



# **UNIVERSIDADE SÃO JUDAS TADEU**

**FACULDADE DE TECNOLOGIA E CIENCIAS EXATAS  
CIÊNCIA DA COMPUTAÇÃO  
SISTEMAS DE INFORMAÇÃO  
ENGENHARIA DE PRODUÇÃO**

## **Usando o ARENA em Simulação**

**FERNANDO MORI**

<http://sites.google.com/site/fmoripro/>

# ÍNDICE

" AULA 1 .....	Pag. 3
" AULA 2 .....	Pag. 182
" AULA 3 .....	Pag. 208
" AULA 4 .....	Pag. 228
" AULA 5 .....	Pag. 246
" AULA 6 .....	Pag. 257
" AULA 7 .....	Pag. 273
" AULA 8 .....	Pag. 298
" AULA 9 .....	Pag. 314
" AULA 10 .....	Pag. 337

# AULA 1

- ” Introdução Teórica
- ” Construção do Primeiro Modelo Arena :  
Modelo do Posto de Lavagem de Veículos.

# Modelagem

” O quê está sendo modelado?

- simulações computacionais lidam com modelos de *sistemas* (facilidades ou processos), tais como:
  - ” sistema de manufatura com máquinas, pessoas, rede de transporte, correias transportadoras e espaço para armazenamento;
  - ” agências bancárias ou algo do gênero, com diferentes tipos de usuários, servidores e facilidades, tais como caixas, máquinas automáticas, atendentes e caixas de auto-serviço;
  - ” redes de fábricas, armazéns e vias de acesso;
  - ” sala de emergência de um hospital, com pessoal, salas, equipamentos, suprimentos e transporte de pacientes;
  - ” rede de computadores com servidoras, clientes, unidades de disco, de fita, impressoras, serviços de rede e operadores;

# Modelagem (cont.)

- “ O quê está sendo modelado?
  - “ sistema rodoviário com segmentos de estrada, trevos, controladores e tráfego;
  - “ escritório de reclamação de seguros, onde formulários são recebidos, conferidos, copiados, preenchidos e remetidos para pessoas;
  - “ planta de produção de produtos químicos, com tanques de armazenagem, reatores e rodovias para caminhões tanques nas quais o produto final é transportado;
  - “ restaurante tipo *fast-food*;
  - “ parque temático, com controle de estoque, bilheteiros e serviço de atendimento aos consumidores;
  - “ *etc.*

# Modelagem (cont.)

- ” Por quê não, simplesmente, brincar com o sistema?
  - . em alguns casos é possível:
    - ” podem-se testar diferentes configurações para os sinais de tráfego, até encontrar aquela que minimiza os congestionamentos;
    - ” um gerente de supermercado pode tentar diferentes políticas de controle de estoque e atribuição de atendentes, até encontrar a mais lucrativa e que disponibiliza o melhor atendimento;
    - ” um sistema computacional pode ser organizado com diferentes configurações de redes e prioridades de atendimento, até prover o melhor serviço.

# Modelagem (cont.)

” Algumas vezes você não pode (ou deve) brincar como o sistema ...

- . você não poderia experimentar diferentes *layouts* de uma fábrica se a mesma não foi ainda construída;
- . mesmo que ela exista, talvez seja muito caro fazê-lo;
- . seria difícil submeter o dobro de usuários a um banco, para ver o que aconteceria se uma agência vizinha fechasse;
- . tentar um novo sistema de embarque, poderia fazer muitos usuários perder seus vôos, se houver imprevistos;
- . efetuar testes em uma sala de emergência de um hospital, certamente não seria admissível.

# Modelagem (cont.)

- ” Com simulação, você pode testar muitas idéias no modelo:
  - . cometa erros no computador, onde eles não contam, em vez de cometê-los no mundo real, onde eles contam.
- ” Tipos de modelos:
  - . físicos:
    - ” modelos reduzidos, modelos em tamanho natural, simulação de salas/cabines de controle, simuladores de vôo.
  - . lógicos/matemáticos:
    - ” conjunto de aproximações e suposições, tanto estrutural quanto quantitativa;
    - ” se o modelo é uma boa aproximação, você pode aprender sobre o sistema, de uma forma barata, fácil e rápida.

# Modelagem (cont.)

” O quê fazer com o modelo lógico?

- . Se é simples o bastante, use matemática tradicional (teoria de filas, equações diferenciais, programação linear) para conseguir “respostas”:
  - ” bom, pois você pode obter soluções “exatas” para o modelo;
  - ” mas pode ter sofrido uma sobre-simplificação (validade?).

” Muitos sistemas complexos necessitam de modelos também complexos

- . dificuldade ou impossibilidade de resolução;
- . simulação pode ser a única saída.

