

# MODUL PRAKTIKUM -SIMULASI DENGAN SOFTWARE ARENA-



**PENYUSUN:**

**SUWARDI GUNAWAN, S.T., M.T.**



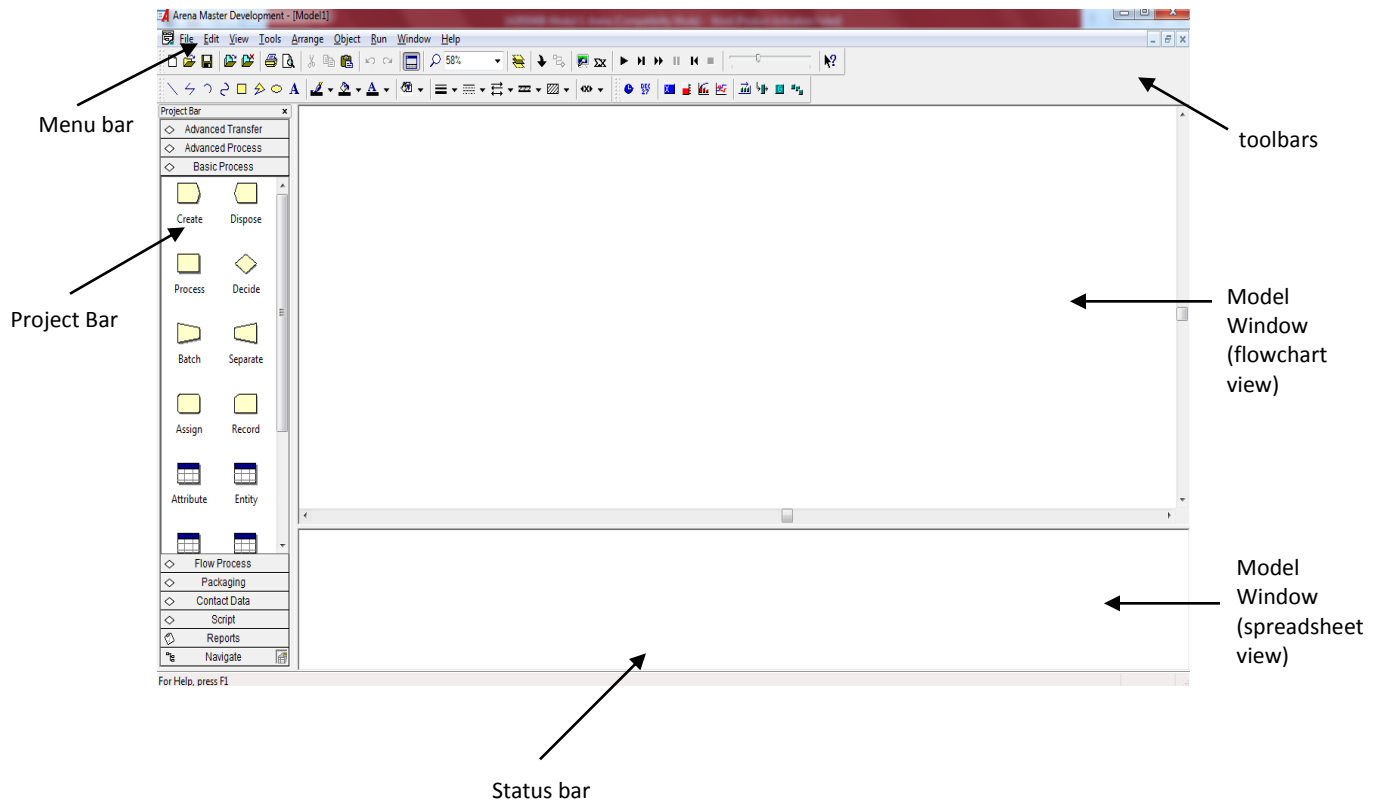
**LABORATORIUM TEKNOLOGI INDUSTRI**

**FAKULTAS TEKNIK**

**UNIVERSITAS MULAWARMAN**

# MODUL 1. PENGENALAN ARENA

Pada menu Start windows pilih program *Rockwell Software* dan kemudian pilih *Arena 14*. Setelah dijalankan maka akan muncul tampilan software *Arena 14* seperti berikut ini.



## a. Menu Bar

Menu bar yang ada di dalam Arena secara umum terdiri dari menu-menu yang identik pada kebanyakan aplikasi untuk windows, seperti menu file (untuk manajemen file pengguna), menu edit, view. Dan tentunya terdapat beberapa menu bar yang disediakan Arena 14 untuk membantu pengerjaan modeling system (seperti tools, arrange, object, dan run).

## b. Project bar

Project bar pada Arena 14 terdiri dari dua hal, yaitu:

- Flowchart module  
Merupakan modul untuk membangun model simulasi dalam arena, terdiri dari modul *basic process*, modul *advance transfer*, dan modul *advance process*.
- Spreadsheet module  
Merupakan modul untuk melihat status dari modul *flowchart* yang digunakan. Status yang ada didapatkan secara otomatis atau diinput secara manual.

## c. Status Bar

Merupakan suatu modul dalam arena yang bertujuan untuk melihat status dari pekerjaan (modul) kita saat ini. Contoh kondisi, *Running* = model simulasi kita sedang dijalankan.

**d. Toolbar**

Merupakan suatu window yang berisi daftar perintah yang sering digunakan dan dipresentasikan dalam bentuk tombol.

**e. Model window (flowchart view)**

Window ini merupakan window induk yang melingkupi seluruh lingkungan kerja Arena. Fungsi utama window ini adalah sebagai tempat *docking* bagi modul-modul yang digunakan.

**f. Model window (spreadsheet view)**

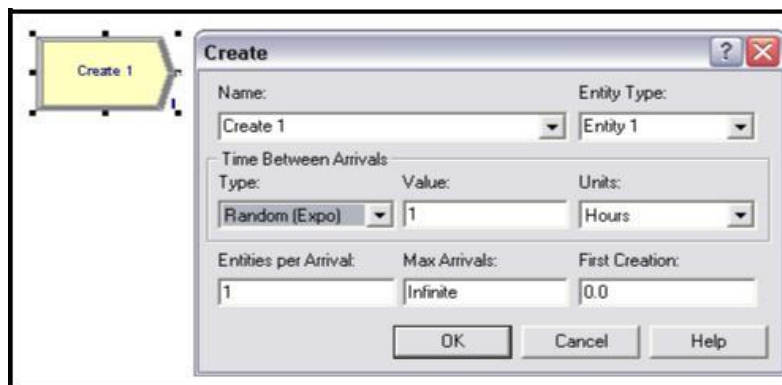
Window ini merupakan window yang digunakan untuk melihat data yang terdapat pada modul-modul yang digunakan pada *flowchart* modul.

## 1. Modul Basic process

Basic process merupakan modul-modul dasar yang digunakan untuk simulasi. Template basic process ini terdiri dari beberapa modul yaitu:

**a. Create**

Modul ini digunakan untuk menggenerate kedatangan entity kedalam simulasi.



Nama : nama modul *create* yang digunakan

Entity type : jenis entity yang digenerate pada simulasi

Type : jenis waktu antar kedatangan entity

- Random (expo)
- Schedule
- Constant
- Espresion

Value : nilai daripada interval kedatangan berdasarkan type yang sudah ditentukan

Units : satuan waktu yang digunakan

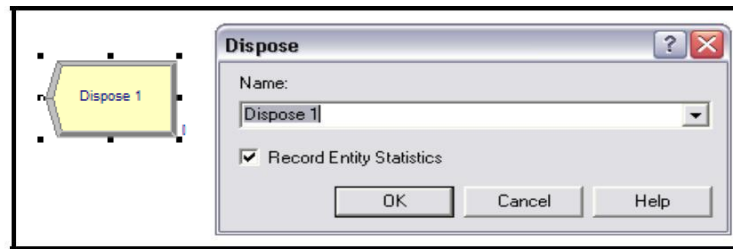
*Entities per arrival* : jumlah kedatangan entity pada setiap kali *generate* dilakukan

*Max arrivals* : jumlah maksimum generasi entity kedalam simulasi

*First creation* : waktu pertama kali *generate entity* kedalam simulasi

## b. Dispose

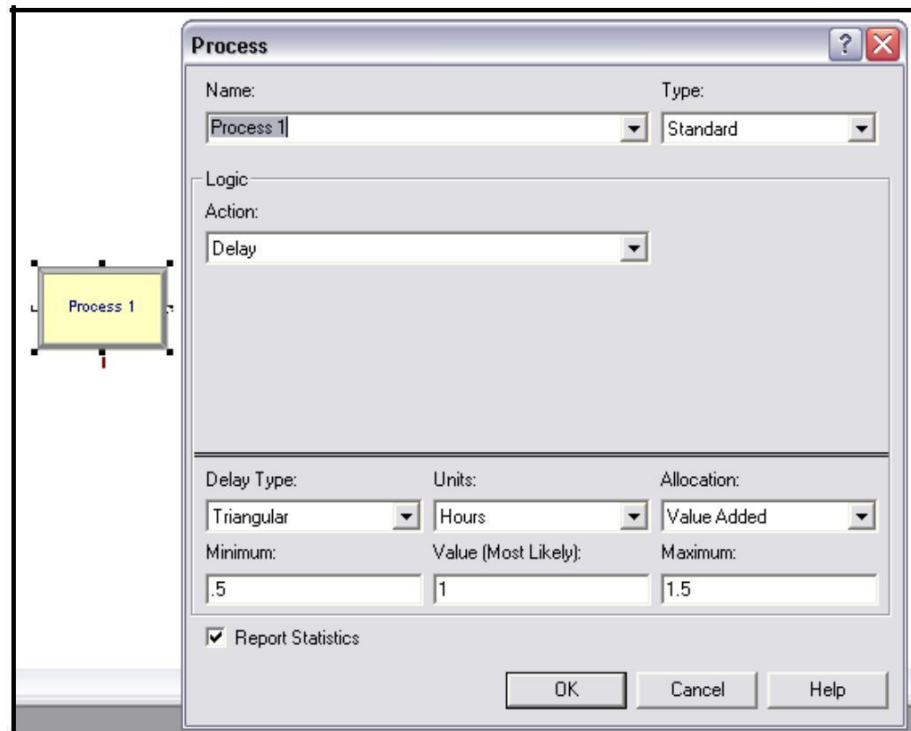
Modul ini digunakan untuk mengeluarkan entity dari sistem



*Record entity statistics* : digunakan untuk mencatat *output standard* daripada ARENA

## c. Process

Modul ini digunakan untuk memproses entity dalam simulasi



*Nama* : nama daripada modul proses yang digunakan

*Type* : tipe dari proses itu sendiri

- Standard → terdiri dari satu proses saja
- Sub model → terdiri dari satu proses atau lebih

*Action* : jenis aktivitas yang dilakukan pada saat modul proses bertipe standard

*Priority* : nilai prioritas dari beberapa jenis proses alternatif

*Resources* : sumber daya yang digunakan dalam melakukan aktivitas proses

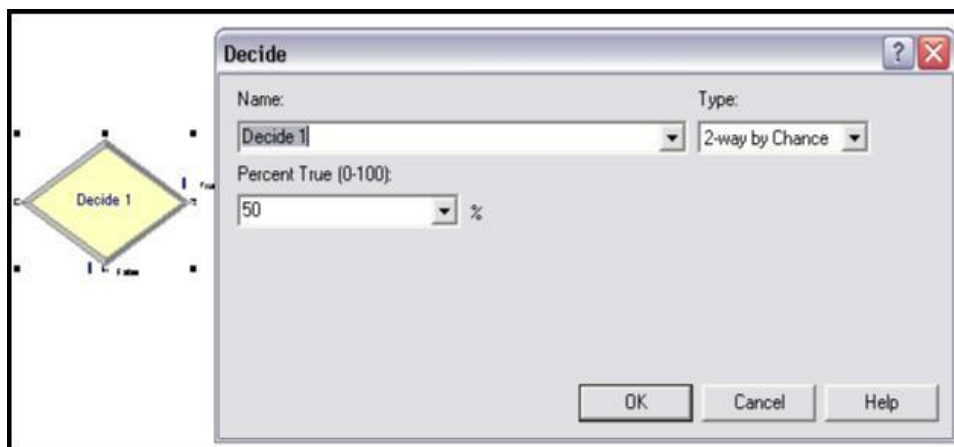
*Delay type* : waktu proses atau bisa juga diasumsikan sebagai waktu *delay* ketika tidak menggunakan *resource* sama sekali

*Allocation* : jenis aktivitas yang terjadi pada modul ini, terdiri dari beberapa jenis antara lain :

- *Value added* → pada proses yang dilakukan terjadi penambahan nilai dari material input menjadi output
- *Non value added* → tidak terjadi proses penambahan nilai dari material input menjadi output (misalkan kegiatan inspeksi)
- *Transfer* → waktu transfer dari satu tempat ke tempat lain
- *Wait* → waktu tunggu sebelum entity melakukan aktivitas berikutnya
- *Other*

#### d. Decide

Modul ini digunakan untuk menentukan keputusan dalam proses, didalamnya termasuk beberapa pilihan untuk membuat keputusan berdasarkan 1 atau beberapa pilihan



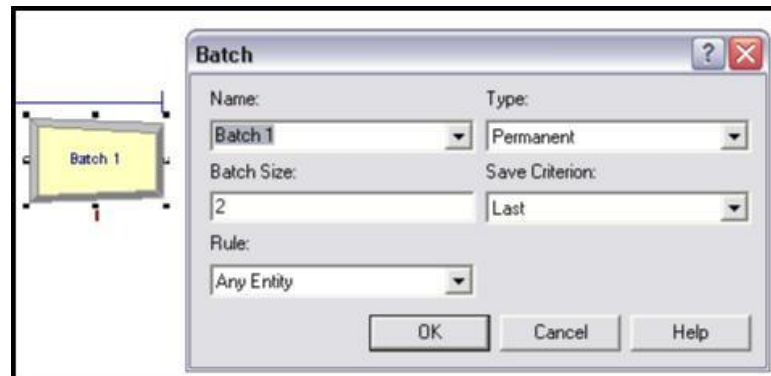
*Type* : mengidentifikasi apakah keputusan berdasarkan pada kondisi dan dapat dispesifikasikan menjadi 2 jenis, yaitu :

- 2 – way : digunakan jika hanya untuk 1 kondisi benar atau salah
  - 2 – way by chance
  - 2 – way by condition
- N – way : digunakan untuk berapapun jumlah kondisi
  - N – way by chance : mendefinisikan satu atau lebih persentase
  - N – way by condition : mendefinisikan satu atau lebih kondisi

Percent true (0-100) : nilai yang digunakan untuk menetapkan entity yang keluar, nilai yang keluar nantinya adalah nilai yang bernilai benar

#### e. Batch

Modul ini digunakan untuk menggabungkan beberapa entity/ assembly



Type : tipe daripada *assembly*, terdiri dari dua jenis yaitu :

- *Temporary* : *assembly* bersifat sementara sehingga dapat dilakukan *disassembly* ketika diperlukan
- *Permanent* : *assembly* bersifat permanen sehingga tidak dapat *breakdown* lagi

*Batch size* : syarat jumlah entity yang sesuai dengan persyaratan yang masuk dalam modul ini untuk dapat dilakukan proses *assembly*

*Save Criterion* : atribut terakhir yang melekat pada output daripada *assembly*. Terdiri dari beberapa kriteria :

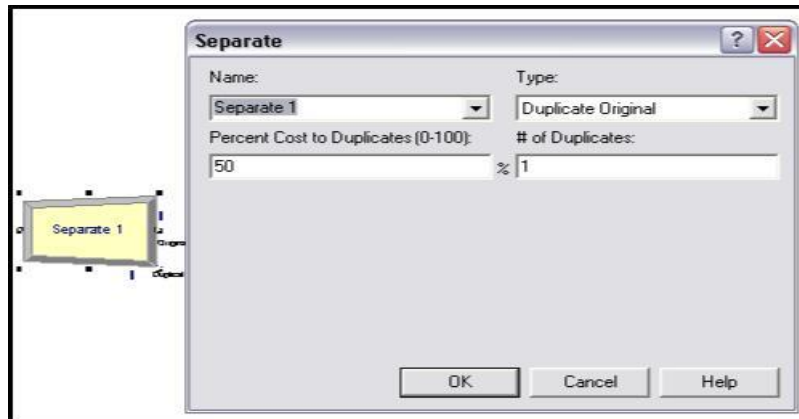
- *First* → atribut yang melekat pada output *assembly* sama dengan atribut entity yang pertama kali masuk dalam proses *assembly*
- *Last* → atribut yang melekat pada output *assembly* sama dengan atribut entity yang terakhir kali masuk dalam proses *assembly*
- *Product* → atribut yang melekat pada output *assembly* berbeda dengan atribut entity yang masuk dalam proses *assembly*

