la mateixa, també podem es poden prendre diferents rèpliques sota una mateixa gràfica si escollim la opció **All** amb les dades barrejades o



podem graficar totes les rèpliques al mateix temps mitjançant l'opció **Lumped**.

L'opció de mitja mòbil genera les dades sense problemes per una mitjana de les observacions reals d'un determinat arxiu de dades o mitjana acumulativa, fins a l'observació actual (**Type = Cumulative**), o per un nombre determinat de punts o mitjana mòbil (**Type = Moving**) que produeix una gràfica per als valors reals i previstos.

També pot ser utilitzat per a generar una mitjana mòbil ponderada exponencialment o suavitzat exponencial (**Type = Exponential**).

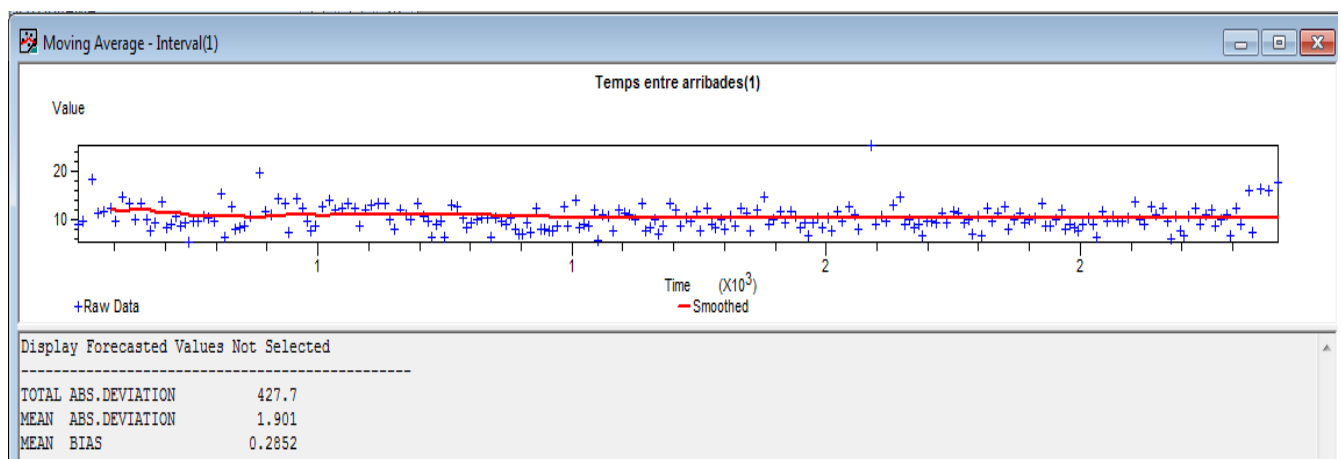
Hora d'inici (**Begin Time**) si s'especifica, aleshores el càlcul d'una mitjana de dades de allisat s'iniciarà a l'hora especificada. S'ha de tenir en compte que degut als canvis en l'hora d'inici, és probable que canviïn els valors previstos, ja que el valor previst en qualsevol moment es calcula utilitzant alguns o tots els preus cessió de punts de dades.

Hora de finalització (**End Time**) es pot especificar com un límit en el rang de temps perquè els valors pronosticats es calculen i es mostren, el seu ús no afecta els valors pronosticats reals.

L'eix X (**X-Axis Label = Time**) estarà per defecte basat en temps i l'eix Y (**Y-Axis Label = Value**) mostra per defecte els valors pronosticats.

**Nota:** Les dades del fitxer **.dat** han de ser filtrades (utilitzant la opció del menú **Output Analyzer : Analyze -> Batch -> Truncate Observations option**), per mostrar les observacions en increments de temps fix abans de calcular una mitjana mòbil exponencial suavitzada, la qual cosa és necessari si es vol fer la previsió acumulada.

A continuació fem clic a OK i apareixerà el següent gràfic mostrant el promig acumulat generat:



La gràfica anterior mostra que en el temps acumulat (2.400 minuts), el promig del temps que triga una entitat en anar des del procés A, al procés D arriba a ser un valor estable a partir de la meitat del temps.

En canvi, si el mateix tingués una desviació exponencial indicadora de la no estabilització. S'hauria de córrer una nova rèplica augmentant el temps en el simulador ARENA, amb la qual cosa obtindríem un nou fitxer de dades **Interval.dat**, fins obtenir una estabilització de les dades amb el mètode especificat.

#### 4.7.2.4. Comparació de mitjanes.

En aquest apartat s'estudiarà si no hi ha diferència estadística entre dos mètodes o models en termes de la mitjana que donen com a sortida.

S'utilitzarà com a model base per aquest exemple l' utilitzat en l'apartat 4.6.2.2, el qual està guardat dins del fitxer anomenat **Exemple\_Generacio\_Analisi\_Reports\_Entitat.doe**, el qual recollia les estadístiques per a una entitat.

Assumirem que per aquest apartat, es vol determinar si no hi ha diferències estadístiques entre dos mètodes en termes de la variable de resposta capturada pel mòdul **RECORD** i anomenada **Estadístiques de la Entitat en D**.

Es proven dos mètodes per al procés C, el primer utilitza la distribució TRIANG(5,10,15) per defecte de l'exemple i el segon utilitza una distribució de temps UNIF(4,14).

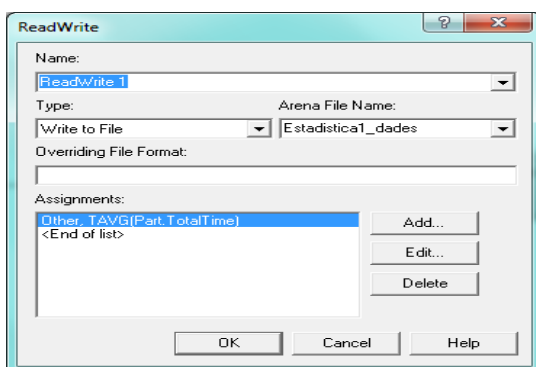
Al model original s'ha d'afegir un mòdul STATISTIC per tal d'obtenir el fitxer de dades amb el resultat de l'expressió **TAVG(Part.TotalTime)** que analitzarem amb el **Output Analyzer**.