# Simulação Computacional

- ” métodos para estudo de uma ampla variedade de modelos de sistemas do mundo real:
  - . utilizam computadores para avaliação numérica;
  - . utilizam *softwares* para imitar a operação do sistema e suas características, ao longo do tempo.
- ” na prática, é o processo de projeto e criação de modelos de sistemas computadorizados, bem como de execução de experimentos;
- ” capacidade: aplicação em sistemas complexos;
- ” tolera modelos complexos.

# Vantagens

- ” Permite incerteza e não estacionariedade
  - . a única certeza é que tudo é incerto;
  - . perigoso ignorar a variabilidade do sistema;
- ” Avanços na relação custo/capacidade computacional
  - . estima-se que 75% da capacidade computacional seja utilizada para simulações;
  - . máquinas dedicadas (p.e. controladores em tempo real).
- ” Avanços nos softwares para simulação
  - . facilidade de uso;
  - . menos restritivos nas modelagens;
  - . capacidade de projeto e análise estatística.

# As Más Noticias

- ” Não gera respostas exatas, mas estimativas aproximadas:
  - . também verdade para muitos outros métodos modernos;
  - . pode limitar o erro ao erro de truncamento da máquina.
- ” Gera respostas aleatórias de simulações estocásticas
  - . planejamento de experimentos e análise estatística das simulações;
  - . problema: métodos estatísticos “padrões” raramente se aplicam.

# Tipos Diferentes de Simulação

- ” Estática versus Dinâmica
  - . o tempo desempenha um papel importante no modelo?
- ” Tempo contínuo versus discreto
  - . pode o estado mudar continuamente ou simplesmente em intervalos discretos de tempo?
- ” Deterministico versus Estocástico
  - . há certeza quanto a tudo, ou não?
- ” A maioria dos modelos são:
  - . *dinâmicos, discretos e estocásticos!*

# Simulando em Computadores

- “ Linguagens de uso geral (FORTRAN)
  - . tedioso, baixo nível, sujeito a erros;
  - . entretanto, flexibilidade quase que total!
- “ Pacotes
  - . subrotinas para processamento de listas, estatísticas e avanço de tempo;
  - . amplamente distribuídas e também amplamente modificadas.
- “ Planilhas
  - . usualmente, somente para modelos estáticos;
  - . cenários financeiros, etc.

# Simulando em Computadores (cont.)

## ” Linguagens de simulação

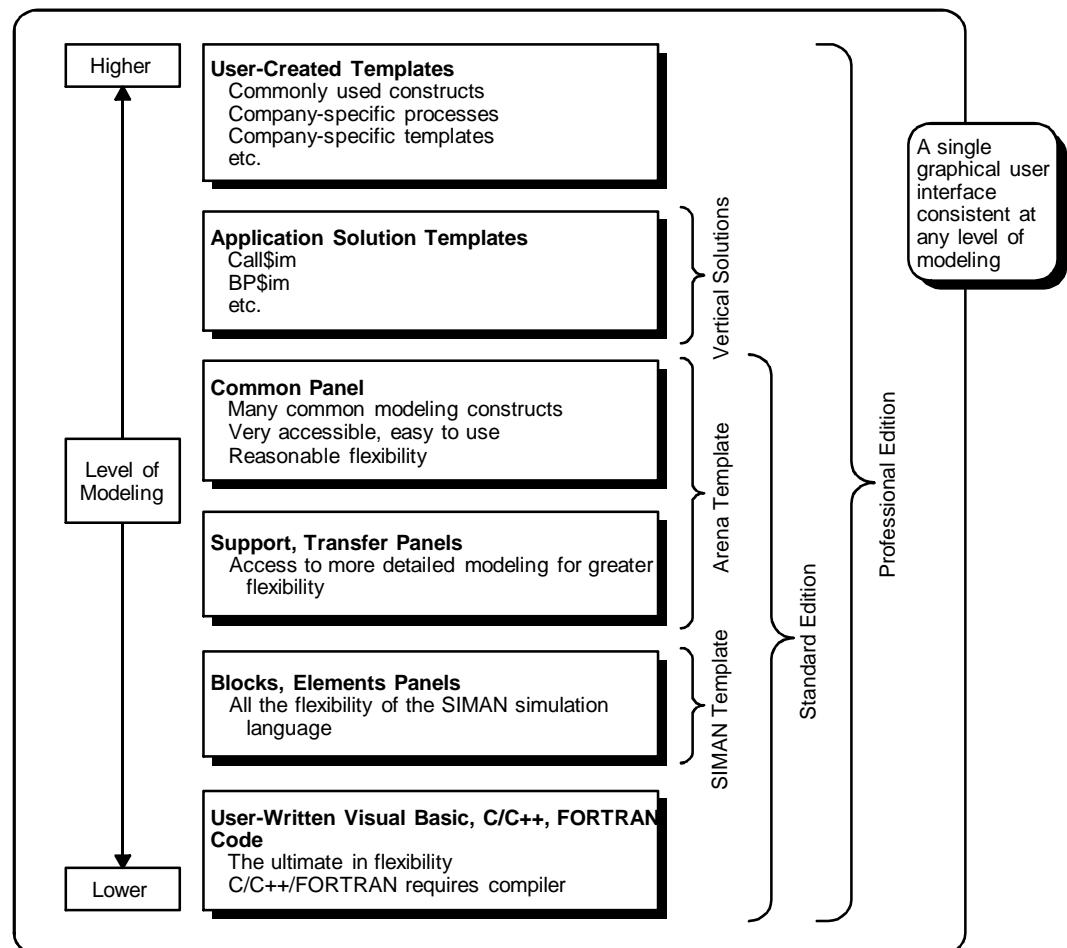
- . GPSS, SIMSCRIPT, SLAM, SIMAN;
- . populares, de uso difundido atualmente;
- . certo investimento para aprendizagem das características, utilização, sintaxe.