*Rule* : aturan entity yang memenuhi syarat untuk digunakan dalam *assembly*. Ada dua jenis aturan yang dapat digunakan, yaitu :

- *Any entity* → setiap entity yang masuk dalam modul ini diasumsikan dapat digunakan untuk *assembly*
- *By attribute* → entity yang dapat digunakan untuk *assembly* adalah entity yang memiliki atribut sesuai dengan yang telah ditentukan

#### f. Separate

Modul ini digunakan untuk men-*disassembly* hasil dari modul batch, atau juga bisa diasumsikan sebagai aliran entity yang terpisah. Misal pada sistem rumah sakit pasien membawa resep dokter, maka aliran antara entity pasien dengan resep akan berbeda pada titik-titik tertentu



Type : tipe daripada modul separate yang digunakan. Terdiri dari dua jenis, yaitu :

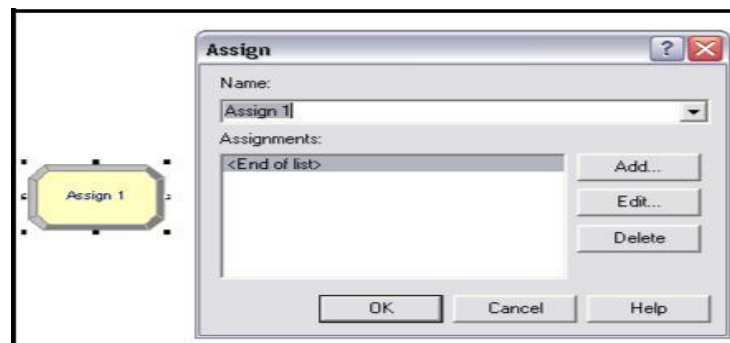
- *Split existing batch* : memisahkan rakitan yang sudah ada (entity yang berasal dari modul batch)
- *Duplicate original* : menduplikat entity yang ada seperti pada kasus pasien dengan resep dokter

Pada saat type modul ini adalah *split existing batch*, maka akan muncul member atribut yang berguna untuk mengirim atribut pada masing-masing entity yang telah *breakdown*. Terdiri dari beberapa jenis, antara lain :

- *Retain original entity values* : nilai pada masing-masing entity sama

#### g. Assign

Modul ini digunakan untuk memasukkan nilai baru pada variabel, entity atribut, entity type, atau variabel lain pada system



Assignments : untuk menspesifikasikan satu atau lebih tugas yang akan dibuat

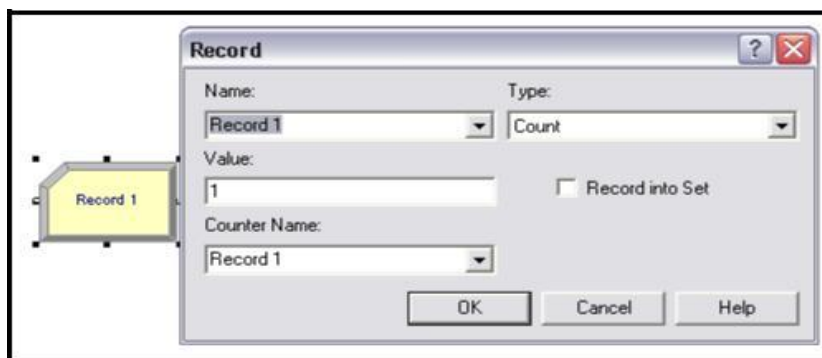
Type : tipe dari tugas yang akan dilakukan terdiri dari :

- *Variable* : nama yang diberikan pada sebuah entity variabel dengan nilai baru
- *Atribute* : nama yang diberikan pada sebuah entity atribut dengan nilai baru
- *Entity type* : sebuah type baru dari entity
- *Entity picture* : sebuah tipe baru berupa gambar
- *Other* : untuk mengidentifikasi untuk atribut yang lainnya

*New value* : nilai baru pada atribut, variabel, atau variabel sistem lainnya. Tidak dapat digunakan untuk *entity tipe* dan *entity picture*

#### h. Record

Modul ini digunakan untuk memunculkan data statistik pada model simulasi, tipe data statistik yang dapat dimunculkan seperti waktu antar kedatangan.



*Type* : terdiri dari *count*, *entity statistic*, *time interval*, *time between*, *expression*

- *Count* : menurunkan atau menaikkan nilai statistic
- *Entity statistic* : menunjukkan nilai statistik secara umum seperti waktu, biaya
- *Time interval* : melacak dan mencatat waktu antar kedatangan
- *Expression* : mencatat nilai dari suatu nilai

*Value* : mencatat data yang menggunakan statistik, tipe yang digunakan adalah ekspresi atau bisa dengan count


*Counter name* : mendefinisikan penambahan/ penurunan data statistik, digunakan jika typenya counter



*Record into set* : *cek box* yang digunakan apakah akan digunakan penanda *tally* alat penghitung lainnya



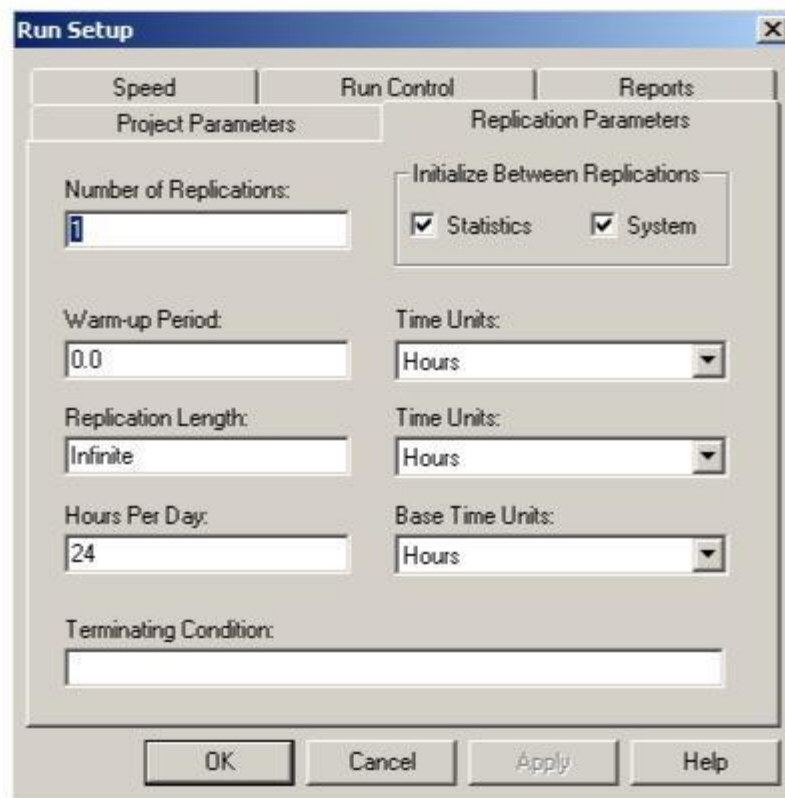
## 2. Menjalankan dan mengatur simulasi

### Running it

Untuk menjalankan model, klik Go button  (atau Run/Go pada menu atau dengan menekan F5). Sebelum kita menjalankan model, kita mungkin ingin memeriksa apakah ada *error* model, kita dapat melakukan dengan cara *run/check model* pada menu atau menekan tombol F4. Setelah *dheck* maka kita dapat melihat apakah model Arena kita terdapat *error* atau tidak.

Bila running ingin dihentikan, hal pertama yang dilakukan adalah menekan *pause button*  kemudian klik *end button* .

Bila model telah siap sebelum kita dapat me-*run* model tersebut. Selain itu kita dapat melakukan *setup run condition*. Caranya: pilih *menu bar, run* → *setup*, akan muncul kotak dialog dengan 5 *tabs*.



#### a. Speed

Tempat mengatur kecepatan dari simulasi yang akan kita *running*

#### b. Run control

Tempat memberikan control apa saja yang berhubungan dnegan *running* Arena yang akan kita *running*

#### c. Reports

Berhubungan dengan *report* yang akan kita dapatkan setelah akhir dari *run model* simulasi. Ada beberapa *default report* yang ada di Arena, tetapi pada *response* ini kita menggunakan SIMAN reports.

#### d. Project parameters

Tempat mengatur *project tittle*, *analyst name* dan memilih output yang ingin kita lihat. Dan yang harus diingat jangan meletakkan tanda baca selain nomor dan alphabets dalam judul atau nama. Contoh yang tidak diijinkan: Mc'Phee → tidak boleh ada tanda '

#### e. Replication parameters

Tempat mengatur *properties* yang ada pada model simulasi yang akan di-*running*. Ada beberapa tab dalam *replication parameters* ini, antara lain:

- *Warm-up period*  
Sejumlah waktu yang digunakan untuk menjadikan model kita *steady-state*. Fungsi dari *warm-up period* adalah untuk mengurangi efek random pada simulasi.
- *Time unit*  
Merupakan satuan waktu yang digunakan sebagai dasar waktu saat menjalankan Arena
- *Replication length*  
Merupakan sejumlah waktu yang ada untuk tiap replikasi yang digunakan di Arena.
- *Hours per day*  
Merupakan banyaknya waktu per hari saat dilakukan running Arena
- *Initialize between replications*
  - ✓ *Initialize system*
    - Bila ada (dicentang), setiap replikasi akan dimulai dengan system kosong disaat waktu 0
    - Bila tidak ada (tidak dicentang), hanya replikasi pertama yang dimulai dengan system kosong disaat waktu 0, dan akan terus berlanjut untuk seterusnya.  
Pekerjaan yang tidak terselesaikan di hari pertama akan dilanjutkan di hari berikutnya.
  - ✓ *Initialize statistics*
    - Bila ada, *report* dengan format yang sama akan di-*generate* untuk setiap replikasi dan data untuk replikasi tunggal bila *initialize system* dipilih. Bila *initialize system* tidak dipilih, data seharusnya untuk replikasi pertama dalam *report 1*, untuk 3 replikasi yang pertama di *report 2*, etc.
    - Bila tidak ada, *report* yang di-*generate* adalah kumulatif, jadi, *report 2* seharusnya terdiri dari statistik dua replikasi yang pertama; *report 3* seharusnya terdiri dari statistik tiga replikasi yang pertama, etc.

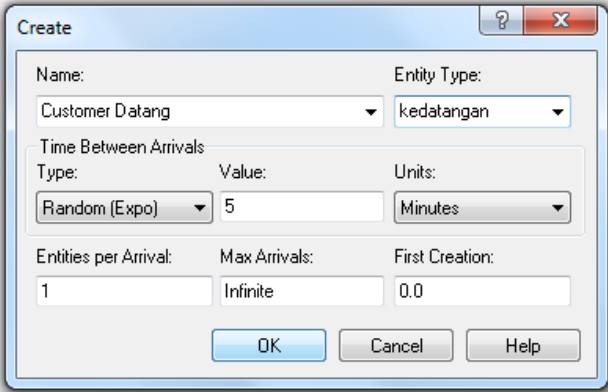
## Modul 2. Simulasi Antrian

### A. Studi kasus 1

Pada suatu pelayanan pendaftaran anggota di kantor BPJS terdapat dua tahapan yang harus dilalui secara berurutan yaitu Pelayanan Pengisian Biodata Anggota dan Penyetoran Iuran Awal yang tiap pelayanan dilayani oleh satu petugas ( 1 Orang Petugas Pelayanan Pengisian Biodata dan 1 Orang Petugas Penyetoran Iuran Awal). Dari data historis yang dikumpulkan dalam satuan menit diketahui bahwa waktu antar kedatangan konsumen yang mendaftar jadi anggota BPJS adalah EXPO (5), sedangkan waktu Pelayanan Pengisian Biodata adalah NORM (6,2) dan Penyetoran Iuran Awal adalah NORM (5,2). Buatlah sistem simulasi tersebut selama 1 hari 8 jam kerja !

### Langkah-langkah Pengerjaan :

1. Untuk memulai nya, pertama buka program Arena 14.
2. *Drag and Drop Modul Create* di Model Window Canvas.
3. *Double click* pada modul *create* yang sudah dibuat, lalu pengisiannya sebagai berikut :

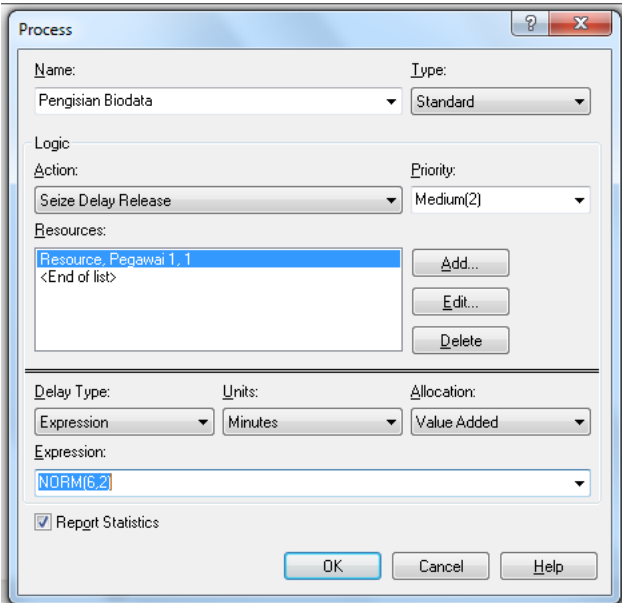


The 'Create' dialog box in Arena simulation software is shown. It has a title bar with a question mark and a close button. The main area contains several fields and buttons:

- Name:** Customer Datang (dropdown)
- Entity Type:** kedatangan (dropdown)
- Time Between Arrivals:**
  - Type:** Random (Expo) (dropdown)
  - Value:** 5 (text input)
  - Units:** Minutes (dropdown)
- Entities per Arrival:** 1 (text input)
- Max Arrivals:** Infinite (text input)
- First Creation:** 0.0 (text input)

At the bottom, there are three buttons: OK, Cancel, and Help.

4. *Drag and Drop modul Proses* di Model Window Canvas
5. *Double click* pada modul proses yang sudah dibuat, lalu pengisiannya sebagai berikut :



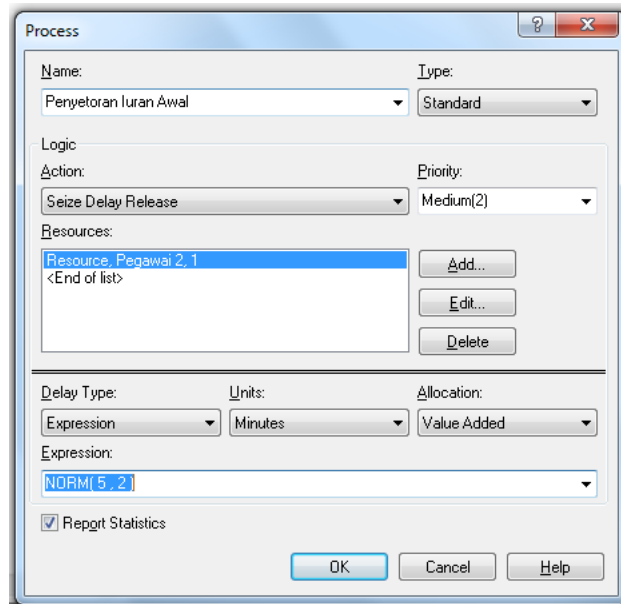
The 'Process' dialog box in Arena simulation software is shown. It has a title bar with a question mark and a close button. The main area contains several fields and buttons:

- Name:** Pengisian Biodata (dropdown)
- Type:** Standard (dropdown)
- Logic:**
  - Action:** Seize Delay Release (dropdown)
  - Priority:** Medium(2) (dropdown)
- Resources:** Resource\_Pegawai 1, 1 (list box), <End of list> (text input). Buttons: Add..., Edit..., Delete.
- Delay Type:** Expression (dropdown)
- Units:** Minutes (dropdown)
- Allocation:** Value Added (dropdown)
- Expression:** NORM(6,2) (text input)
- Report Statistics

At the bottom, there are three buttons: OK, Cancel, and Help.