La figura següent mostra les dades que haurem d'incloure en el mateix:

Statistic - Advanced Process					
	Name	Type	Expression	Report Label	Output File
1	Estadistica_1	Time-Persistent	TAVG(Part.TotalTime)	Estadistica_1	Estadistica_1.dat

Double-click here to add a new row.

Per lo tant es crearà un primer fitxer **Estadistica\_1.dat** a partir del model original dins d'un fitxer anomenat **Exemple\_Generacio\_Analisi\_Reports\_Entitat\_Modif\_1.doe** i un segon fitxer **Exemple\_Generacio\_Analisi\_Reports\_Entitat\_Modif\_2.doe** a partir del model original dins d'un fitxer anomenat **Estadistica\_2.dat**.

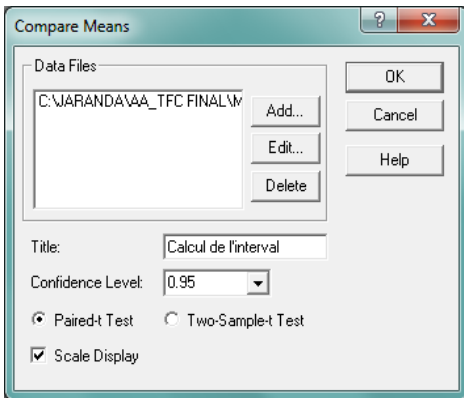


Les dades d'aquests fitxers no es poden consultar directament doncs estan codificades. Aleshores dins de cadascun d'aquest models també afegirem un mòdul **ReadWrite** amb les dades de la figura de l'esquerra per a guardar les dades de la mateixa expressió dins d'un fitxer de text anomenat **Estadistica1\_dades.dat** per al primer model i **Estadistica2\_dades.dat** per al segon model.

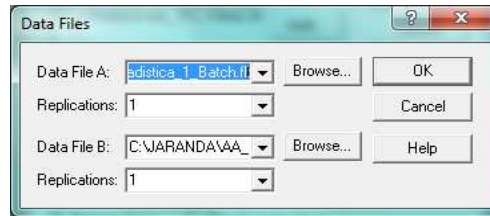
**Nota:** Les dades de cada fitxer **\*.dat** abans de ser utilitzades pel **Output Analyzer** han de ser agrupades (utilitzant la opció del menú **Output Analyzer : Analyze -> Batch -> Truncate Observations option**), per mostrar les observacions agrupades en blocs de tamany fix i poder realitzar la comparació de mitjanes. Com a resultat d'aquesta agrupació es crearan dos nous fitxers de dades els quals seran cridats

per l'analitzador **Output Analyzer**, **Estadistica\_1\_Batch.flt** i **Estadistica\_2\_Batch.flt**.

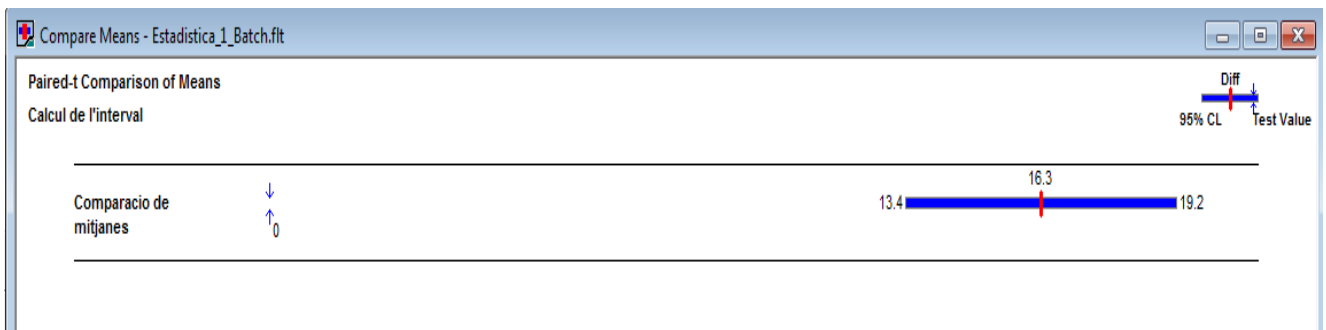
Per a realitzar veure la gràfica comparativa de mitjanes obrim el simulador **Output Analyzer** i mitjançant l'opció de menú **Analyze -> Compare Means...** apareixerà el següent formulari:



Dins del qual afegirem mitjançant l'opció Add... i el formulari que s'obre els dos fitxers **.flt** anteriors.



Una vegada s'han carregat els dos fitxers apareixerà la gràfica amb l' interval de confiança per a



l'anàlisi:

En l'anterior gràfica s'aprecia que el zero no es troba inclòs dins l' interval de confiança per a les diferències de mitjanes, per lo tant es pot concloure gràficament que ambdues son diferents.

A continuació es mostra també la taula resum analítica que s'obté de la prova **Paired-T Means**

Paired-T Means Comparison : Calcul de l'interval						
IDENTIFIER	ESTD. MEAN DIFFERENCE	STANDARD DEVIATION	0.950 C.I. HALF-WIDTH	MINIMUM VALUE	MAXIMUM VALUE	NUMBER OF OBS
Comparacio de mitjanes	16.3	9.66	2.87	12.5	19.2	46
				11.1	21.1	46
REJECT H0 => MEANS ARE NOT EQUAL AT 0.05 LEVEL						

### Comparison.

La diferència mitjana entre les dues poblacions és de 16,3 amb una desviació estàndard de 9,66.

Aleshores de les dades de la mateixa taula s'extreu el resultat mitjançant el qual rebutgem la hipòtesi 0

-> Les mitjanes no són iguals amb un nivell de confiança del 95 %.