## ” Simuladores de alto-nível

- . facilidade de uso, interface gráfica;
- . pode ser restrito a certos domínios (manufatura, telecomunicações, etc.);
- . flexibilidade limitada (validade do modelo?).

# Onde o Arena se Encaixa

- „ Estrutura Hierárquica
  - múltiplos níveis de modelagem
  - pode combinar diferentes níveis no mesmo modelo
  - frequentemente, inicia em alto nível, descendo à medida da necessidade
- „ Vantagem da facilidade de uso dos simuladores sem perda de flexibilidade



# Quando Simulações São Utilizadas

- ” Utilizações da simulação tem evoluído com o hardware e software
- ” Primeiros anos (1950s-1960s)
  - . muito cara, ferramenta de uso especializado;
  - . necessitava computadores grandes e treinamento especial;
  - . primordialmente em FORTRAN (ou mesmo em linguagem Assembler);
  - . custo de processamento de cerca de US\$ 1000/hora, para uma máquina inferior a um computador do tipo 286.

# Quando Simulações São Utilizadas (cont.)

- ” Anos formadores (1970s-início dos 1980s)
  - . computadores barateiam e ficam mais rápidos;
  - . reconhecimento mais amplo do valor da simulação;
  - . melhoria dos softwares de simulação, mas ainda havia linguagens a serem apreendidas, digitadas e processadas em lote;
  - . freqüentemente utilizadas para limpar desastres nas indústrias automobilística e aeroespaciais:
    - ” planta automobilística: demanda por um certo modelo;
    - ” linha de montagem com desempenho insatisfatório;
    - ” simulação, identificação do problema;
    - ” mas a demanda reduz-seô a simulação foi tarde demais

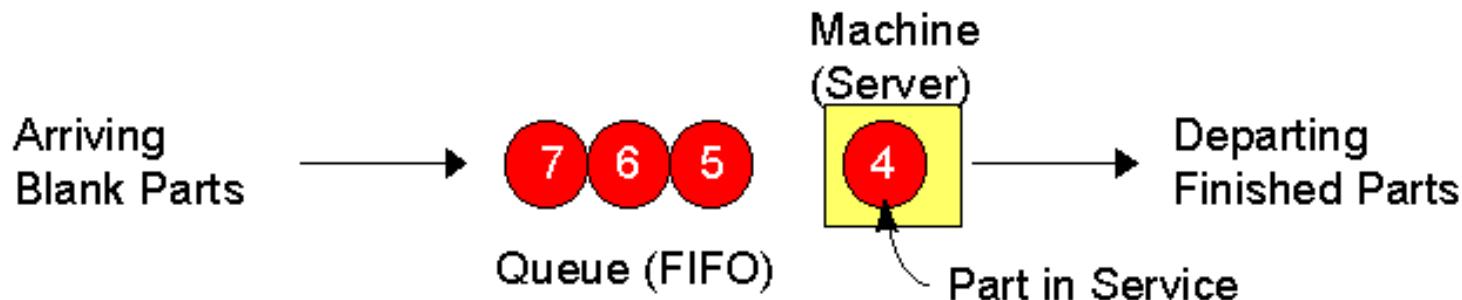
# Quando Simulações São Utilizadas (cont.)

- ” Passado recente (início dos 1980s)
  - . capacidade dos micromarcadores;
  - . software expandem permitindo animação;
  - . aceitação ainda mais ampla em outras áreas:
    - ” aplicações tradicionais na manufatura;
    - ” serviços;
    - ” sistema de saúde;
    - ” negócios;
  - . ainda restrito a grande firmas;

# Quando Simulações São Utilizadas (cont.)

- “ O presente
  - . proliferando em firmas menores;
  - . se tornando uma ferramenta padrão;
  - . sendo utilizadas mais cedo, ainda na fase de projeto;
  - . controle em tempo real.
- “ O futuro
  - . exploração da interoperacionalidade entre sistemas operacionais;
  - . modelos especialmente construídos para indústrias, firmas;
  - . planejamento estatístico e análise estatística automatizados.