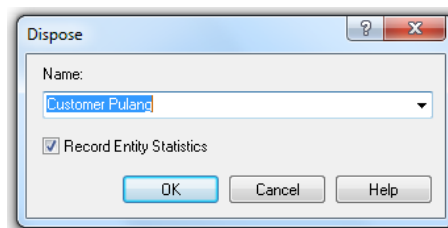
6. Drag and Drop modul Proses di Model Window Canvas

7. Double click pada modul proses yang sudah dibuat, lalu pengisiannya sebagai berikut :

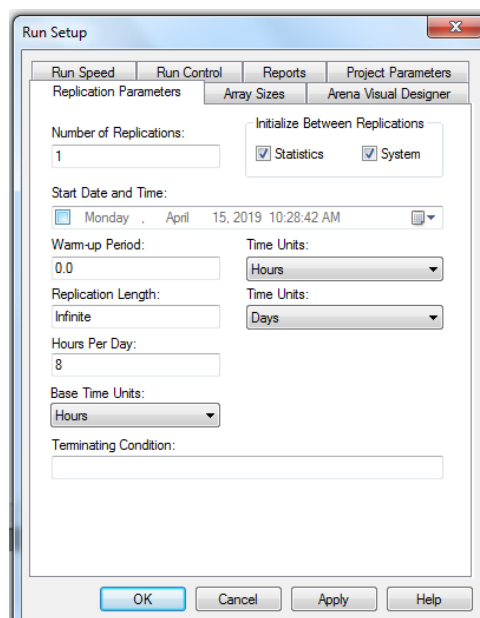


8. Drag and Drop modul Dispose di Model Window Canvas

9. Double click pada modul dispose yang sudah dibuat, lalu pengisiannya sebagai berikut :



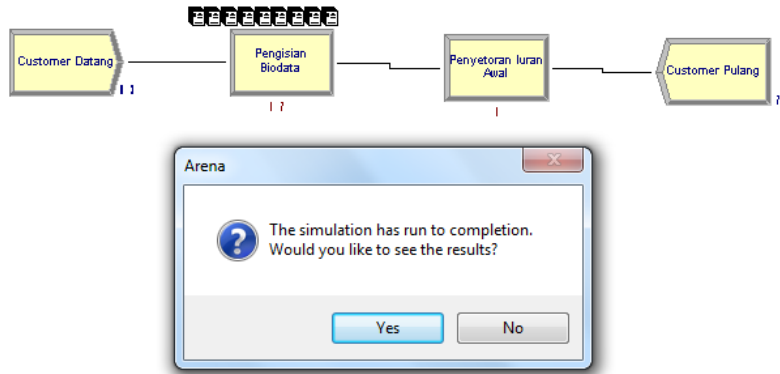
10. Klik Run – Run Setup, lalu pengisiannya sebagai berikut :



11. Kemudian klik tombol berikut untuk menjalankan simulasi



12. Setelah beberapa waktu dijalankan kemudian akan muncul seperti gambar dibawah ini. Hal tersebut menunjukkan bahwa simulasi telah selesai dan apakah akan melihat hasil report dari simulasi ? klik Yes jika ingin melihat hasil *reportnya*, klik No jika sebaliknya.



13. Setelah di klik tombol Yes, maka akan muncul tampilan *report* seperti berikut :

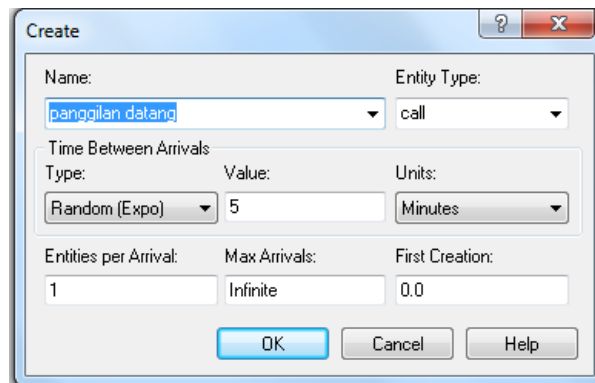
A screenshot of the Arena software interface. The top menu bar includes File, Edit, View, Tools, Arrange, Object, Run, Window, and Help. Below the menu bar is a toolbar with various icons. The main window displays a report titled 'Category Overview' for 'Unnamed Project'. The report shows the time '10:42:46AM' and the date 'April 15, 2019'. The report content includes: 'Replications: 1', 'Time Units: Hours', and 'Key Performance Indicators'. A table shows the 'System' with 'Number Out' and 'Average' values of 75. The left sidebar contains a 'Project Bar' with various simulation components like 'Advanced Transfer', 'Basic Process', 'Flow Process', etc. The bottom right corner of the window has a 'powered by crystal' logo.

## B. Studi Kasus 2

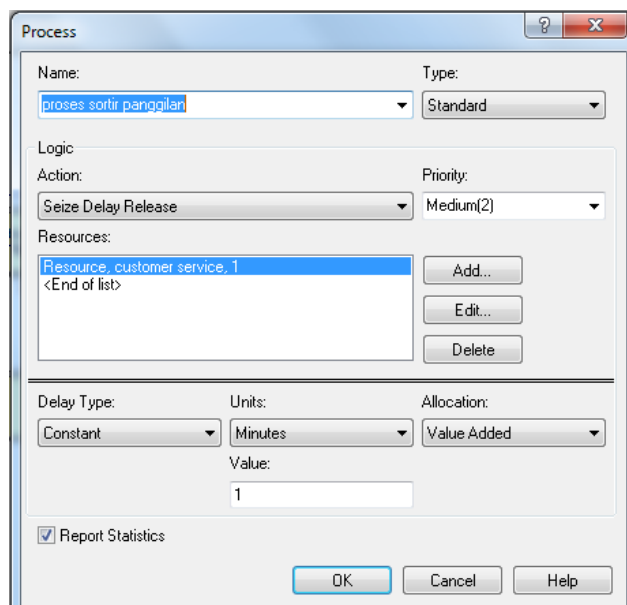
Suatu panggilan telepon datang mengikuti distribusi eksponensial setiap 20 menit. Lalu akan dilakukan proses sortir panggilan oleh seorang *customer service* 1 selama 1 menit. Probabilitas panggilan dilanjutkan kepada *customer service* 2 adalah 80%. Jika panggilan telepon tidak dilanjutkan, maka panggilan akan diputus. Jika disambungkan maka panggilan akan disambungkan kepada *customer service*. Sambungan panggilan tersebut dilakukan secara bergantian oleh 2 orang operator panggilan dengan lama proses berdistribusi normal yang berata-rata 1 menit dan *standard deviasi* sebesar 0,2. Buatlah sistem simulasi tersebut!

### Langkah-langkah Pengerjaan :

1. Untuk memulai nya, pertama buka program Arena 14.
2. *Drag and Drop Modul Create* di Model Window Canvas.
3. *Double click* pada modul *create* yang sudah dibuat, lalu pengisiannya sebagai berikut :

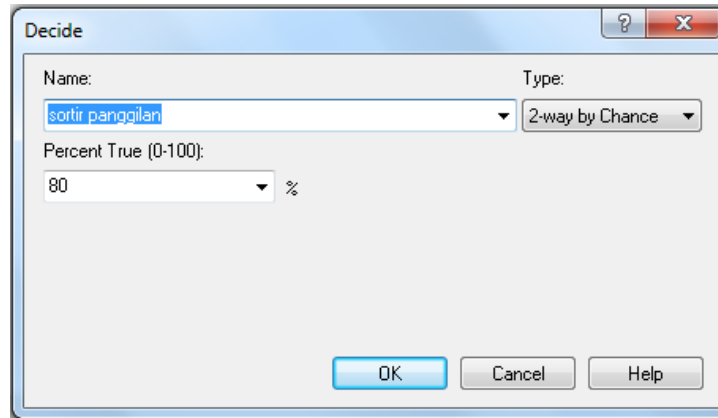


4. *Drag and Drop* modul Proses di Model Window Canvas
5. *Double click* pada modul proses yang sudah dibuat, lalu pengisiannya sebagai berikut :



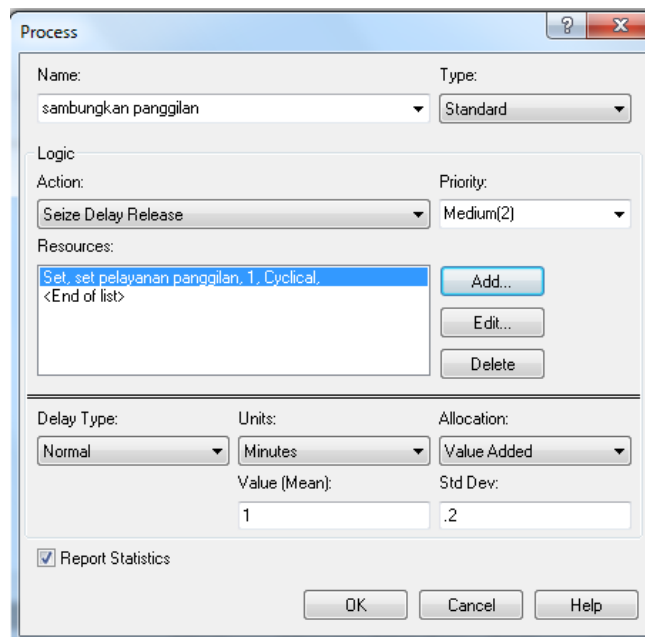
6. Drag and Drop modul *Decide* di Model Window Canvas

7. Double click pada modul *decide* yang sudah dibuat, lalu pengisiannya sebagai berikut :



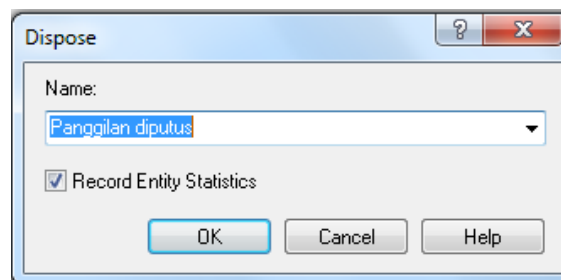
8. Drag and Drop modul Proses di Model Window Canvas

9. Double click pada modul proses yang sudah dibuat, lalu pengisiannya sebagai berikut :



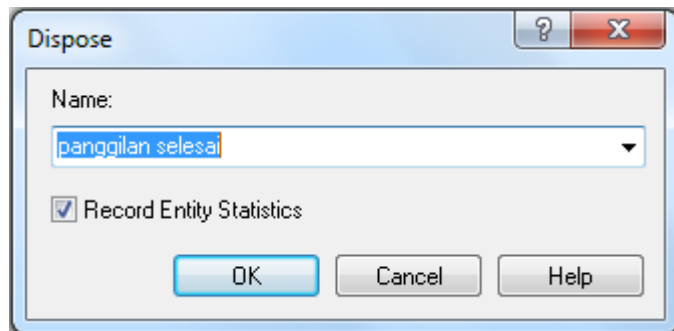
10. Drag and Drop modul *Dispose* di Model Window Canvas

11. Double click pada modul *dispose* yang sudah dibuat, lalu pengisiannya sebagai berikut :

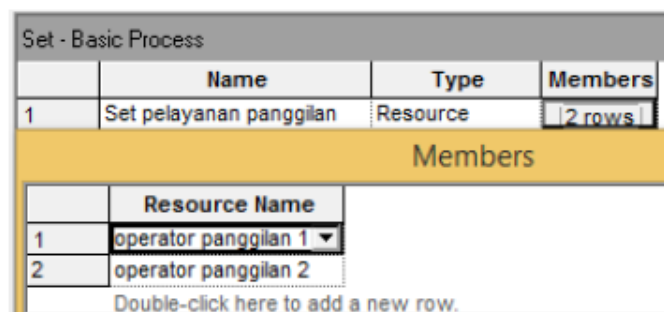


11. Drag and Drop modul *Dispose* di Model Window Canvas

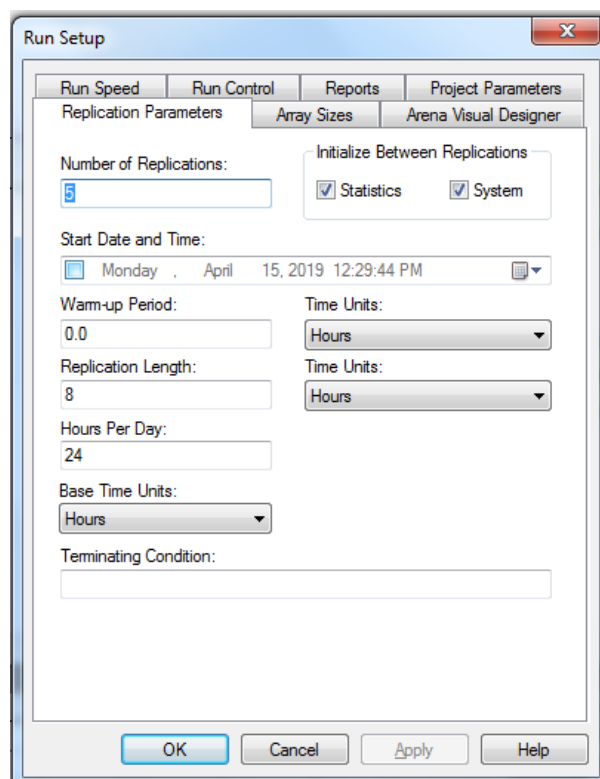
12. Double click pada modul dispose yang sudah dibuat, lalu pengisiannya sebagai berikut :



13. Definisikan *set resource* diatas. Klik pada modul *resource* di *basic process*. Klik 0 rows pada *members* di *set* pelayanan panggilan. Isi *resource name* dengan nama operator panggilan 1 dan operator panggilan 2 seperti gambar berikut :

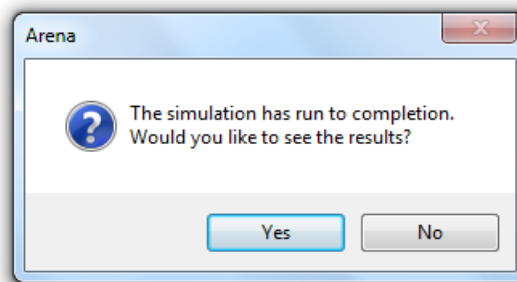
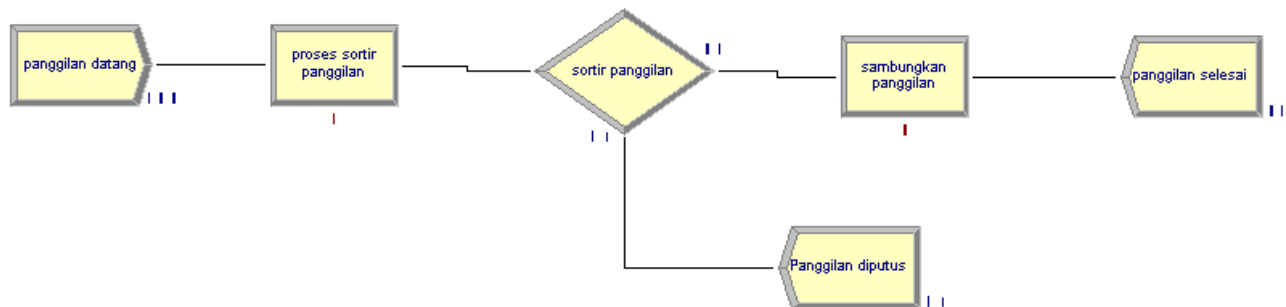


14. Klik *Run – Run Setup*, lalu pengisiannya sebagai berikut :





15. Klik *Icon Play* pada *toolbars*, maka simulasi akan berjalan
16. Setelah beberapa waktu dijalankan kemudian akan muncul seperti gambar dibawah ini. Hal tersebut menunjukkan bahwa simulasi telah selesai dan apakah akan melihat hasil report dari simulasi ? klik *Yes* jika ingin melihat hasil *reportnya*, klik *No* jika sebaliknya.



17. Setelah di klik tombol *Yes*, maka akan muncul tampilan *report* seperti berikut :

The screenshot shows a report window titled 'Category Overview' for 'Unnamed Project'. The report is dated April 15, 2019, at 12:37:51PM. It indicates that the simulation ran for 5 replications with time units in hours. The 'Key Performance Indicators' section shows that the 'System' has an 'Average' value of 97 for the 'Number Out' metric. A 'Subreport:Summary' button is visible at the bottom.

| System     | Average |
|------------|---------|
| Number Out | 97      |

## Modul 3. Arena Advance

*Software Arena Advanced* memiliki modul yang lebih banyak dan bervariasi dalam penggunaannya untuk memodelkan suatu sistem. Sistem simulasi yang akan diterapkan dalam praktikum *Arena Advanced* ini selain menggunakan *Basic Process* juga menggunakan *Advanced Process Panel* dan *Advanced Transfer Panel*.

### A. Advanced Process Panel :

*Advanced process panel* adalah panel yang memiliki beberapa modul yang memiliki fungsi dan aplikasi proses yang lebih bervariasi dari pada *panel basic process*. Panel tersebut dibagi menjadi *General Flowchart Module* dan *Data Module* (User Guide *Arena*, 2005:49).