#### 4.7.2.5. Anàlisi de variància per a un sol factor (Anova) .

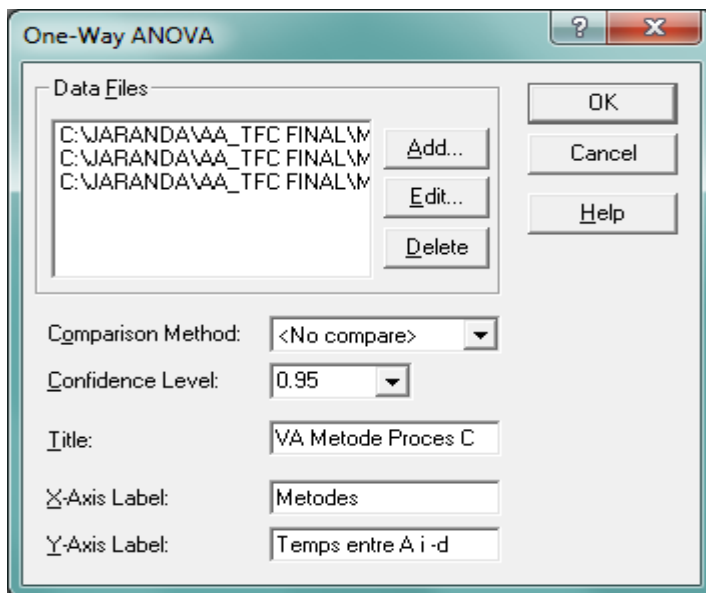
Suposem que en el model base per aquest exemple el de l'apartat 4.6.2.2, el qual està guardat dins del fitxer anomenat **Exemple\_Generacio\_Analisi\_Reports\_Entitat.doe**, es te un tercer mètode per al Procés C i el mateix segueix una distribució exponencial del mitjana 8 minuts.

En aquest apartat es desitja provar si els diferents mètodes o tractaments tenen efectes sobre la variable de resposta, la qual analitza el temps entre A i D.

S'ha de modificar el model amb la finalitat d'implementar el nou mètode, executar-lo una altra vegada mitjançant l'emmagatzament de les dades dins d'un nou arxiu **\*.dat**, per exemple, **Estadistica3\_dades.dat** a partir del model original dins d'un nou fitxer de dades codificat al que anomenarem de nou com a **Exemple\_Generacio\_Analisi\_Reports\_Entitat\_Modif\_3.doe**.

Les dades d'aquests fitxers no es poden consultar directament doncs estan codificades. Aleshores dins de cadascun d'aquest models també afegirem un mòdul **ReadWrite** amb les dades de la figura de l'esquerra per a guardar les dades de la mateixa expressió dins d'un fitxer de text anomenat **Estadistica3\_dades.dat**.

Per a veure la gràfica per a l'anàlisi de la variància obrirem el simulador **Output Analyzer** i mitjançant l'opció de menú **Analyze -> One-Way ANOVA...** apareixerà el següent formulari:

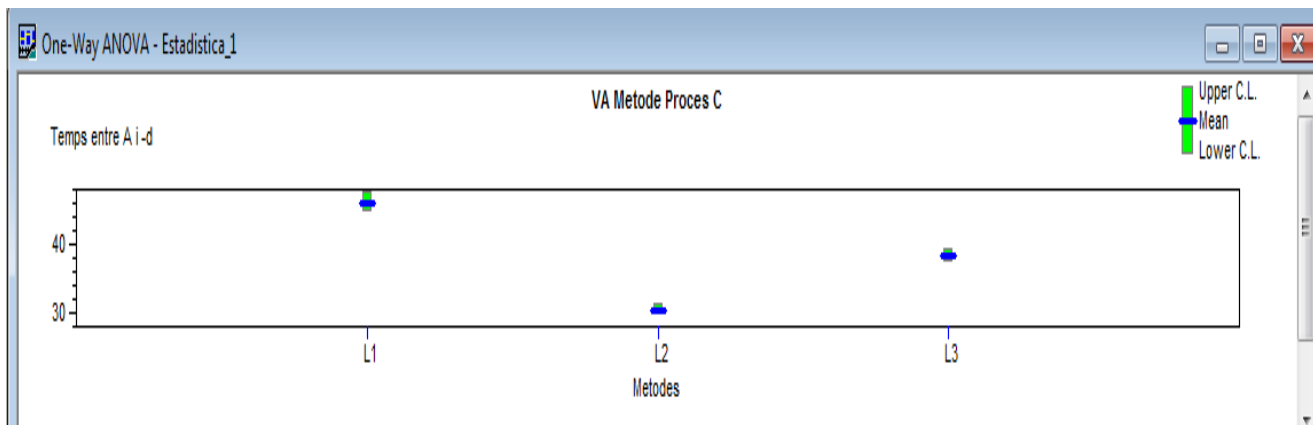


Dins del mateix mitjançant el botó **Add...** afegirem els tres fitxers **\*.dat**, dos generats en l'apartat anterior i el d'aquest anomenats **EstadisticaX\_dades.dat**, amb els estadístics generats dins del model comparats.

**Nota:** Si es vol es escollir mitjançant l'opció **Comparison Method** un anàlisi de diferències de mitjanes mitjançant la prova de Tukey, el nombre d'observacions ha de ser igual. Les dades del fitxer **Estadistica3\_dades.dat** abans de ser utilitzades per al **Output Analyzer** han

de ser truncades o agrupades (utilitzant la opció del menú **Output Analyzer : Analyze -> Batch -> Truncate Observations option**), per mostrar les observacions agrupades en blocs de tamany fix i poder realitzar la comparació de mitjanes. Com a resultat d'aquesta agrupació es crearà un nou fitxer de dades amb extensió **\*.flt**, el qual serà carregat pel **Output Analyzer**.

Una vegada es prem el botó OK apareixerà la gràfica amb l'interval de confiança per a l'anàlisi:



En l'anterior gràfica s'aprecia l'interval de confiança per a les diferències de mitjanes i s'observa que les mateixes no són iguals.

A continuació es mostra també la taula resum analítica que s'obté de la prova **One-Way ANOVA Table**.

One-Way ANOVA Table : VA Metode Proces C				
L1 -	Estadística_1			
L2 -	Estadística_2			
L3 -	Estadística_3			
SOURCE OF VARIATION	SUM SQUARES	DF	MEAN SQUARES	F-EXP
BETWEEN TREATMENTS	2.933e+004	2	1.467e+004	186.744
ERROR (W.TREATMENTS)	5.505e+004	701	78.5	
TOTAL	8.438e+004	703	F-CRIT	3.008
REJECT H0 => NOT ALL MEANS ARE EQUAL AT 0.05 LEVEL				

Les dades donen directament l'estadígraf de comparació amb  $F\text{-EXP} = \text{MSE} / \text{MSD} = (14670/78,5) = 186,744$ .

Mentre que de la taula de la funció F, tabulada per G. W. Snedecor, s'obté que el valor crític  $F\text{-CRIT}_{0,95,2,701} = 3,008$ .

Aleshores donat que  $F\text{-EXP} > F\text{-CRIT}_{0,95,2,701}$ , **Output Analyzer** conclou l'anàlisi de la variància per a un sol factor, rebutjant la hipòtesi  $H_0$  -> Les mitjanes no són iguals amb un nivell de confiança del 95 %.

### 4.7.3. INTERFICIE ARENA-EXCEL.

En moltes ocasions es requereix fer un estudi o anàlisi més profund dels resultats de la simulació d'un model creat amb ARENA. Es cert que aquest programa ja ens dóna l'estadístic de cada variable que entra en joc amb el model simulat, però no compte amb un eina especialitzada amb la que es pugui fer, per exemple, una comparació entre els resultats de la simulació d'un model respecte una altra.

Per desenvolupar aquests tipus d'estudis es necessari recórrer a d'altres eines comercials, com Microsoft Office i concretament el seu paquet de càlcul Excel.

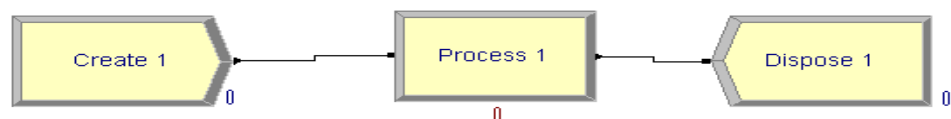
A continuació s'analitzen, pas a pas, com s'han d'executar els models que s'han creat en ARENA des d'una fulla de càlcul com Microsoft Excel i com s'han de capturar els resultats de la simulació per a fer l'estudi desitjat sobre aquesta fulla de càlcul.

Per exemple, si estudiem un sistema de cues MMK s'identifiquen tres tipus de paràmetres. El primer d'ells és la taxa d'arribades ( $\lambda$ ) d'entitats al sistema, el qual representa el nombre d'unitats per unitat de temps que arriben al sistema. El segon és la taxa de servei ( $\mu$ ) o el nombre d'entitats per unitat de temps que atén un servidor i el qual és el mateix per a tots els servidors. El tercer paràmetre és el nombre de servidors ( $k$ ) de les que disposa el sistema per atendre a les entitats. Dins d'aquest exemple s'han escollit tres servidors,  $k=3$ , és a dir, es pretén modelar un sistema MM3.

Per tal de simular amb ARENA i realitzar l'estudi amb Excel, seguirem els següents passos:

#### 1. CREACIO DEL MODEL EN ARENA.

- Inicialment s'ha d'obrir un nou model en ARENA.
- S'insereixen tres mòduls del panell de processos bàsics: Create, Process i Dispose i a continuació es connecten. Aquest procediment es mostra a continuació:

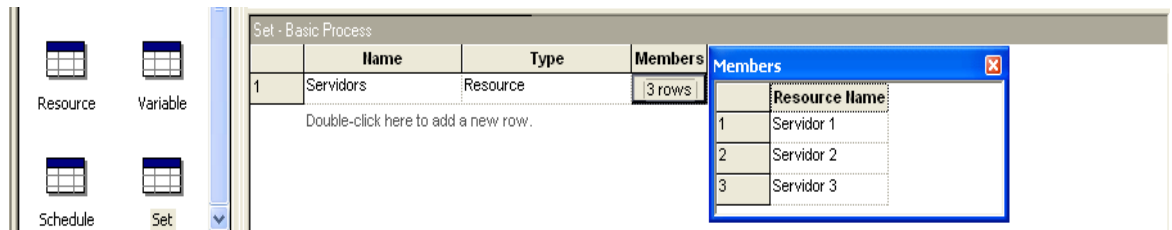


- Es necessiten tres servidors dins d'aquest model. Per a desenvolupar el model d'aquests servidors es creen tres recursos (cada servidor és un recurs utilitzable del sistema).

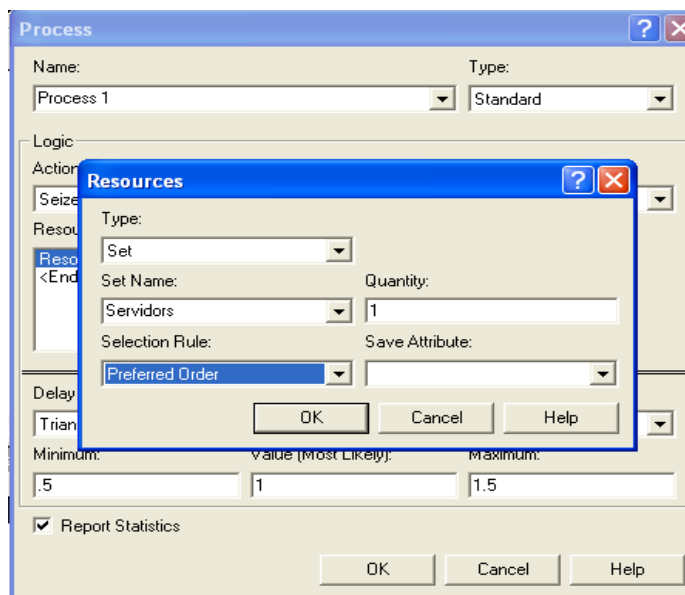
A continuació es selecciona un mòdul Resource i s'addicionen tres recursos anomenats, Servidor 1, Servidor2, Servidor3, tal i com es mostra a la següent figura:

Resource - Basic Process									
	Name	Type	Capacity	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures	Report Statistics
1	Servidor 1	Fixed Capacity	1	0.0	0.0	0.0		0 rows	<input checked="" type="checkbox"/>
2	Servidor 2	Fixed Capacity	1	0.0	0.0	0.0		0 rows	<input checked="" type="checkbox"/>
3	Servidor 3	Fixed Capacity	1	0.0	0.0	0.0		0 rows	<input checked="" type="checkbox"/>

Mitjançant el mòdul Set s'afegeixen un conjunt de recursos anomenat Servidors. Posteriorment dins de ell afegirem els recursos Servidor 1, Servidor2, Servidor3 com a membres (Members) del conjunt Set, tal i com es mostra a la següent figura:



- A continuació dins de la lògica del mòdul Process s'especifica que aquest treballi amb el conjunt de Servidors. Es farà doble clic en el bloc Process i s'especificarà com acció (Action) Seize Delay Release.
- Mitjançant el botó Add s'especifica que les quantitats que arribin a aquest bloc facin us d'un dels recursos (Quantity=1) que pertanyen al conjunt de servidors (Type=Set, Set Name=Servidors) si es troba disponible. Posteriorment escollirem com a regla de selecció la Preferred Order, per tal que s'esculli el primer servidor que estigui lliure.