### B. General Flowchart Module

*General flowchart module* adalah kumpulan dari objek yang ditempatkan pada jendela model untuk mendeskripsikan proses simulasi. Berikut ini adalah beberapa modul pada *general flowchart module*.

#### 1. Hold Module :

Modul ini akan menahan sebuah entitas dalam sebuah antrian dengan beberapa pilihan, yakni: untuk menunggu sinyal, menunggu untuk kondisi tertentu yang diinginkan kemudian dilakukan pemindaian, atau tertahan selama waktu yang tidak terbatas (menjadi hilang kemudian dengan *Remove module*).

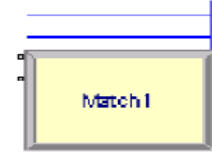


Contoh: Saat menunggu lampu lalu lintas menjadi hijau, menunggu mengambil kue saat oven bunyi  
Daftar elemen petunjuk yang terdapat dalam *Hold Module*:

| Petunjuk              | Deskripsi  |
|-----------------------|--|
| <b>Name</b>           | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .  |
| <i>Type</i>           | Menunjukkan alasan untuk memegang entitas karena sebuah alas an tertentu atau antrian internal. Terdapat 3 pilihan, yaitu: <i>Wait for Signal, Scan for Condition, Infinite Hold</i> .   |
| <i>Wait for Value</i> | Kode sinyal untuk entitas yang menunggu. Hanya berlaku ketika <i>Wait for Signal</i> di gunakan pada <i>Hold Module</i> .  |
| <i>Limit</i>          | Angka Maksimum dari entitas yang menunggu yang akan di lepaskan atas lepasnya <i>signal module</i> .   |
| <i>Condition</i>      | Menentukan sebuah kondisi yang akan dievaluasi untuk menahan entitas di modul. Jika kondisi yang dievaluasi benar, maka entitas akan meninggalkan <i>module</i> , sedangkan tidak cocok dengan kondisi maka entitas akan menunggu. |
| <i>Queue Type</i>     | Membedakan jenis dari antrian yang digunakan untuk menahan entitas. Terdapat 5 pilihan, yaitu : <i>Queue, Set, Internal, Attribute, Expression</i>   |
| <i>Queue Name</i>     | Didalam <i>field</i> ini hanya terlihat jika memilih <i>queue</i> , dan ini mendefinisikan nama sebuah antrian.  |
| <i>Set Name</i>       | Kolom ini hanya terlihat jika memilih <i>Queue Type Set</i> , kolom ini hanya menampilkan deklarasikan nama dari Antrian <i>Set</i> tersebut   |
| <i>Set Index</i>      | Kolom ini hanya akan terlihat jika <i>Queue Type Set</i> , dan mendefinisikan indeks ke <i>queue set</i> . Catatan bahwa ini adalah indeks ke <i>set</i> dan tidak nama dari antrian di <i>set</i> .                               |
| <i>Attribute</i>      | Kolom ini hanya ada jika memilih <i>Queue Set Attribute</i> . Sebuah Atribut akan di masukan ke dalam kolom ini yang akan diproses untuk menunjukan dimana antrian akan digunakan  |
| <i>Expression</i>     | Kolom ini hanya ada jika memilih <i>Queue Type Expression</i> . <i>Expression</i> akan dimasukan kedalam kolom ini yang akan diproses untuk menunjukan dimana antrian akan digunakan.  |

## 2. Match Module :

*Match module* membawa beberapa entitas sekaligus untuk menunggu di antrian yang berbeda. Saat entitas datang pada *Match Module*, entitas akan tetap pada antriannya sampai terjadi kecocokan. Disaat ada satu kecocokan, satu entitas pada setiap antrian akan lepas dan bertemu. Entitas yang bertemu akan tersinkronisasi untuk meninggalkan modul.



Contoh penggunaan: Mempertemukan produk yang bervariasi untuk pesanan pelanggan, Sinkronisasi pelanggan yang keluar dengan pesanan terisi.

Daftar elemen petunjuk yang terdapat dalam *Match Module*:

| Petunjuk               | Deskripsi   |
|------------------------|---|
| <b>Name</b>            | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .   |
| <i>Number of Match</i> | Sebuah jumlah entitas yang akan dicocokkan yang harus disatukan didalam antrian yang berbeda sebelum sebuah penggabungan akan diselesaikan  |
| <i>Type</i>            | Metode dari pencocokan dari entitas yang datang. Terdapat 2 pilihan didalamnya, yaitu <i>Any Entities</i> dan <i>Based on Attribute</i> .   |
| <i>Attribute Name</i>  | Nama atribut yang akan digunakan untuk mengidentifikasi kedatangan dari pencocokan entitas. Hanya tampil jika memilih <i>Based on Attribute</i>   |
| <i>Condition</i>       | Menentukan sebuah kondisi yang akan dievaluasi untuk menahan entitas di modul. Jika kondisi yang dievaluasi benar, maka entitas akan meninggalkan modul, sedangkan tidak cocok dengan kondisi maka entitas akan menunggu. |

## 3. Signal Module :

*Signal module* mengirimkan sebuah *signal* atau tanda untuk setiap *hold module* dalam model yang sedang menunggu sinyal untuk melepaskan entitas tertentu. Saat entitas datang ke *signal module*, sinyal dievaluasi dan kode sinyal dikirimkan. pada saat itu, entitas di *Hold Module* yang menunggu untuk sinyal yang sama akan hilang dari antrian.



Contoh penggunaan: Memberikan tanda operator untuk menyelesaikan pesanan yang menunggu komponen lain.

Daftar elemen petunjuk yang terdapat dalam *Signal Module*:

| Petunjuk             | Deskripsi   |
|----------------------|---|
| <b>Name</b>          | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .   |
| <i>Signals Value</i> | Nilai dari sebuah <i>signal</i> untuk mengirim entitas didalam <i>Hold Module</i>                       |
| <i>Limit</i>         | Angka maksimum dari entitas yang akan di lepaskan dari <i>Hold Module</i> ketika <i>signal</i> diterima |

## 4. Delay Module :

*Delay module* menunda sebuah entitas dengan spesifikasi waktu tertentu. Saat sebuah entitas datang pada *Delay module*, pernyataan waktu tunda dievaluasi dan entitas tetap pada modul sampai periode waktu tertentu.



Contoh: Proses pengecekan di bank, melakukan proses mesin jahit

Daftar elemen petunjuk yang terdapat dalam *Delay Module*:

| Petunjuk          | Deskripsi   |
|-------------------|---|
| <b>Name</b>       | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .                                       |
| <i>Allocation</i> | Jenis dari kategori dimana entitas terjadi <i>Delay time</i> / waktu proses dan biaya di masukan. |
| <i>Delay Time</i> | Membedakan nilai dari waktu proses dari entitas   |
| <i>Units</i>      | Unit waktu yang digunakan untuk <i>Delay time</i>   |

### 5. Release Module :

Release module digunakan untuk melepaskan beberapa resource yang telah memproses entitas sebelumnya.

Contoh: Melepaskan pekerja Setelah menjahit dari kain.

Daftar elemen petunjuk yang terdapat dalam Release Module:



| Petunjuk       | Deskripsi  |
|----------------|--|
| Name           | Nama module yang ditampilkan pada interface.   |
| Type           | Jenis dari resource untuk releasing, baik menentukan resource tertentu, maupun kumpulan dari resource.   |
| Resource Name  | Nama dari resource yang akan di released.  |
| Set Name       | Nama dari resource set yang akan di released.  |
| Attribute Name | Nama dari attribute yang ditentukan dimana akan di released.   |
| Expression     | Nama dari Expression yang ditentukan dimana akan di released.  |
| Quantity       | Jumlah dari resource yang akan dilepaskan. Untuk set, nilai ini hanya angka dari resource yang akan dilepaskan (berdasarkan resource kapasitasnya)       |
| Released Rule  | Metode penentuan dari resource dengan sebuah set untuk melepaskan. Pilihannya ada 3 yaitu, last member seized, first member seized, dan Specific member. |
| Set Index      | Anggota index dari resource set yang akan dilepaskan   |

### 6. Seize Module

Seize Module digunakan untuk mengalokasikan unit satuan atau lebih resource menjadi entitas. Seize module dapat digunakan untuk meraih, menangkap unit dari resource khusus, anggota dari resource set, atau resource yang terdefinisi sebagai metode alternatif, seperti atribut atau ekspresi. Saat entitas memasuki modul ini, entitas akan menunggu di antrian sampai semua resource tertentu tersedia secara serempak. Tipe alokasi dari resource dapat ditentukan.

Daftar elemen petunjuk yang terdapat dalam Seize Module:



| Petunjuk       | Deskripsi  |
|----------------|--|
| Name           | Nama module yang ditampilkan pada interface.   |
| Allocation     | Menentukan pada kategori mana biaya penggunaan operator akan dialokasikan untuk sebuah entitas melalui Seize Module.   |
| Priority       | Nilai prioritas dari entitas yang menunggu pada modul ini untuk resource yang spesifik jika 1 atau lebih entitas dari modul lain sedang menunggu resource yang sama. |
| Type           | Tipe dari resource yang akan dipanggil, baik dari resource yang spesifik atau memilih dari sekumpulan resource (resource set).                                       |
| Resource Name  | Nama dari resource yang akan dipanggil.  |
| Set Name       | Nama dari resource set dimana sebuah member akan dipanggil.  |
| Attribute Name | Nama dari atribut yang menyimpan nama resource yang akan dipanggil.  |
| Selection Rule | Metode pemilihan dari resource yang tersedia dalam set.  |
| Save Attribute | Nama atribut yang digunakan untuk menyimpan index number ke dalam set dari member yang dipilih.  |
| Set Index      | Anggota index dari resource set yang akan dilepaskan   |
| Queue Type     | Menentukan tipe antrian yang digunakan untuk menahan entitas ketika menunggu pemanggilan resource.   |

### C. Data Module

Data module adalah kumpulan objek yang ada di tampilan lembar kerja dari model yang mendefinisikan karakteristik bermacam-macam elemen proses seperti advanced set module dan Expression module. Berikut ini adalah bebrapa modul dalam data module Advanced Transfer Panel.

### 1. Advanced Set Module

*Advance set module* menentukan *set* antrian, *set storage*, dan *set-set* yang lain, dan masing-masing bagiannya.

Contoh: berbagai pintu keluar di toko (antrian).



### 2. Expression module

*Expression module* adalah ekspresi dan nilai-nilai yang berhubungan.

Contoh: *Expression* kompleks untuk waktu pesanan masuk.



## D. Advanced Transfer Process

*Advanced transfer panel* adalah *panel* yang memiliki beberapa modul yang memiliki fungsi dan aplikasi transfer atau transportasi yang lebih bervariasi. *Panel* tersebut dibagi menjadi 4 bagian yaitu *General Flowchart Module*, *Conveyor Flowchart modules*, *Transporter Flowchart Modules*, dan *Data Modules* (User Guide Arena, 2005:71).

## E. General Flowchart Modules

*General flowchart module* adalah kumpulan dari objek yang ditempatkan pada jendela model untuk mendeskripsikan proses simulasi. *General flowchart module* berwarna merah. Berikut ini adalah beberapa modul pada *General flowchart module*.

### 1. Station Module :

*Station module* mendefinisikan sebuah *station* (atau kumpulan *station*) yang cocok secara fisik atau logis lokasi dimana proses dilakukan. Jika *station module* terdefinisi sebagai sebuah kumpulan *station*, maka secara efektif akan menjadi lokasi yang multi proses.



Contoh penggunaan: Menetapkan area pembubutan, menetapkan area persiapan makanan.

Daftar elemen petunjuk yang terdapat dalam *Station Module*.

| Petunjuk                       | Deskripsi  |
|--------------------------------|--|
| <i>Name</i>                    | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .  |
| <i>Station Type</i>            | Jenis dari stasiun yang akan didefinisikan, baik sebagai individual <i>station</i> atau stasiun <i>set</i>   |
| <i>Station Name</i>            | Nama dari individual <i>station</i>  |
| <i>Set Name</i>                | Nama dari <i>Set Station</i>   |
| <i>Parent Activity Area</i>    | Nama dari area aktivitas induk   |
| <i>Associated Intersection</i> | Nama dari tempat pertemuan dengan stasiun ini dalam sebuah tuntunan <i>transporter network</i>   |
| <i>Report Statistics</i>       | Ketentuan acak atau bukan statistik akan otomatis akan dikumpulkan dan disimpan dalam database laporan dan yang sesuai daerah aktivitas.   |
| <i>Save Attribute</i>          | Nama atribut yang digunakan untuk disimpan angka index sampai stasiun <i>set</i> dari <i>member</i> yang diilih  |
| <i>Station Set Members</i>     | Nama dari stasiun yang menjadi member dari stasiun <i>set</i>  |
| <i>Station Name</i>            | Sebuah stasiun yang diberikan hanya dapat berada sekali dalam model. Oleh karena itu, stasiun individu hanya dapat menjadi anggota dari satu stasiun <i>set</i> , dan bahwa stasiun individu mungkin bukan nama stasiun di modul lain. |
| <i>Parent Activity Area</i>    | Nama dari area aktivitas induk untuk anggota <i>set</i> stasiun  |
| <i>Associated Intersection</i> | Nama dari tempat pertemuan dengan stasiun ini dalam sebuah tuntunan <i>transporter network</i>   |
| <i>Report Statistics</i>       | Ketentuan acak atau bukan statistik akan otomatis akan dikumpulkan dan disimpan dalam database laporan dan yang sesuai daerah aktivitas.   |

## 2. Route Module :

*Route module* mentransferkan entitas ke *station* tertentu atau ke *station* selanjutnya di rangkaian *station* kunjungan tertentu untuk entitas. Didalam *route* diasumsikan bahwa *resource* tersedia setiap saat.



Contoh penggunaan: Mengirimkan *part* ke *station* proses selanjutnya berdasarkan rutenya.

Daftar elemen petunjuk yang terdapat dalam *Route Module*:

| Petunjuk                | Deskripsi  |
|-------------------------|--|
| <i>Name</i>             | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .  |
| <i>Route Time</i>       | Waktu pengiriman dari lokasi entitas sekarang ke lokasi tujuan   |
| <i>Units</i>            | Unit waktu dari parameter <i>route-time</i>  |
| <i>Destination Type</i> | Metode untuk menentukan tempat tujuan entitas. Pemilihan berdasarkan <i>By Sequence</i> membutuhkan pendeklarasian terlebih dahulu di <i>module assign</i> |
| <i>Station Name</i>     | Nama dari stasiun yang akan dituju   |
| <i>Transfer Type</i>    | Menentukan bagaimana entitas akan ditransferkan keluar modul ke destinasi tujuan   |
| <i>Save Attribute</i>   | Mendefinisikan nama dari atribut yang akan disimpan nama stasiun yang dipilih  |
| <i>Route Time</i>       | Waktu pindah dari entitas dari stasiun awal ke stasiun yang ditentukan di modul  |
| <i>Units</i>            | Waktu yang dibutuhkan untuk parameter <i>route-time</i>  |

## F. Conveyor Flowchart Modules

*Conveyor flowchart modules* adalah kumpulan dari objek yang ditempatkan pada jendela model untuk mendeskripsikan proses simulasi dengan fungsi yang khusus yaitu *conveyor*. *Conveyor flowchart modules* berwarna hijau (User Guide *Arena*,2005:79).

Berikut ini adalah beberapa modul pada *conveyor flowchart modules*.

### 1. Access Module :

*Access module* mengalokasikan satu atau lebih *cell* dari *conveyor* ke entitas untuk pemindahan dari satu *station* ke yang lain. Saat entitas dikendalikan dari *cell* pada *conveyor*, kemudian dapat dibawa ke *station* selanjutnya. Saat entitas datang ke *Access module*, entitas menunggu sampai jumlah tepat dari *cell* yang berdampingan pada *conveyor* kosong dan meluruskan dengan entitas lokasi *station*.

Contoh penggunaan: *Part* masuk dalam *conveyor* untuk dikirimkan ke daerah percepatan.