- Per tal de poder manipular cadascun dels blocs mitjançant codi, cada bloc té un identificador únic (Tag). Un mòdul es modifica quan fem clic dret sobre ell i seleccionant la opció Properties. ARENA mostrarà doncs el Tag actual del mòdul seleccionat i permet modificar-lo. Per a l'exemple en estudi, col·locarem Entitats al TAG del mòdul Create i Proces al Tag del mòdul Process.
- Guardarem el model amb el nom MM3.doc

## 2. CREACIO DE LA FULLA DE CALCUL EN EXCEL.

- Inicialment s'ha d'obrir un nou llibre Excel.
- En una de les fulles es copia exactament el següent format:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1			PARAMETRES										
2													
3		TAXA D'ARRIBADES		1 entitats/minut									
4		TAXA DE SERVEI		2 entitats/minut									
5													
6		RESULTATS DE LA SIMULACIO											
7													
8		LONGITUD PROMIG		entitats									
9		DE LA CUA											
10													
11		TEMPS PROMIG		minuts									
12		D'ESPERA A CUA											
13													
14		TEMPS PROMIG		minuts									
15		DE LES ENTITATS EN											
16		EL SISTEMA											
17													

- Es guarda el llibre amb el nom MM3.xls. Per al correcte funcionament del cas d'estudi, els arxius de la fulla de càlcul i el model en ARENA s'han de guardar dins de la mateixa carpeta.

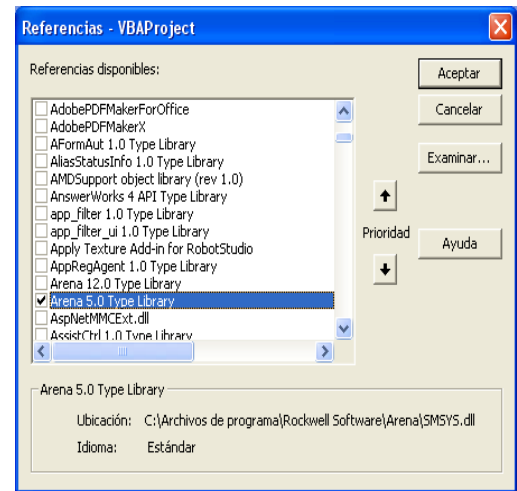
## 3. ENLLAÇ ARENA - EXCEL :

Amb aquest propòsit s'utilitza Visual Basic per aplicacions, el qual es troba dins de totes les aplicacions de Microsoft Office.

- S'obre l'editor de Visual Basic mitjançant el menú principal de Microsoft Excel. Eines -> Macro -> Editor de Visual Basic o la combinació de tecles Alt-F11.
- Quan aquest editor es troba obert, es selecciona l'opció Mòdul dins del menú Insertar. La inserció d'un mòdul és necessària per a la creació de procediments definits per l'usuari.
- Dins del mòdul és precís crear un procediment anomenat Connexio, que serà l'encarregat d'executar el model que s'ha creat en ARENA per a capturar els resultats de la simulació.



- Abans de començar a escriure el codi, és precís assegurar-se de tenir disponible la llibreria d'objectes del paquet ARENA, per a poder manipular els models creats amb aquesta eina. Per lo tant, és necessari afegir una referència a la llibreria mencionada, per a fer-ho obrim el menú Herramientas i es selecciona l'opció Referencias. S'ha de buscar la llibreria Arena Type Library i seleccionar-la.



Aquesta llibreria utilitza les classes Arena.Application, Arena.Model i Arena.Module per tal d'executar l'aplicació ARENA.exe, obrir un model creat per a la simulació i modificar els paràmetres dels mòduls que aquest incorpora respectivament.

- Quan es finalitza la simulació d'un model, ARENA genera una arxiu de base de dades Access amb la informació estadística del model. Per tal de consultar-lo també des de Excel haurem d'afegir també una referència a la llibreria d' Objectes d'Accés a Dades (DAO) de Microsoft anomenada Microsoft DAO 3.51 Object Library.

Aquesta llibreria utilitza les classes DAO.Database i DAO.Recordset per tal d'accedir a la base de dades i recuperar els valors dels registres emmagatzemats en les seves taules.

Amb aquestes dues referències ja s'està preparat per a escriure el codi necessari per a la creació de l'enllaç ARENA - EXCEL.

#### 4. CODI CREACIO MACRO D'ENLLAÇ EXCEL EN VISUAL BASIC :

A continuació es mostra tot el codi del procediment Connexio i dins del mateix s'explica entre cometes la funció de cadascuna de les seves parts i aquest text no s'ha d'incorporar al procediment de Visual Basic d' Excel.

##### **Sub Connexio()**

*'Les següents tres línies corresponen a la declaració de les variables objecte que pertanyen a la classe ARENA*

**Dim app As ArenaEx.Application**

**Dim model As ArenaEx.Model**

**Dim modul As ArenaEx.Module**

*'La següent línia es refereix a la declaració de la variable objecte que serveix per a manipular la fulla de càlcul de la qual provenen els paràmetres d'entrada i els resultats de sortida per ARENA.*

**Dim fulla As Excel.Worksheet**

*'Les següents dues línies corresponen a la declaració de les variables objecte que pertanyen a la classe DAO per a l'accés a la base de dades dins la qual ARENA emmagatzema els resultats de la simulació.*

**Dim bd As DAO.Database****Dim taula As DAO.Recordset**

*'Declaracions de variable que s'utilitzaran posteriorment.*

**Dim ruta, criteri As String****Dim id As Long****Dim x, y As Single**

*'Es fa que la variable fulla faci referència a la fulla de càlcul dins la qual es troba el format de treball.*

**Set fulla= Hoja1**

*'A continuació s'emmagatzemen dins de dues variables els paràmetres que s'han d'enviar al model ARENA. Els valors x e y corresponen a les taxes d'arribada i de sortida d'entitats, respectivament. Per aquest tipus de paràmetres la funció de distribució exponencial és l'apropiada, però el temps entre arribades o el temps entre sortides requereixen de paràmetre, per lo tant es fa el càlcul d'aquest temps com l'invers de les seves respectives taxes.*

**x = fulla.Cells(3,3)**

**y = fulla.Cells(4,3)**

**t1 = 1/Val(x)**

**t2 = 1/Val(y)**

*'En una variable de cadena s'emmagatzema la ruta (path) en la qual es troba el llibre.*

**ruta = ThisWorkbook.Path**

*'A continuació s'executa l'aplicació Arena.exe remotament i s'obre el model creat anteriorment. Com es sap que l'arxiu del model es troba dins de la mateixa carpeta que el llibre Excel, s'utilitza el contingut de la variable ruta per ubicar l'arxiu.*

**Set app = CreateObject("Arena.Application")****Set model = app.Models.Open(ruta + "\MM3.doe")**

*'ARENA assigna a cada mòdul del model un nombre únic que identifica a cada bloc. Mitjançant el mètode FIND de la classe Arena.Model.Modules es pot recuperar aquest nombre del mòdul, especificant el Tag que l'identifica.*

*Es busca el mòdul Create mitjançant el seu Tag anomenat Entitats.*

**id = model.Modules.Find(smFindTag, "Entitats")**

*'Es fa que la variable modul reverenciï al mòdul Create del model, fent ús del nombre identificador d'aquest mòdul.*

**Set modul = model.Modules(id)**

*'Mitjançant la propietat Data(operand) de la classe Arena.Module es poden modificar els paràmetres de cada mòdul. Però s'ha de tenir en compte que el paràmetre Operand de la propietat Data no correspon, moltes vegades amb el valor del paràmetre que apareix dins la caixa de diàleg de cada*

mòdul dins ARENA. Per exemple, per al paràmetre *Type* del mòdul *Create*, el nom de l'operand respectiu és *InterarrivalType*. Dins la carpeta on es troba instal·lat ARENA es troben uns arxius de text anomenats *BasicProcess.txt* i *AdvancedProcess.txt*, dins dels quals s'especifiquen els noms dels operands a utilitzar dins de la propietat *Data* per a tots els paràmetres de cadascun dels mòduls que pertanyen a cadascun dels panells ARENA.

*Es modifica el paràmetre Type del mòdul Create.*

**modul.Data("Interarrival Type") = "Expression"**

*'A continuació es modifica el paràmetre Expression del mòdul Create. Mitjançant la concatenació es pretén introduir una expressió vàlida per ARENA com EXPO(0.33). Es precis tenien en compte prèviament que la distribució exponencial és l'apropiada.*

**modul.Data("Expression") = "EXPO("& Str(t1) &")"**

*'S'especifica el paràmetre Units del mòdul Create per a determinar les unitats de temps per als temps entre arribades successives.*

**modul.Data("Units") = "Minutes"**

*'Es realitza el mateix procediment amb el mòdul Process, al qual se li havia assignat prèviament el Tag Proces.*

**id = model.Modules.Find(smFindTag,"Proces")**

**Set modul = model.Modules(id)**

**modul.Data("DelayType") = "Expression"**

**modul.Data("Expression") = "EXPO("& Str(t2) &")"**

**modul.Data("Units") = "Minutes"**

*'Es precis assegurar que es genera l'arxiu base de dades amb les estadístiques del model un cop finalitzada l'execució de la simulació.*

**model.DisableReportDatabase = False**

*'Per evitar que surti al final de cada rèplica un missatge que preguntí si es vol veure el report ARENA, es col·loca la següent línia:*

**model.DisplayDefaultReport = smNeverDisplay**

*'Amb les següents línies s'especifica la unitat de temps base per al model (amb la qual es registrarà el rellotge de la simulació i amb la qual es donaran els resultats dins de la base de dades), el nombre de rèpliques que es volen, la longitud de cada rèplica, les unitats de temps amb les quals es mesuren aquestes rèpliques i la velocitat de simulació.*

**model.BaseTimeUnits = smMinutes**

**model.NumberOfReplications = 1**

**model.ReplicationLength = 1**

**model.ReplicationLengthTimeUnits = smHours**

**model.RunSpeed = 10**

*'S'inicia l'execució del model i es fa que Visual Basic no executi cap línia de codi fins que el model no acabi la seva execució. Una vegada s'acabi la rèplica, se li envia una ordre End al model per tal que també es posi fi a la simulació.*

**model.Go smGoWait**

**model.End**

*'Es guarden els canvis realitzats i es tanca el model.*

**model.Save**

**model.Close**

*'ARENA un cop s'ha finalitzat la simulació genera la base de dades de suport dins dl mateix directori en el qual es troba l'arxiu del model. En aquest cas en particular també és el mateix directori on es troba el llibre Excel i el mateix es troba emmagatzemant dins de la variable ruta.*

*A continuació es realitza la connexió amb la base de dades.*

**Set bd=Workspaces(0).OpenDatabase(ruta + "\mm3.mdb")**

*'Dins dels objectes de la base de dades hi ha una consulta o vista anomenada StatsAndOutputQry, la mateixa conté totes les estadístiques col·leccionades per ARENA durant la simulació. Els camps principals d'aquesta vista son: Value (valor), AvgObs (promig observat), MaxObs (valor màxim observat), MinObs (valor mínim observat), SourceProcess.Name (objecte-entitat, recurs, cua, etc. sobre el que es va generar l'estadística) i SourceDataType.Name (propietat observada). Per a una millor comprensió del que s'ha exposat és recomanable que s'obri la pròpia base de dades amb Microsoft Access, amb el propòsit d'examinar els camps i es decideixi quina parella de valors dels camps SourceProcess.Name i SourceDataType.Name es la que correspon a l'estadística que es desitja recuperar.*