Daftar elemen petunjuk yang terdapat dalam *Access Module*:

| Petunjuk              | Deskripsi   |
|-----------------------|---|
| <i>Name</i>           | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .   |
| <i>Conveyor Name</i>  | Nama dari <i>conveyor</i> yang akan dibangkitkan  |
| <i># of Cells</i>     | Jumlah sel <i>conveyor</i> yang dibutuhkan sesuai dengan entitas untuk perpindahan dengan <i>conveyor</i>   |
| <i>Queue Type</i>     | Menentukan <i>type</i> dari antrian yang digunakan untuk menahan entitas. Terdiri atas <i>Queue</i> , <i>Set</i> , <i>Internal</i> , <i>Attribute</i> dan <i>Expression</i> |
| <i>Queue Name</i>     | Nama dari antrian yang akan menahan entitas hingga <i>conveyor</i> dibutuhkan   |
| <i>Set Name</i>       | Nama dari <i>set</i> antrian  |
| <i>Save Attribute</i> | Mendefinisikan nama dari atribut yang akan disimpan nama stasiun yang dipilih   |
| <i>Set Index</i>      | Mendefinisikan dari <i>index set</i> antrian  |
| <i>Attribute Name</i> | Mendefinisikan nama antrian dari atribut yang dikirim dimana entitas akan di pindahkan  |
| <i>Expression</i>     | Mendefinisikan nama antrian dari <i>Expression</i> yang akan dikirim dimana entitas akan dipindahkan  |

## 2. Convey Module

*Convey module* memindahkan entitas pada *conveyor* dari lokasi tertentu ke *station* tujuan. Waktu tunda untuk membawa entitas dari satu *station* ke selanjutnya berdasarkan kecepatan *conveyor* jarak antar *station*.



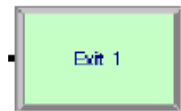
Contoh penggunaan: Membawa *part* dari *station* pengisian ke *station* proses menggunakan *conveyor*.

Daftar elemen petunjuk yang terdapat dalam *Convey Module*:

| Petunjuk                | Deskripsi  |
|-------------------------|--|
| <i>Name</i>             | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .  |
| <i>Conveyor Name</i>    | Nama dari <i>conveyor</i> yang akan digunakan  |
| <i>Destination Type</i> | Metode dari spesifikasi tujuan entitas ( <i>Sequence, station, Attribute, or Expression</i> )        |
| <i>Station Name</i>     | Nama stasiun yang dideklarasikan   |
| <i>Attribute Name</i>   | Mendefinisikan nama antrian dari atribut yang dikirim dimana entitas akan di pindahkan               |
| <i>Expression</i>       | Mendefinisikan nama antrian dari <i>Expression</i> yang akan dikirim dimana entitas akan dipindahkan |

## 3. Exit Module :

*Exit module* melepaskan entitas *cell* di *conveyor* tertentu. Jika entitas yang lain menunggu dalam antrian *conveyor* di *station* yang sama ketika *cell* itu lepas, entitas kemudian masuk ke *conveyor*.



Contoh penggunaannya adalah *part* yang cacat dipindahkan dari *conveyor* dan dibuang.

Daftar elemen petunjuk yang terdapat dalam *Exit Module*:

| Petunjuk             | Deskripsi   |
|----------------------|---|
| <i>Name</i>          | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .   |
| <i>Conveyor Name</i> | Nama dari <i>conveyor</i> yang akan dibangkitkan  |
| <i># of Cells</i>    | Jumlah sel <i>conveyor</i> yang dibutuhkan sesuai dengan entitas untuk perpindahan dengan <i>conveyor</i> |

## G. Transporter Flowchart Modules

*Transporter flowchart modules* adalah kumpulan dari objek yang ditempatkan pada jendela model untuk mendeskripsikan proses simulasi dengan fungsi yang khusus yaitu *transporter*. *Transporter flowchart modules* berwarna biru (User Guide *Arena*, 2005:82). Berikut ini adalah beberapa modul pada *Transporter Flowchart Modules*.

### 1. Request Module:

Sebuah unit *transporter* spesifik dapat ditentukan atau seleksi berdasarkan aturan. Ketika entitas tiba di *request module*, maka entitas akan dialokasikan *transporter* ketika salah satu tersedia. Entitas tetap pada *request module* sampai unit *transporter* telah mencapai stasiun entitas. Kemudian entitas bergerak keluar dari *module Request*.



Daftar elemen petunjuk yang terdapat dalam *Request Module*:

| Petunjuk                | Deskripsi   |
|-------------------------|---|
| <i>Name</i>             | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .   |
| <i>Transporter Name</i> | Nama dari <i>transporter</i> untuk dialokasikan   |
| <i>Selection Rule</i>   | Peraturan untuk menentukan dimana <i>transporter</i> dialokasikan ke dalam entitas. Terdiri atas <i>Cyclical, Random, Preferred Order, Specific Member, Largest Distance, dan Smallest Distance</i> |
| <i>Save Attribute</i>   | Menandakan nama atribut yang akan disimpan unit member <i>transporter</i> terpilih  |
| <i>Unit Number</i>      | Menentukan unit <i>transporter</i> spesifik dalam <i>transporter</i> yang diminta   |

|                        |   |
|------------------------|---|
| <i>Priority</i>        | Nilai prioritas entitas menunggu didalam modul ini untuk segera dilakukan pemindahan jika menggunakan <i>transporter</i> yang sama  |
| <i>Entity Location</i> | Lokasi dimana <i>transporter</i> akan pindah Setelah dialokasikan   |
| <i>Velocity</i>        | Kecepatan <i>transporter</i> tersebut berjalan untuk memindahkan entitas  |
| <i>Units</i>           | Satuan unit kecepatan   |
| <i>Queue Type</i>      | Jenis antrian yang digunakan untuk menahan entitas yang menunggu akses dari <i>transporter</i> .<br>Terdiri atas, <i>Queue, Set, Internal, Attribute</i> atau <i>Expression</i> |
| <i>Queue Name</i>      | Nama dari antrian   |
| <i>Set Name</i>        | Nama dari <i>set</i> antrian  |
| <i>Set Index</i>       | Sebuah <i>Index</i> yang dimasukan kedalam <i>set</i> antrian   |
| <i>Attribute Name</i>  | Nama Atribut yang akan dievaluasi sesuai nama antrian   |
| <i>Expression</i>      | <i>Expression</i> yang dievaluasi sesuai dengan nama antrian  |

## 2. Transport Module

*Transport module* mentransfer entitas pengendali ke stasiun tujuan. Setelah penundaan waktu yang diperlukan untuk transportasi, entitas muncul kembali dalam model di *module Station*.



Contoh penggunaan: *Forklift* mengangkut *pallet* dari *part* ke stasiun pemrosesan selanjutnya

Daftar elemen petunjuk yang terdapat dalam *Transport Module*:

| Petunjuk                            | Deskripsi  |
|-------------------------------------|--|
| <i>Name</i>                         | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .  |
| <i>Transporter Name</i>             | Nama dari <i>transporter</i> untuk dialokasikan  |
| <i>Unit Number</i>                  | Menentukan unit <i>transporter</i> spesifik dalam <i>transporter</i> yang diminta  |
| <i>Entity Destination Type</i>      | Menentukan metode untuk stasiun tujuan yang spesifik. Hanya tampil jika memilih <i>sequence</i>                                    |
| <i>Station Name</i>                 | Nama dari stasiun  |
| <i>Attribute Name</i>               | Nama dari atribut yang disimpan ke nama stasiun yang mana entitas akan di <i>route</i>   |
| <i>Units</i>                        | Satuan unit kecepatan  |
| <i>Expression</i>                   | <i>Expression</i> yang dievaluasi ke stasiun ke jaringan lokasi dimana <i>transporter</i> di arahkan untuk bergerak                |
| <i>Velocity</i>                     | Kecepatan <i>transporter</i> tersebut berjalan untuk memindahkan entitas   |
| <i>Guided Tran Destination Type</i> | Memungkinkan spesifikasi tujuan <i>transporter</i> yang berbeda beda dari tujuan entitas.  |
| <i>Station Name</i>                 | Mendefinisikan nama stasiun dengan sebuah hubungan dimana <i>transporter</i> akan bergerak   |
| <i>Attribute Name</i>               | Mendefinisikan nama atribut yang menyimpan nama stasiun dengan hubungan terkait dimana <i>transporter</i> diarahkan akan bergerak. |
| <i>Intersection Name</i>            | Mendefinisikan nama dari perpotongan dimana <i>transporter</i> diarahkan agar bergerak.  |
| <i>Network Link Name</i>            | Mendefinisikan nama dari <i>network link</i> dimana <i>transporter</i> diarahkan agar bererak                                      |
| <i>Zone</i>                         | Zona jumlah yang spesifik didalam <i>Network Link Name</i>   |

## 3. Free Module

*Free module* melepaskan *entity* yang terakhir dialokasikan di *transporter* unit. Jika entitas lain sedang menunggu dalam antrian untuk diminta atau mengalokasikan *transporter*, maka *transporter* akan diberikan kepada entitas tersebut.



Contoh penggunaan: Part menunggu dilepaskan oleh *forklift* untuk diletakkan di truk pengiriman.

Daftar elemen petunjuk yang terdapat dalam *Free Module*:



| Petunjuk         | Deskripsi  |
|------------------|--|
| Name             | Nama <i>module</i> yang ditampilkan pada <i>interface</i> .  |
| Transporter Name | Nama dari <i>transporter</i> untuk dialokasikan  |
| Velocity         | Kecepatan dari <i>conveyor</i> saat mulai beroperasi. Nilai ini akan menjadi kecepatan <i>conveyor</i> secara permanent, hingga masuk ke <i>module</i> trans lainnya |
| Units Number     | Menentukan dimana unit <i>transporter</i> didalam <i>set transporter</i> yang akan dibebaskan  |

## Studi Kasus 1

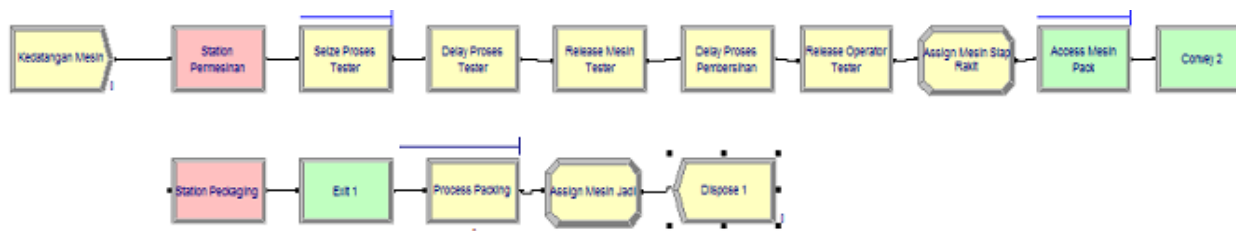
Perusahaan PT. X merupakan perusahaan di bidang pembuatan *sparepart* mesin sepeda motor. Pertama, **kedatangan mesin** di **stasiun permesinan** yang kedatangan setiap 10 menit, masing-masing terdapat 10 mesin yang masuk ke dalam sistem. Mesin dijadwalkan akan terjadi 5 kali kedatangan. Setelah itu, mesin sepeda motor dilakukan tester dengan menggunakan **mesin tester** dan **operator tester** dengan waktu 3 menit. Setelah itu, **operator tester melanjutkan proses selanjutnya** yaitu pembersihan mesin dengan waktu 2 menit yang menghasilkan mesin siap rakit. Setelah mesin sepeda motor telah dibersihkan, akan langsung **dikirimkan ke stasiun perakitan** oleh *conveyor* dengan kecepatan 2 m/s sejauh 10 meter. Lalu dilakukan **proses packing mesin** yang dilakukan **manual oleh operator packaging dan mesin packaging** selama 7 menit. Setelah itu sepeda motor jadi dapat dikirim ke gudang.

Note: Diperhatikan penggunaan fungsi (*seize, delay, release*) dan proses *module* serta diperhatikan penggunaan *transport module*.

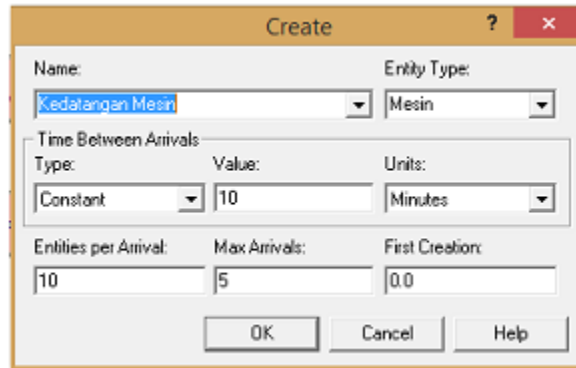
### 1. Langkah-langkah Pembuatan Model Arena

Langkah-langkah yang perlu dilakukan dalam praktikum ini adalah:

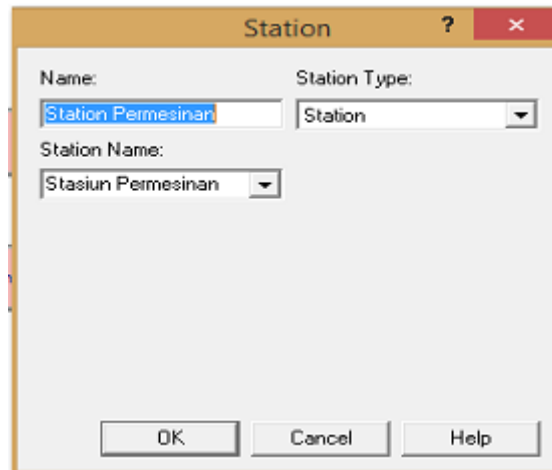
- Membuka *software* Arena.
- Memulai *project* baru. Untuk memulai pembuatan *file* dilakukan dengan membuat sebuah lembar baru dengan mengklik *icon new* atau *file-new* ( atau CTRL + N). Secara otomatis akan diberi nama Model 1 dengan ekstensi *.doe* jika disimpan. Nama tersebut dapat diubah pada saat menyimpan model. Tampilan dapat diperbesar dengan menekan disudut kanan atas.
- Pada *toolbar* sebelah kiri terdapat *basic process panel* yang secara otomatis akan terbuka apabila Arena dijalankan.
- Buatlah model seperti gambar dibawah ini:



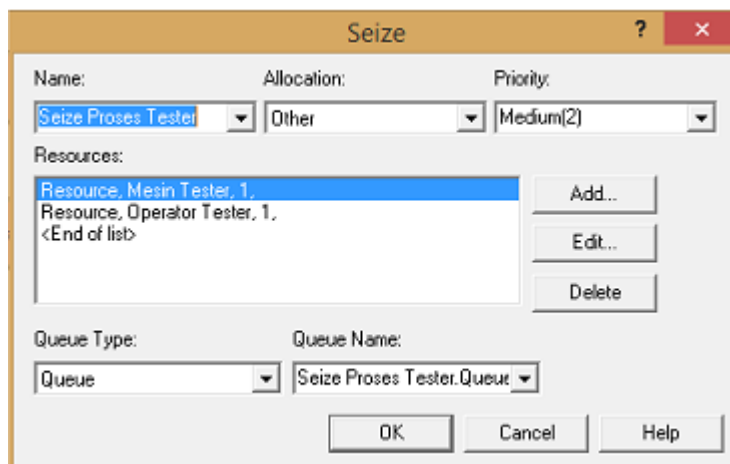
- Pertama, untuk membuat **kedatangan mesin** di **stasiun permesinan** yang kedatangan setiap 10 menit , masing-masing terdapat 10 mesin yang masuk kedalam sistem. Mesin dijadwalkan akan terjadi 5 kali kedatangan pakailah *create module*.



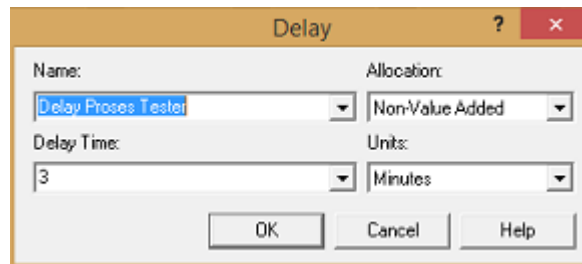
Lalu setelah membuat *create module* buatlah *station module* untuk mendeklarasikan stasiun permesinan dengan isi sebagai berikut:



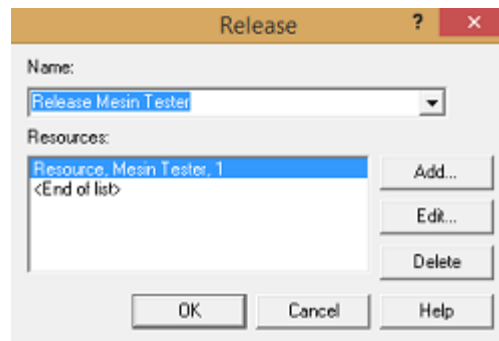
- f. Setelah itu, mesin sepeda motor dilakukan tester dengan menggunakan **mesin tester** dan **operator tester** dengan waktu 3 menit. Setelah itu, **operator tester melanjutkan proses selanjutnya** yaitu pembersihan mesin dengan waktu 2 menit yang menghasilkan mesin siap rakit. Maka pertama-tama kita harus memasukkan resource terlebih dahulu menggunakan seize module yaitu Mesin Tester dan Operator Tester.



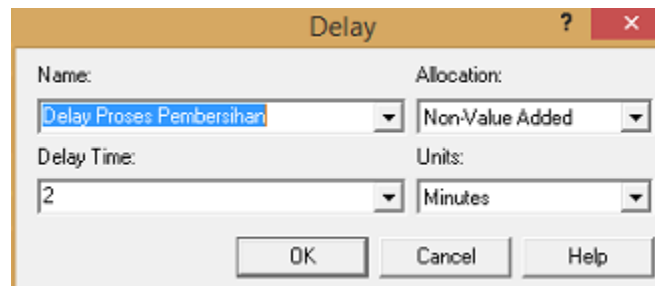
Untuk memasukan waktu proses, kita akan menggunakan *Delay Module*.



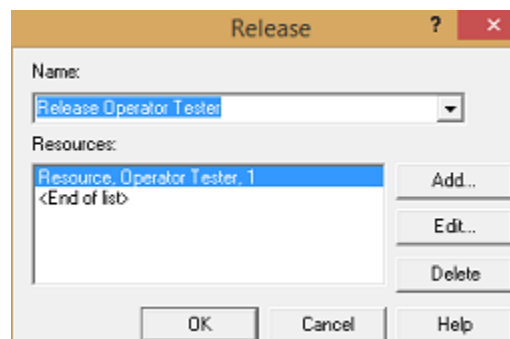
Karena Operator Tester tetap melanjutkan proses Pembersihan, maka *resource* yang harus dilepas adalah hanya Mesin Tester sedangkan operator masih harus ada disistem. Maka menggunakan *Release Module* sebagai berikut.



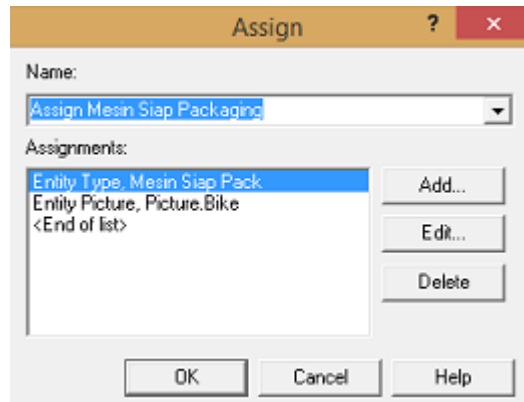
Setelah itu, **operator tester melanjutkan proses selanjutnya** yaitu pembersihan mesin dengan waktu 2 menit yang menghasilkan mesin siap rakit. Karena *Resource* sudah ada di sistem, maka tidak perlu melakukan *seize* lagi.



Setelah proses Pembersihan telah selesai, maka kita perlu melepaskan semua *resource* yang ada.



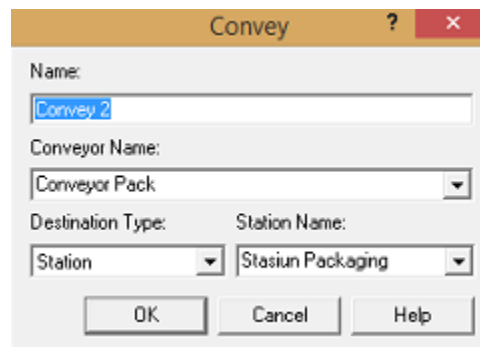
Karena hasil dari proses diatas merupakan entitas baru, yaitu Mesin Siap *Packaging* maka memakai *Assign Module* untuk mendeklarasikan entitas baru.



- g. Setelah mesin sepeda motor telah dibersihkan, akan langsung **dikirimkan ke stasiun perakitan** oleh *conveyor* dengan kecepatan 2 m/s sejauh 10 meter. Maka pertama-tama kita harus mendeklarasikan *access module* untuk mengalokasikan *conveyor* di sistem.



Lalu kita menggunakan *Convey Module*, untuk memerintahkan *conveyor* untuk mengambil entitas tersebut dan mengirimkan ke stasiun yang dituju.



Note: untuk mengisi *Station Name*, untuk memudahkan, kita harus membuat *station* tujuan terlebih dahulu untuk mengurangi peluang kesalahan.

- h. Setelah mesin sepeda motor telah dibersihkan, akan langsung **dikirimkan ke stasiun perakitan** oleh *conveyor* dengan kecepatan 2 m/s sejauh 10 meter.

Sebelum melanjutkan, kita harus mengatur ketentuan-ketentuan yang ada di *conveyor* yang terdapat pada *Conveyor Data Module* sebagai berikut. Lalu kita atur kecepatan *conveyor* 2 m/s.

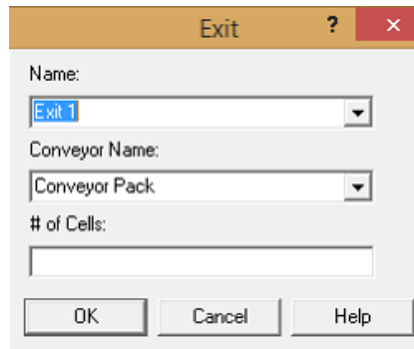
|   | Name          | Segment Name          | Type         | Velocity | Units      | Cell Size | Max Cells Occupied | Accumulation Length | Initial Status | Report Statistics                   |
|---|---------------|-----------------------|--------------|----------|------------|-----------|--------------------|---------------------|----------------|-------------------------------------|
| 1 | Conveyor Pack | Conveyor Pack.Segment | Accumulating | 2        | Per Second | 1         | 1                  | 1                   | Active         | <input checked="" type="checkbox"/> |

- i. Setelah langkah tersebut kita juga harus mendeklarasikan berapa jarak dari stasiun Permesinan dan stasiun *Packaging* dengan menggunakan *Segment Data Module*. *Beginning Station* kita isikan Stasiun Permesinan, dan klik bagian *Next Station* untuk tujuannya. Lalu isikan *Station Packaging*, dan *Length* nya sejauh 10 meter.
- j. Lalu untuk melepaskan *conveyor* dari entitas maka kita perlu melepaskan *conveyor* menggunakan *Exit Module*.

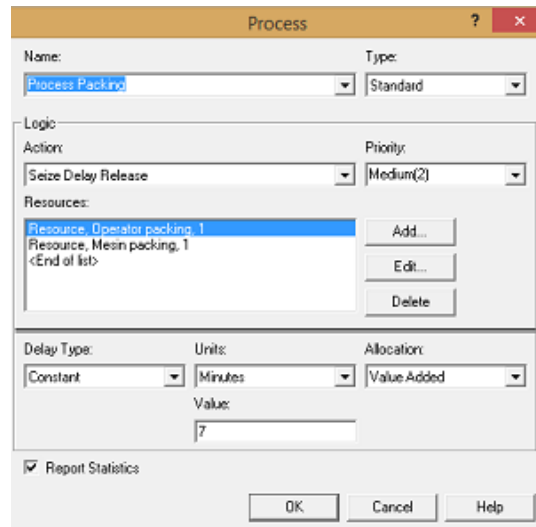
|   | Name                  | Beginning Station  | Next Stations |
|---|-----------------------|--------------------|---------------|
| 1 | Conveyor Pack.Segment | Stasiun Permesinan | 1 rows        |

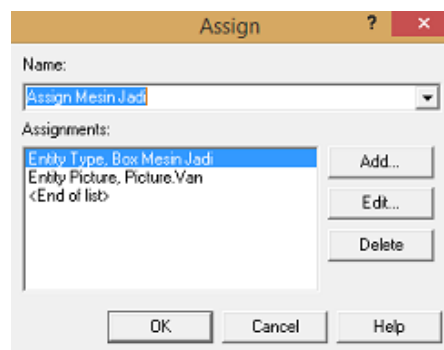
|   | Next Station      | Length |
|---|-------------------|--------|
| 1 | Stasiun Packaging | 10     |



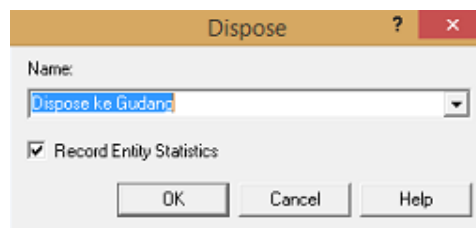
- k. Lalu dilakukan *process packing* mesin yang dilakukan *manual* oleh operator *packaging* dan mesin *packaging* selama 7 menit.



- l. Lalu kita perlu mendeklarasikan entitas barang jadi untuk dapat mengetahui hasil analisis *output* dari produk jadi dengan menggunakan *Assign Module*.



- m. Lalu *Box Mesin Jadi* dikirimkan ke gudang menggunakan *module dispose*.



## Modul 4. Arena Advance 2

### Studi Kasus

PT. Jaya Makmur merupakan sebuah perusahaan yang bergerak dalam bidang produksi sepatu, salah satunya adalah sepatu wanita. Pabrik ini rutin melakukan produksi sepatu setiap harinya dengan menggunakan tenaga kerja sebanyak 6 orang. Tenaga kerja pada PT. Jaya Makmur bekerja selama 8 jam perhari dengan upah ketika sibuk sebanyak 3/jam, ketika idle 1/jam dan setiap kali bekerja 2. Proses pembuatan sepatu pada PT. Jaya Makmur dimulai dari kedatangan 10 pola kap setiap 5 menit sekali. Kemudian pola kap tersebut dipilah berdasarkan polanya, dimana setiap 2 pola yang datang digolongkan sebagai pola kap pria dan 2 pola berikutnya sebagai pola kap wanita. Pola kap pria selanjutnya dipindahkan menuju stasiun sepatu pria sedangkan pola kap wanita akan dijahit dengan sol sepatu.

Sol sepatu yang akan dijahit dengan pola kap wanita tersedia sebanyak 20 sol sepatu yang kemudian akan melalui proses pemotongan menjadi 5 sol sepatu selama  $N(3,0.2)$  MIN yang dilakukan oleh dua operator potong secara bersamaan. Setelah melalui proses pemotongan, sol sepatu akan dijahit dengan pola kap wanita. Proses penjahitan ini berlangsung selama  $N(20,0.25)$  MIN yang dilakukan oleh 2 operator jahit secara bergantian. Dari proses penjahitan, dihasilkan scrap sebanyak 5%. Scrap tersebut kemudian dibawa menuju pembuangan.

Proses terakhir dalam pembuatan sepatu wanita ialah penempelan heels pada sepatu tanpa hak. Heels datang ke dalam stasiun kerja setiap 2 menit sekali dengan jumlah sekali kedatangan sebanyak 100 heels dan kedatangan hanya terjadi sebanyak 100 kali. Proses penempelan heels dilakukan oleh 2 operator hak dan berlangsung selama  $N(2,0.4)$  MIN. Sepatu wanita yang telah jadi kemudian akan dijual seharga 50 per 1 sepatu.

### Algoritma Program

1. Mulai
2. Kedatangan 10 pola kap (menggunakan modul *create*)
3. Pemberian variable antrian pola kap (menggunakan modul *assign*)
4. Apakah antrian pola  $\leq 2$ ? Jika tidak lanjut ke langkah 7 (menggunakan modul *decide*)
5. Pemberian Atribut antri penyatuan sepatu pria (menggunakan modul *assign*)
6. Output menuju stasiun sepatu pria (menggunakan modul *dispose*)
7. Apakah antrian pola  $\leq 4$ ? Jika tidak lanjut ke langkah 9 (menggunakan modul *decide*)
8. Pemberian Atribut antri penyatuan sepatu wanita (menggunakan modul *assign*)
9. Pemberian variable antrian pola = 0 (menggunakan modul *assign*)
10. Kedatangan 20 sol (menggunakan modul *create*)
11. Proses pemotongan sol sepatu menjadi 5 bagian (menggunakan modul *process*)
12. Penggandaan sol sepatu (menggunakan modul *separate*)
13. Penamaan entitas dan pemberian atribut sol wanita (menggunakan modul *assign*)
14. Penggabungan pola kap wanita dan sol wanita (menggunakan modul *batch*)
15. Proses penjahitan sepatu (menggunakan modul *process*)
16. Pemilahan scrap dengan probabilitas 5% dari hasil proses penjahitan. Jika bukan *scrap* lanjut ke langkah 19 (menggunakan modul *decide*)
17. Pemberian nama entitas baru sebagai *scrap* (menggunakan modul *assign*)
18. *Output* entitas *scrap* (menggunakan modul *dispose*)

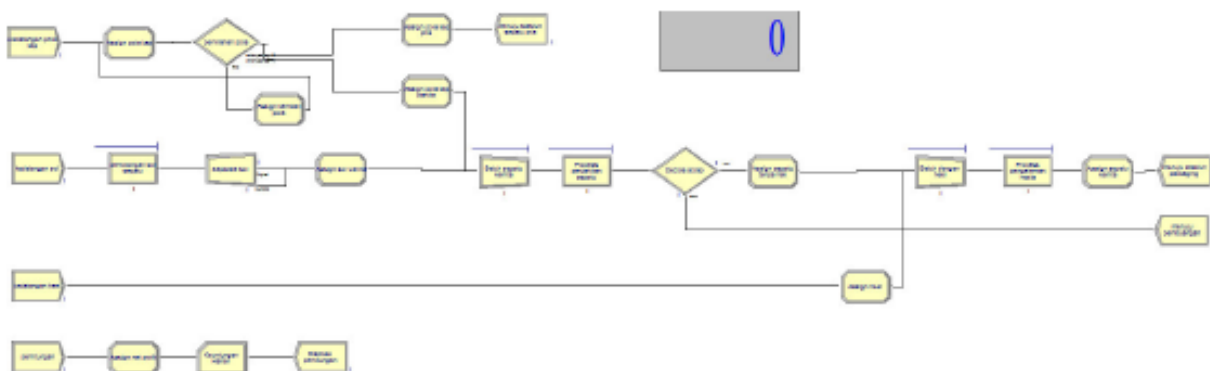
19. Penamaan entitas dan pemberian atribut sepatu wanita (menggunakan modul *assign*)
20. Kedatangan 10 heels (menggunakan modul *create*)
21. Pemberian Atribut antri penggabungan hak (menggunakan modul *assign*)
22. Penggabungan sepatu wanita dan heels (menggunakan modul *batch*)
23. Proses pengeleman heels (menggunakan modul *process*)
24. Pemberian nama entitas baru sepatu heels (menggunakan modul *assign*)
25. *Output* entitas sepatu *heels* (menggunakan modul *dispose*)
26. Kedatangan entitas dummy (menggunakan modul *create*)
27. Perhitungan *net profit* (menggunakan modul *assign*)
28. Mendapat jumlah *net profit* (menggunakan modul *record*)
29. *Output* entitas *dummy* (menggunakan modul *dispose*)
30. Selesai

Hasil pendistribusian waktu setiap proses (setelah melalui input analyzer) digambarkan pada tabel berikut :

| No | Proses                   | Distribusi waktu | Expression       |
|----|--------------------------|------------------|------------------|
| 1  | Kedatangan karet         | Eksponensial     | 5                |
| 2  | Kedatangan plastik       | Eksponensial     | 30               |
| 3  | Pemotongan karet besar   | Normal           | NORM ( 1 , 0.2 ) |
| 4  | Roll                     | Normal           | NORM( 3 , 0.5 )  |
| 5  | Pewarnaan Karet          | Normal           | NORM ( 5 , 0.25) |
| 6  | Pemotongan Bentuk Sandal | Normal           | NORM(1.5 , 0.2)  |
| 7  | Pembuatan Strap          | Normal           | NORM(13, 0.3)    |
| 8  | Pengeleman Strap         | Normal           | NORM(2 , 0.3)    |
| 9  | <i>Packaging</i>         | Normal           | NORM( 1 , 0.3)   |

### Prosedur Praktikum

1. Buat tiga modul ***create***, tujuh modul ***process***, tiga modul ***batch***, tiga modul ***separate***, tiga belas modul ***assign***, dua modul ***decide***, dua modul ***record***, dan lima modul ***dispose***. Gunakan “*zoom in*” untuk melihat dan mengatur *layout* keseluruhan. Jangan menambahkan informasi atau koneksi lain sebelumnya.
2. Berikut merupakan tampilan dari keseluruhan modul tersebut.