*En la següent línia es mostra la cadena SQL amb la que capturarem les dades emmagatzemades dins de la base de dades de la vista anomenada StatsAndOutputQry.*

**Sql="Select Value as valor, AvgObs as promig, MaxObs as maxim," + \_**

**"MinObs as minim, SourceProcess.Name as objecte," + \_**

**"SourceDataType.Name as propietat " + \_**

**"From StatsAndOutputQry"**

*'Es fa la consulta a la base de dades i el seu resultat queda emmagatzemat dins de la variable taula.*

**Set taula=bd.OpenRecordset(Sql, dbOpenDynaset)**

*'Es verifica que s'hagin recuperat registres.*

**If Not taula.EOF Then**

*'Es fa un moviment cap a l'inici del conjunt de registres recuperats.*

**taula.Movefirst**

*‘Mitjançant una cadena de condició SQL s’especifiquen els valors de la parella objecte-propietat (SourceProcess.Name- ‘SourceDataType.Name) de l’estadística que es desitja recuperar. En aquest cas es tracta del nombre promig de persones a la cua i per això l’objecte sobre el qual es va generar l’estadística va ser Process 1.Queue, la propietat Number Waiting i la dada que es desitja és el valor promig.*

**criteri = “objecte=’Process 1.Queue’ and propietat=’Number Waiting”**

**taula.FindFirst criteri**

*‘A continuació es guarda l’estadística dins de la fulla de càlcul específicament dins de la celda que s’ha destinat per aquest fi.*

**fulla.Cells(8,3) = Str(taula.Fields(“promig”))**

*‘Com es va mencionar anteriorment, el procés de selecció dels valors adequats per als camps objecte i propietat només és satisfactori si es consulta directament la base de dades i s’analitzen els valors que ARENA a emmagatzemat en ella.*

*El procediment per a recuperar els valors de cada estadística és el mateix.*

**taula.Movefirst**

*‘Es desitja recuperar el temps d’espera en cua.*

**criteri = “objecte=’Process 1.Queue’ and propietat=’Waiting Time”**

**taula.FindFirst criteri**

**fulla.Cells(11,3) = Str(taula.Fields(“promig”))**

**taula.Movefirst**

*‘Es desitja recuperar el temps total de les entitats al sistema.*

**criteri = “objecte=’ENTITY 1’ and propietat=’Total Time”**

**taula.FindFirst criteri**

**fulla.Cells(14,3) = Str(taula.Fields(“promig”))**

**End If**

*‘Finalment es finalitza el procediment Connexio (), tancant el conjunt de registres i la base de dades.*

**Taula.close**

**bd.close**

**End Sub**

El procediment no canvia en el cas que el model contingui mòduls que utilitzin comptadors (tallies) o que el càlcul d’algunes estadístiques addicionals s’hagi especificat (mitjançant el mòdul Statistic, per exemple). Aquestes estadístiques també queden emmagatzemades dins la base de dades.

L'únic pas previ a la captura d'aquests resultats des d'una fulla de càlcul en EXCEL és obrir la base de dades e identificar sota quines parelles, SourceProcessName i SourceDataType, s'han emmagatzemat els resultats observats durant la simulació.

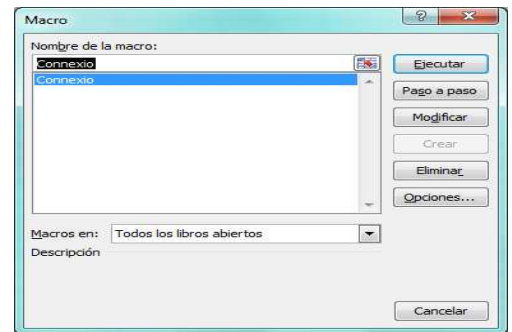
Aquest procés d'identificació és bastant senzill, donat que ARENA utilitza els mateixos nombres de mòduls, comptadors, tallies i estadístiques que l'usuari especifica en el model.

Una vegada s'acaba d'escriure tot el codi, es guarden els canvis realitzats, es tanca la finestra de l'editor de Visual Basic i es retorna a la finestra tradicional de EXCEL. la mateixa carpeta.

### 5. EXECUCIO DE LA INTERFICIE ARENA-EXCEL :

Finalment ja es pot posar en pràctica el funcionament de l'enllaç ARENA-EXCEL. Lo primer serà subministrar a la fulla de càlcul les taxes d'arribades d'entitats al sistema i la taxa de servei per a cadascun dels servidors i a continuació :

- Es selecciona l'opció Macros... del menú Herramientas i apareixerà una finestra amb el llistat dels procediments que s'han creat (en aquest cas un de sol). Es fa clic sobre el procediment Connexio i posteriorment es pressiona el botó Ejecutar.



- En primer plà EXCEL obrirà el programa ARENA.exe remotament i executarà la simulació anomenada MM3.doe.

- Una vegada acabada l'execució del model en ARENA, veurem la fulla de càlcul omplerta amb les dades observades durant la simulació del model i obtindrem el següent resultat:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1			PARAMETRES											
2														
3		TAXA D'ARRIBADES		1 entitats/minut										
4		TAXA DE SERVEI		2 entitats/minut										
5														
6		RESULTATS DE LA SIMULACIO												
7														
8		LONGITUD PROMIG DE LA CUA		0,002511 entitats										
9														
10														
11		TEMPS PROMIG D'ESPERA A CUA		0,003074 minuts										
12														
13														
14		TEMPS PROMIG DE LES ENTITATS EN EL SISTEMA		0,565465 minuts										
15														
16														
17														

#### 4.7.4. ARENA VISUAL BASIC FOR APPLICATIONS (VBA).

L'entorn de desenvolupament integrat de Visual Basic For Applications (VBA) està disponible per a totes les aplicacions Microsoft Office i en moltes altres variades aplicacions com ARENA, dins del mateix paquet i es crida mitjançant la combinació de tecles Alt + F11.

Mitjançant aquesta potent eina els desenvolupadors d'aplicacions avançats, poden extreure tot el potencial existent dins d'aquests paquets tancats dins d'un entorn pensat per usuaris amb un nivell inferior.