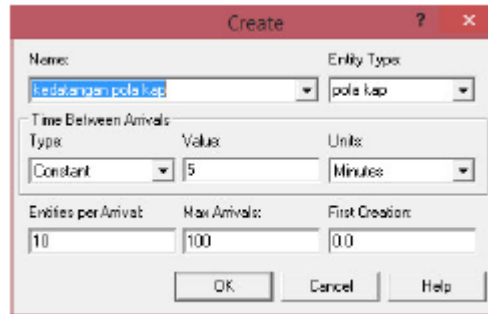
### 3. Untuk modul *create* 1

Untuk modul *create* 1, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

- Isi **Name**: kedatangan pola kap
- Isi **Entity Type**: pola kap
- Pada kolom **Time Between Arrivals** dapat diisi sebagai berikut:

**Type**: Constant, **Expression**: 5, **Units**: Minutes, **Entities Per Arrival**: 10, **Max Arrival**: 100

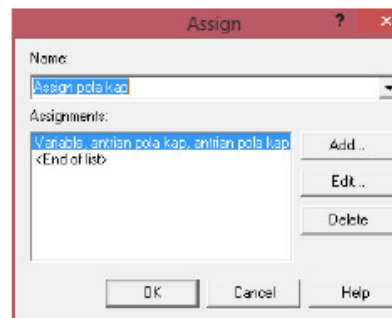
- Kemudian Klik OK.



### 4. Untuk modul *assign* 1

Untuk modul *assign*1, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

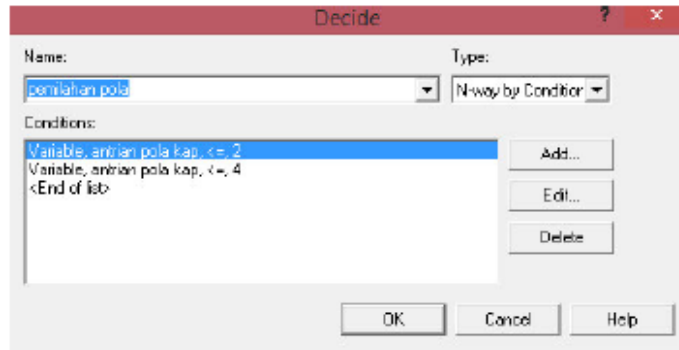
- Name**: Assign pola kap
- Assignments**: Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Variable**. Pada **Variable Name** isikan dengan **antrian pola kap** dengan **New Value** **antrian pola kap+1** Klik OK.



### 5. Untuk modul *decide* 1

Untuk modul *decide* 1, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

- Name**: pemilahan pola
- Type**: N-way by condition
- Condition**: Klik Add Pada kolom **If** pilih **Variable**, lalu isi named dengan **antrian pola kap** kolom **Is** dengan tanda  $\leq$  dan kolom **Value** dengan **2**. Klik OK. Klik Add lagi, kemudian pada kolom **If** pilih **Variable**, lalu isi named dengan **antrian pola kap** kolom **Is** dengan tanda  $\leq$  dan kolom **Value** dengan **4**. Klik OK.



6. Untuk modul *assign 2*

Untuk modul *assign 2*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

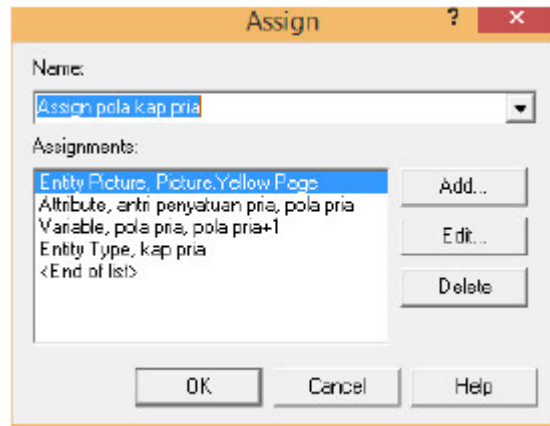
a. **Name:** Assign pola kap pria

b. **Assignments:** Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Entity Picture**. Pada **Entity Picture** isikan dengan **Picture.Yellow Page**. Klik OK.

Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Attribute**. Pada **Attribute Name** isikan dengan **antri penyatuan pria**. Dengan **New Value pola pria**. Klik OK.

Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Variable**. Pada **Variable Name** isikan dengan **pola pria**. Dengan **New Value pola pria+1** Klik OK.

Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Entity Type**. Pada **Entity Type** isikan dengan **kap pria**. Klik OK.



7. Untuk modul *assign 3*

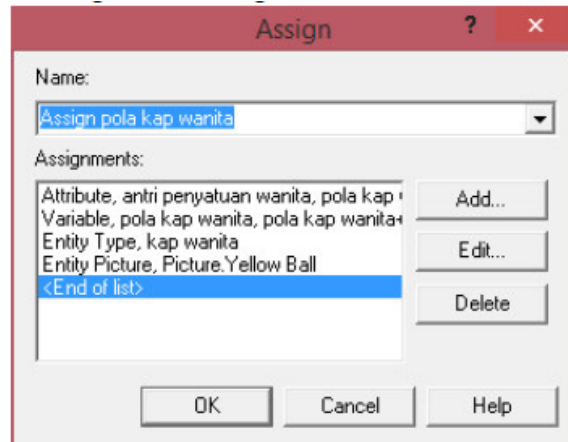
Untuk modul *assign 3*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

a. **Name:** Assign pola kap wanita

b. **Assignments:** Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Attribute**. Pada **Attribute Name** isikan dengan **antri penyatuan wanita**. Dengan **New Value pola kap wanita**. Klik OK.

Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Variable**. Pada **Variable Name** isikan dengan **pola kap wanita**. Dengan **New Value pola kap wanita+1** Klik OK. Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Entity Type**. Pada **Entity Type** isikan dengan **kap wanita**. Klik OK.

Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Entity Picture**. Pada **Entity Picture** isikan dengan **Picture.Yellow Ball**. Klik OK.

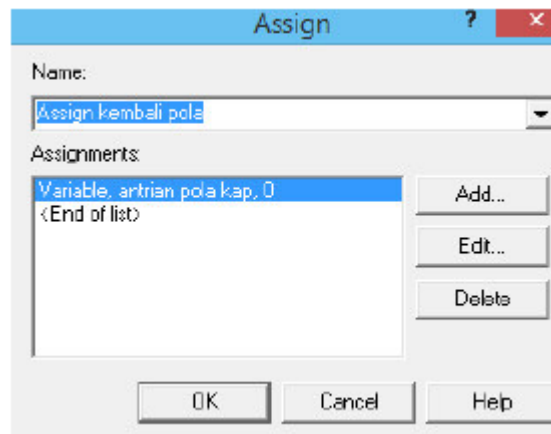


8. Untuk modul *assign 4*

Untuk modul *assign 4*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

a. **Name:** Assign kembali pola

b. **Assignments:** Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Variable**. Pada **Variable Name** isikan dengan **antrian pola kap**. Dengan **New Value 0** Klik OK.

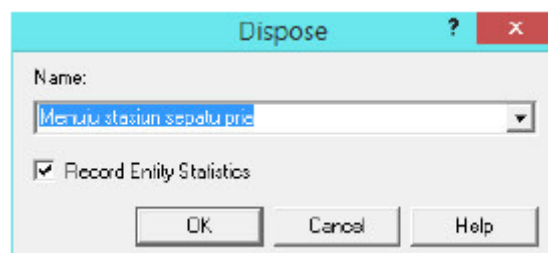


9. Untuk modul *dispose 1*

Untuk modul *dispose 1*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

a. Isikan **Name:** Menuju stasiun sepatu pria

b. Klik OK



10. Untuk modul *create* 2

Untuk modul *create* 2, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

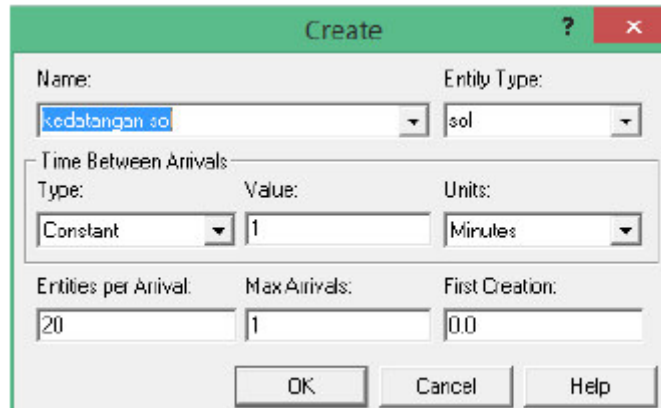
a. Isi **Name**: kedatangan sol

b. Isi **Entity Type**: sol

c. Pada kolom **Time Between Arrivals** dapat diisi sebagai berikut:

**Type**: Constant, **Expression**: 1, **Units**: Minutes, **Entities Per Arrival**: 20, **Max Arrival**: 1

d. Kemudian Klik OK.



11. Untuk modul *process* 1

Untuk modul *process* 1, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

a. Isi **Name**: pemotongan sol sepatu

b. Pada kolom **Logic**, diisi sebagai berikut: **Action**: Seize Delay Release, **Priority**: Medium(2)

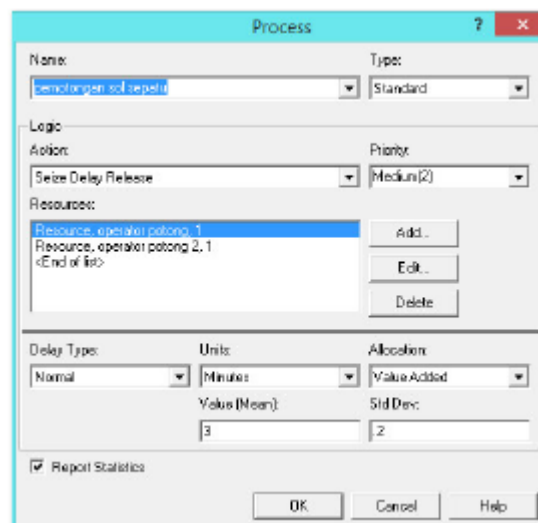
**Resources**: Klik **Add**, lalu akan keluar kotak dialog **Resources**. Pada pilihan **Type**, pilih **Resource**. Pada **Resource Name** isikan dengan operator potong dengan quantity 1. Klik OK.

c. **Delay Type**: Normal

**Units**: Minutes **Allocation**: Value Added **Value (Mean)**: 3 **Std Dev**: 0.2

d. Centang **Report Statistic**

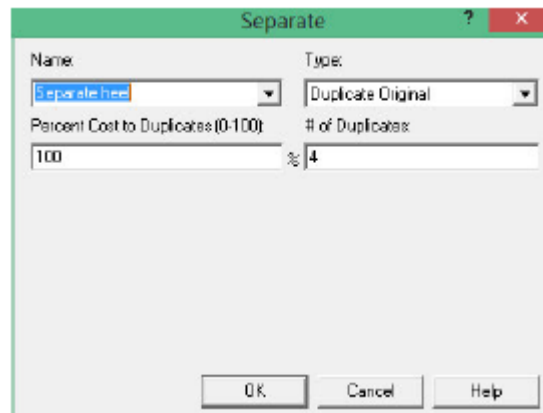
e. Klik **OK**



12. Untuk modul *Separate* 1

Untuk modul *separate* 1, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

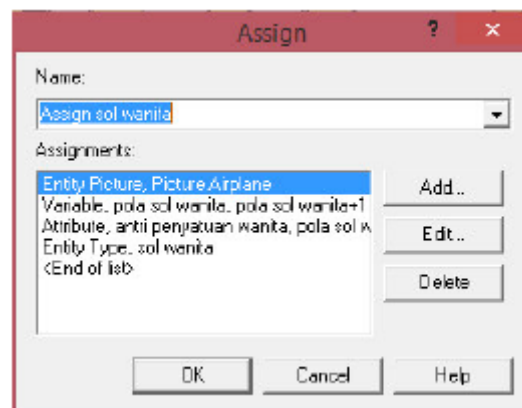
- a. **Name:** *Separate heel*
- b. **Type:** *Duplicate Original*
- c. **Percent Cost to duplicates (0-100):**100
- d. **# of Duplicates:** 4
- e. Klik **OK**



13. Untuk modul *assign* 5

Untuk modul *assign* 5, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

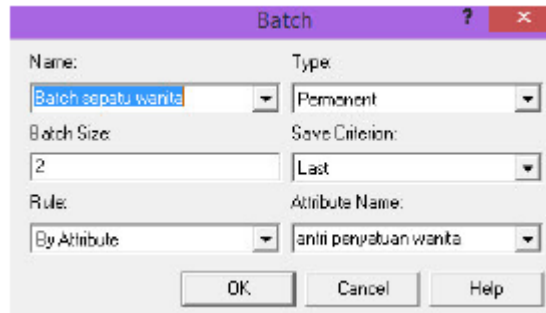
- a. **Name:** *Assign sol wanita*
- b. **Assignments:** Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Entity Picture**. Pada **Entity Picture** isikan dengan **Picture.Airplane**. Klik **OK**.  
Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Variable**. Pada **Variable Name** isikan dengan **pola sol wanita**. Dengan **New Value** **pola sol wanita+1** Klik **OK**. Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Attribute**. Pada **Attribute Name** isikan dengan **antri penyatuan wanita**. Dengan **New Value** **pola sol wanita**. Klik **OK**.  
Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Entity Type**. Pada **Entity Type** isikan dengan **sol wanita**. Klik **OK**.



14. Untuk modul *batch* 1

Untuk modul *batch* 1, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog berikut dengan isian sebagai berikut:

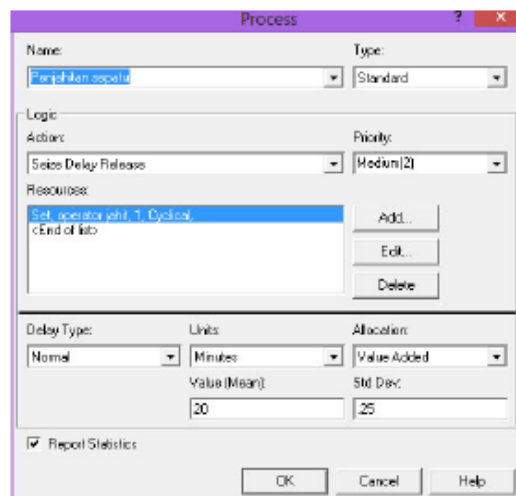
- a. **Name:** *Batch* sepatu wanita
- b. **Type:** *Permanent*
- c. **Batch Size:** 2
- d. **Save Criterion:** *Last*
- e. **Rule:** *By Attribute*
- f. **Attribute Name:** antri penyatuan wanita
- g. Klik **OK**



15. Untuk modul *process* 2

Untuk modul *process* 2, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

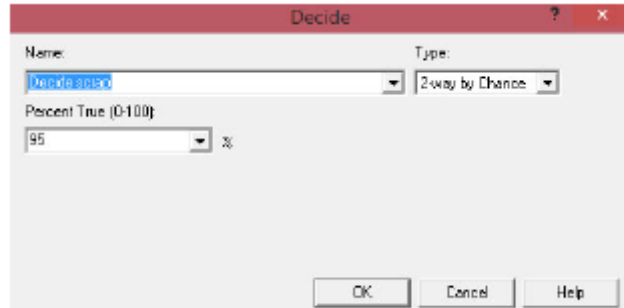
- a. Isi **Name:** Penjahitan sepatu
- b. Pada kolom **Logic**, diisi sebagai berikut: **Action:** Seize Delay Release, **Priority:** Medium(2)  
**Resources:** Klik **Add**, lalu akan keluar, kotak dialog **Resources**. Pada pilihan **Type**, pilih **Set**. Pada **Set Name** isikan dengan **operator jahit** dengan **Quantity** 1. Pada **Selection Rule** isikan dengan **Cyclical**. Klik **OK**.
- c. **Delay Type:** Normal, **Units:** Minutes **Allocation:Value Added Value (Mean):** 20 **Std Dev:** 0.25
- d. Centang **Report Statistic**
- e. Klik **OK**



16. Untuk modul decide 2

Untuk modul decide 2, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

- a. **Name:** Decide scrap
- b. **Type:** 2-way by chance
- c. **Percent True (0-100):** 95
- d. Klik **OK**.



17. Untuk modul assign 6

Untuk modul assign 6, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

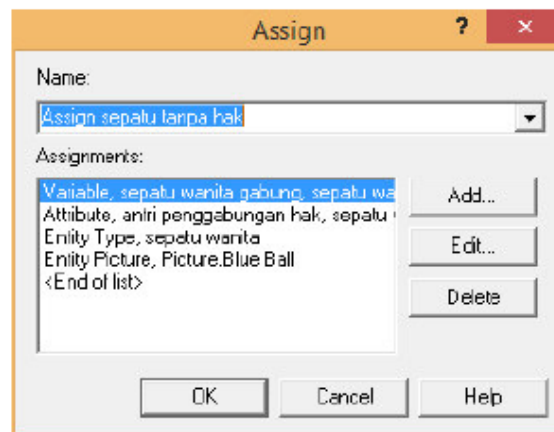
- a. **Name:** Assign sepatu tanpa hak
- b. **Assignments:** Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Variable**. Pada **Variable Name** isikan dengan **sepatu wanita gabung**. Dengan **New Value** **sepatu wanita gabung+1**. Klik **OK**.

Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Attribute**. Pada **Attribute Name** isikan dengan **antri penggabungan hak**. Dengan **New Value** diisi **sepatu wanita gabung**.

Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Entity Type**. Pada **Entity Type** isikan dengan **sepatu wanita**. Klik **OK**.

Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Entity Picture**. Pada **Entity Picture** isikan dengan **Picture.Blue Ball**. Klik **OK**.

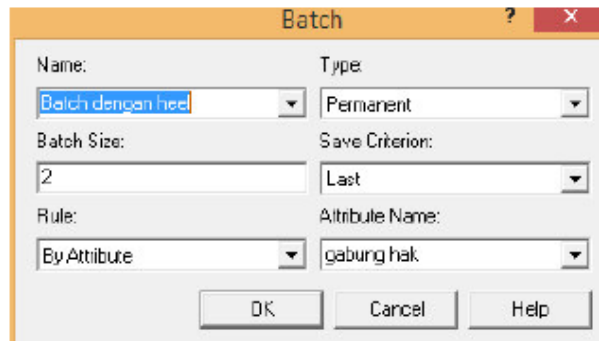
- c. Klik **OK**.



18. Untuk modul *batch 2*

Untuk modul *batch 2*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog berikut dengan isian sebagai berikut:

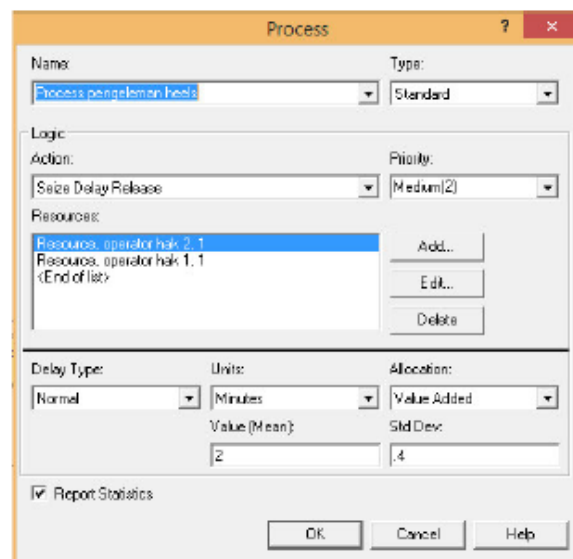
- a. **Name:** *Batch dengan heel*
- b. **Type:** *Permanent*
- c. **Batch Size:** *2*
- d. **Save Criterion:** *Last*
- e. **Rule:** *By Attribute*
- f. **Attribute Name:** *gabung hak*
- g. Klik **OK**



19. Untuk modul *process 3*

Untuk modul *process 3*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

- a. Isi **Name:** *Process pengeleman heels*
- b. Pada kolom **Logic**, diisi sebagai berikut: **Action:** *Seize Delay Release*, **Priority:** *Medium(2)*  
**Resources:** Klik **Add**, lalu akan keluar kotak dialog **Resources**. Pada pilihan **Type**, pilih **Resource**. Pada **Resource Name** isikan dengan **operator hak 1**, dengan quantity **1**. Klik **OK**.  
Klik **Add** lagi, lalu akan keluar kotak dialog **Resources**. Pada pilihan **Type**, pilih **Resource**. Pada **Resource Name** isikan dengan **operator hak 2**, dengan quantity **1**. Klik **OK**.
- c. **Delay Type:** *Normal*, **Units:** *Minutes* **Allocation:** *Value Added* **Value (Mean):** *2* **Std Dev:** *0.4*
- d. Centang **Report Statistic**
- e. Klik **OK**





20. Untuk modul *assign 7*

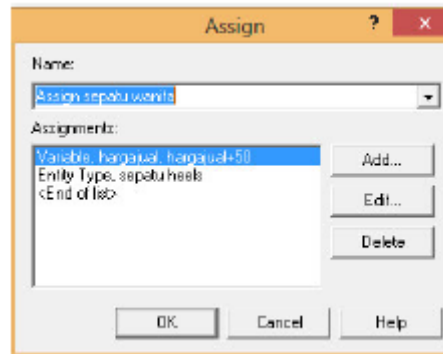
Untuk modul *assign7*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

a. **Name:** Assign sepatu wanita

b. **Assignments:** Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Variable**. Pada **Variable Name** isikan dengan **hargajual**. Dengan **New Value** **hargajual+50** Klik OK.

Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Entity Type**. Pada **Entity Type** isikan dengan **sepatu heels**. Klik OK.

c. Klik **OK**.

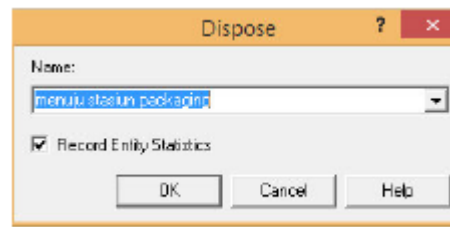


21. Untuk modul *dispose 2*

Untuk modul *dispose 2*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

a. Isikan **Name:** Menuju stasiun packaging

b. Klik **OK**



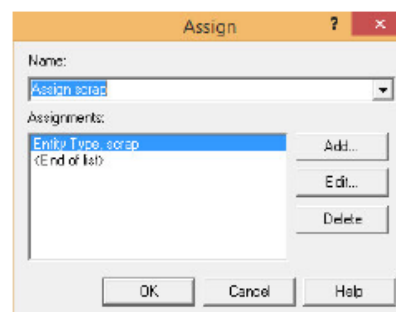
22. Untuk modul *assign 8*

Untuk modul *assign 8*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

a. **Name:** Assign scrap

b. **Assignments:** Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Entity Type**. Pada **Entity Type** isikan dengan **scrap**. Klik OK.

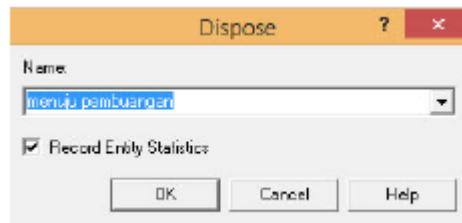
c. Klik **OK**.



23. Untuk modul *dispose* 3

Untuk modul *dispose* 3, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

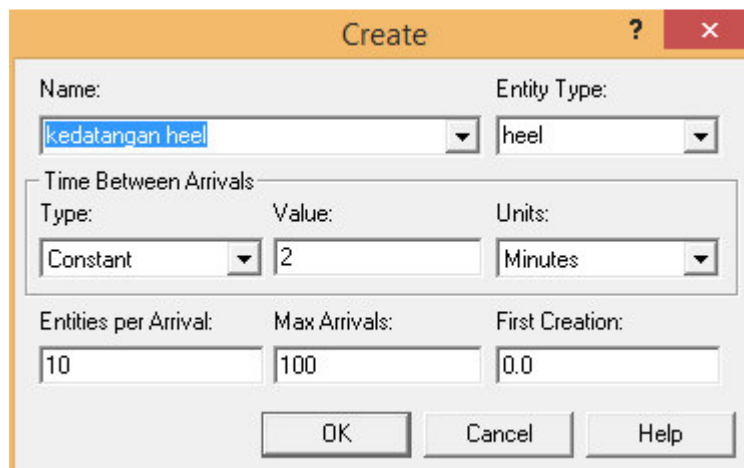
- Isikan **Name**: menuju pembuangan
- Klik OK



24. Untuk modul *create* 3

Untuk modul *create* 3, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

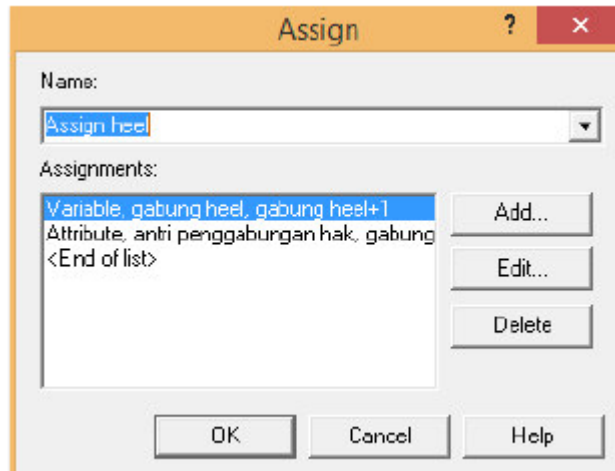
- Isi **Name**: kedatangan heel
- Isi **Entity Type**: heel
- Pada kolom **Time Between Arrivals** dapat diisi sebagai berikut: **Type**: Constant, **Expression**: 2, **Units**: Minutes, **Entities Per Arrival**: 10, **Max Arrival**: 100
- Kemudian Klik OK.



25. Untuk modul *assign* 9

Untuk modul *assign* 9, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

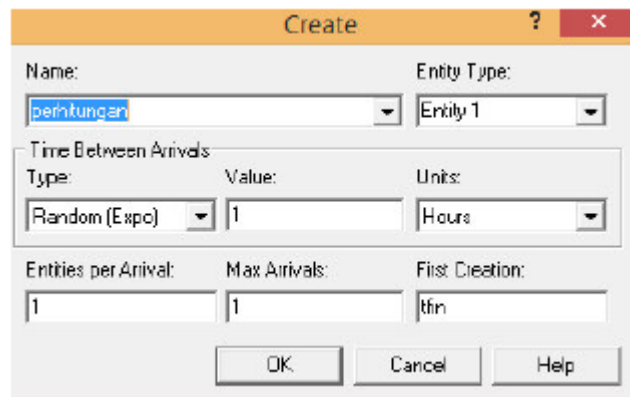
- Name**: Assign heel
- Assignments**: Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Variable**. Pada **Variable Name** isikan dengan **gabung heel**. Dengan **New Value** **gabung heel+1**  
Klik OK.  
Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Attribute**. Pada **Attribute Name** isikan dengan **antri penggabungan hak**. Pada **New Value** diisi **gabung heel** Klik OK.
- Klik **OK**.



26. Untuk modul *create 4*

Untuk modul *create 4*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

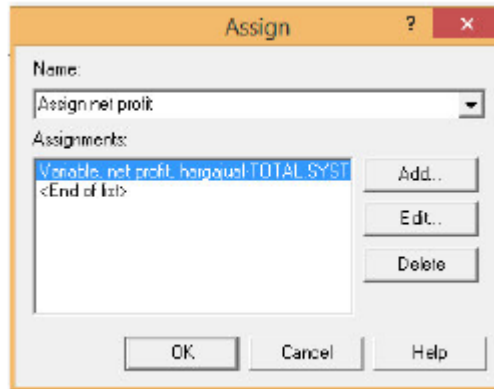
- Isi **Name**: perhitungan
- Isi **Entity Type**: Entity 1
- Pada kolom **Time Between Arrivals** dapat diisi sebagai berikut: **Type**: *Random (Expo)*, **Value**: 1, **Units**: *Minutes*, **Entities Per Arrival**: 1, **Max Arrival**: 1
- Kemudian Klik OK.



27. Untuk modul *assign 10*

Untuk modul *assign 10*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog di bawah ini dengan isian sebagai berikut:

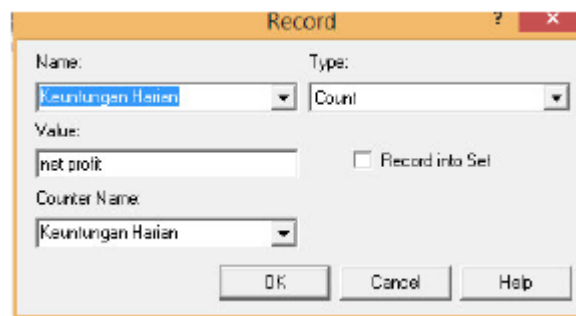
- Name**: Assign net profit
- Assignments**: Klik **Add**, lalu akan keluar kotak dialog **Assignments**. Pada pilihan **Type**, pilih **Variable**. Pada **Variable Name** isikan dengan **net profit**. Dengan **New Value** **hargajual- TOTAL.SYSTEMCOST** Klik OK.
- Klik **OK**.



28. Untuk modul *record 1*

Untuk modul *record 1*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

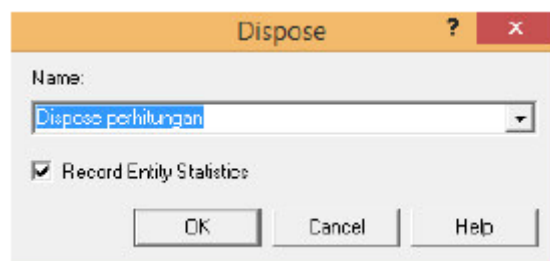
- Name:** Keuntungan Harian
- Type:** Count
- Value:** net profit
- Counter Name:** Keuntungan Harian



29. Untuk modul *dispose 4*

Untuk modul *dispose 4*, klik dua kali pada modul tersebut sehingga terlihat kotak dialog dan lengkapi kotak dialog dibawah ini dengan isian sebagai berikut:

- Isikan **Name:** Dispose perhitungan
- Klik OK



30. Mendefinisikan biaya pada *resources*, pada *project bar* terdapat *basic process panel*, kemudian klik *resources*, isikan biaya operator sesuai dengan biaya yang ditentukan untuk setiap operator dan mesin yang ada. Setelah itu mendefinisikan jam kerja *resource* pada *schedule spread sheet*, dimana pada *project bar* terdapat *basic process panel*, kemudian klik *schedule*, isikan *Name* sesuai dengan gambar dibawah, kemudian klik *durations* maka akan muncul gambar seperti dibawah isikan sesuai *capacity*

resources ☐ kosongkan pada kotak kelima sesuai gambar. Lalu klik resources di basic process panel, klik tab type ☐Based on Schedule. Lalu klik tab Schedule Name ☐“Schedule 1”.

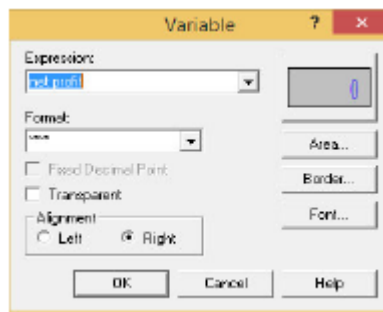
| Resource - Basic Process |                   |                |          |             |             |         |                |          |                   |
|--------------------------|-------------------|----------------|----------|-------------|-------------|---------|----------------|----------|-------------------|
|                          | Name              | Type           | Capacity | Busy / Hour | Kilo / Hour | Per Use | State/Set Name | Failures | Report Statistics |
| 1                        | operator jahit    | Fixed Capacity | 1        | 3           | 1           | 2       |                | 0 rows   | ☑                 |
| 2                        | operator jahit2   | Fixed Capacity | 1        | 3           | 1           | 2       |                | 0 rows   | ☑                 |
| 3                        | operator jahit 2  | Fixed Capacity | 1        | 3           | 1           | 2       |                | 0 rows   | ☑                 |
| 4                        | operator jahit 1  | Fixed Capacity | 1        | 3           | 1           | 2       |                | 0 rows   | ☑                 |
| 5                        | operator potong   | Fixed Capacity | 1        | 3           | 1           | 2       |                | 0 rows   | ☑                 |
| 6                        | operator potong 2 | Fixed Capacity | 1        | 3           | 1           | 2       |                | 0 rows   | ☑                 |

|   | Name      | Type     | Time Units | Durations |
|---|-----------|----------|------------|-----------|
| 1 | jam kerja | Capacity | Hours      | 4 rows    |



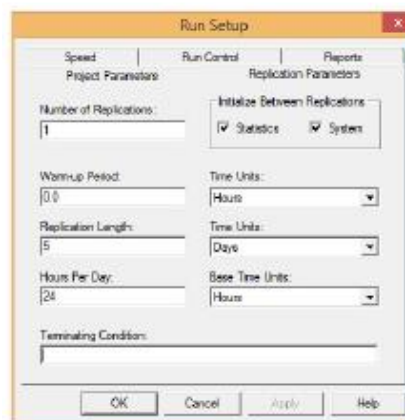
31. Untuk melihat net profit, maka pilih variable pada toolbar, isikan:

- Expression:** net profit
- Format:** \*\*\*\*
- Klik OK



32. Setelah itu pada menu bar klik ribbon run setup. Pada Replication Parameters, isikan:

- Number of Replication:** 1
- Replication Length:** 5
- Hours Per Day:** 24
- Time Units :** days
- Base Time Units:** hours
- Klik OK



32. Klik GO untuk menjalankan model sistem antrian