Així doncs, la seva utilització es preveu complexa i pel seu aprofitament, cada paquet que incorpora VBA ja predisposa una sèrie de rutines exclusives per a cada sistema. Les mateixes permeten facilitar la integració d'un llenguatge de programació a baix nivell com Visual Basic amb les interfícies d'usuari de cadascun. Per al constructor de models de simulació ARENA, aquestes rutines es poden cridar des de l'aplicació quan el model es carrega, s'executa o finalitza i també mentre les entitats flueixen pels diferents mòduls de l'entorn simulat.

VBA també inclou suport per a Microsoft Forms, per tal de crear formularis personalitzats i controls ActiveX, que faciliten la creació ràpida de noves interfícies d'usuari dins del propi entorn on es troba integrat.

#### RUTINES VBA PER AL MODEL LOGIC ARENA.

La principal interacció entre ARENA i VBA és via el conjunt de rutines i el model lògic ARENA. Aquestes s'executen automàticament quan s'activen els events predeterminats i depenen de la seva aplicació inclús poden intercanviar dades amb el model lògic ARENA.

Dins de l'apartat pràctic d'aquest projecte es desenvoluparan diferents exercicis que complementaran aquesta secció. Del total de 27 tipus de rutines disponibles dins l'entorn VBA per ARENA en destacarem les cinc considerades com d'ús bàsic per tal d'introduir i obrir tot projecte de simulació al món exterior i mostrar una petita part del potencial que dona VBA a qualsevol entorn on s'ha integrat.

1. Subrutina **RunBeginSimulation**: S'executa tant punt el model comença la seva execució i pot ser utilitzada per exemple per tal de llegir les dades d'entrada, generar un informe amb les dades d'entrada o inicialitzar automàticament els valors de les variables i els paràmetres del model.
2. Subrutina **RunBeginReplication**: S'executa cada vegada que comença una rèplica de la simulació dins del model. Pot ser utilitzada per exemple per a fer reset el comptadors de cada

nova rèplica, netejar l'estat del sistema, obrir fitxers específics per a cada rèplica o programar diferents events discrets.

3. Subrutina **RunEndReplication**: S'executa cada vegada que finalitza una rèplica de la simulació dins del model. Pot ser utilitzada per exemple per mostrar els resultats de la rèplica, netejar l'estat del sistema o per tancar fitxers específics per aquella rèplica.
4. Subrutina **RunEndSimulation**: S'executa tant punt el model acaba la seva execució i pot ser utilitzada per exemple per generar un informe amb les dades de sortida o per tancar fulles de treball actives.
5. Subrutina **VBA\_Block\_n\_Fire**: S'executa tant punt entra una entitat dins del nou bloc VBA creat. Dins d'aquesta rutina es guarda la lògica per a un tipus de bloc definit completament per l'usuari. Amb la qual cosa la capacitat d'expansió de l'entorn ARENA és infinita, doncs depenent del nivell de l'usuari es poden complementar els blocs que ARENA inclou per defecte amb aquells creats segons les noves necessitat no previstes pel fabricant de l'aplicació.

Per identificar quina rutina VBA s'executarà quan una entitat entra a un bloc VBA concret, s'associa el número de bloc al de la rutina mitjançant la variable n, la qual indicarà el nombre que ocupa el bloc VBA específic dins d'un ordre dins del total de blocs VBA creats al model.

La llista de rutines VBA disponible dins l'entorn ARENA, la seva descripció, aplicació i paràmetres es mostren dins de la llista següent:

Rutines VBA		Paràmetres				
VBA Model Logic	Descripció / Aplicació	Tipus	1	2	3	4
DocumentOpen	S'executa quan s'obre un fitxer a no ser que es premi la tecla Shift	Subrutina				
DocumentSave	S'executa quan es desa un fitxer	Subrutina				
OnclearStatistics	S'executa quan el model neteja les estadístiques	Subrutina				
OnfileClose	S'executa quan es tanca un fitxer de dades	Subrutina	Files Element Number	File Name		
OnfileRead	S'executa quan es llegeix un fitxer de dades	Subrutina	Files Element Number	File Name	Number Of ITems	readBuffer
OnfileWrite	S'executa quan s'escriu dins d' un fitxer de dades	Subrutina	Files Element Number	File Name	Number Of ITems	readBuffer
Onkeystroke	S'executa quan l'usuari prem una tecla	Subrutina	Key Code			
RealTimeInitialize	Inicialitza el Real Time Process	Long Funtion	processName	Remote Process Name		
RealTimeReceive	Rep un missatge amb el Real Time	Long Funtion	Message			



VBA Model Logic (cont)	Descripció / Aplicació	Tipus	1	2	3	4
RealTimeSend	Envia un missatge amb el Real Time	Long Function	Message			
RealTimeTerminate	Finalitza el Real Time Process	Long Function				
RunBegin	S'executa es prem la icona Run o la tecla F5	Subrutina				
RunBeginReplication	S'executa quan comença l'execució de cada rèplica	Subrutina				
RunBeginSimulation	S'executa quan comença l'execució de la simulació	Subrutina				
RunBreak	S'executa quan s'atura l'execució mitjançant la tecla ESC	Subrutina				
RunEnd	S'executa quan finalitza l'execució de la simulació	Subrutina				
RunEndReplication	S'executa quan finalitza l'execució de cada rèplica	Subrutina				
RunEndSimulation	S'executa quan finalitza l'execució de la simulació	Subrutina				
RunFastForward	S'executa quan es prem la icona Run Fast Forward	Subrutina				
RunPause	S'executa quan es prem la icona Run Pause	Subrutina				
RunRestart	S'executa quan es prem la icona Run Restart	Subrutina				
RunResume	S'executa quan es prem la icona Run Resume	Subrutina				
RunStep	S'executa quan es prem la icona Run Step o es prem F8	Subrutina				
UserContinuousEquation	S'executa en cada pas continu d'integració	Subrutina				
UserFunction	S'executa cada vegada que es troba UF dins del codi d'una expressió	Double Function	entityID	functionID		
UserRule	S'executa cada vegada que es troba UR dins del codi d'una expressió	Double Function	entityID	ruleID		
VBA Block	Descripció / Aplicació	Tipus	1	2	3	4
VBA_Block_n_Fire	S'executa quan entra una entitat dins el bloc VBA n	Subrutina				