# Conceitos Fundamentais: Um Sistema Simplificado



- “ objetivo geral:
  - . estimar a produção esperada;
  - . tempo na fila, tamanho da fila, proporção de tempo no qual a máquina está ocupada;
- “ bom senso
  - . seja consistente (partes de tempo);
  - . seja razoável (interpretação, arredondamentos).

# Especificações do Modelo

- ” inicialmente (tempo 0), sistema vazio e desocupado;
- ” unidade de tempo: minutos;
- ” tempos de chegada: 0,00, 6,84, 9,24, 11,94, 14,53;
  - . *tempo entre chegadas*: 6,84, 2,40, 2,70, 2,59, 0,73;
- ” tempos de serviço: 4,58, 2,96, 5,86, 3,21, 3,11;
- ” parar após transcorridos 15 minutos de simulação.

# Objetivos do Estudo: Medidas de Desempenho

- “ número total de partes produzidas P;
- “ tempo médio de espera das partes na fila:

$$\frac{\sum_{i=1}^N D_i}{N}$$

$N$  = número de partes produzidas;  
 $D_i$  = tempo de espera da i-ésima unidade;  
conhecidos:       $D_1 = 0$  (por quê?)  
                         $N \geq 1$  (por quê?)

- “ tempo máximo de espera das partes na fila:

$$\max_{i=1,\dots,N} D_i$$

# Objetivos do Estudo: Medidas de Desempenho (cont.)

- “ número médio de partes na fila:

$$\frac{\int_0^{15} Q(t) dt}{15}$$

$Q(t)$  = número de partes na fila  
no tempo  $t$

- “ número máximo de partes na fila:

$$\max_{0 \leq t \leq 15} Q(t)$$

- “ tempo médio e máximo de ciclo das partes (tempo no sistema)

$$\sum_{i=1}^P F_i / P, \quad \max_{i=1, \dots, P} F_i \quad F_i = \text{tempo ciclo } i\text{-ésima unidade}$$

# Objetivos do Estudo: Medidas de Desempenho (cont.)

” utilização da máquina (proporção de tempo de ocupação):

$$\frac{\int_0^{15} B(t) dt}{15}$$

$$B(t) = \begin{cases} 1 & \text{if the machine is busy at time } t \\ 0 & \text{if the machine is idle at time } t \end{cases}$$

” muitas outras possíveis (sobrecarga de informação?).

# Opções de Análise

## ” palpite

- . tempo médio entre chegadas = 3,05 min
- . tempo médio de serviço = 3,94 min
- . tamanho da fila tende a explodir (talvez não em apenas 15 min);
- . se o tempo médio entre chegadas fosse maior que o tempo médio de serviço e estes tivessem variância nula, nunca haveria filas;
- . a realidade: entre estes dois extremos;
- . palpite tem suas limitações!

# Opções de Análise (cont.)

- “ teoria de filas
  - . requer hipóteses adicionais a respeito do modelo;
  - . modelo popular e simples, a fila **M/M/1**
    - “ tempo entre chegadas ~ exponencial
    - “ tempo de serviço ~ exponencial e independente das chegadas
    - “  $E(\text{serviço}) < E(\text{chegadas})$
    - “ regime permanente (tempo muito longo, eterno);
    - “ resultados analíticos exatos, p.e., tempo médio de espera na fila:
  - . problemas: validade, estimativa das médias, janela de tempo;
  - . freqüentemente, útil, como primeira aproximação.

$$\frac{\mu_S^2}{\mu_A - \mu_S} \quad \begin{aligned} \mu_S &= E(\text{Service Time}) \\ \mu_A &= E(\text{Interarrival Time}) \end{aligned}$$

# Simulação Mecânica

- ” operações (chegadas, tempos de serviço) individuais ocorrerão exatamente como na realidade;
- ” movimentos e mudanças ocorrem no tempo correto, na ordem correta;
- ” interação entre as diferentes partes;
- ” instalação de “observadores” para obtenção de medidas de desempenho;
- ” abordagem “força bruta”;
- ” nada de misterioso ou refinado
  - . mas muitos detalhes e anotações;
  - . software para simulação pode fazê-lo para você.

# Partes de Um Modelo de Simulação

## ” entidades

- “agentes” que se movem, mudam de estado, afetam e são afetados por outras entidades;
- *objetos dinâmicos* • são criados, se movem, vão embora (talvez);
- usualmente, representam coisas “reais”:
  - ” nosso modelo, entidades são partes;
- pode haver entidades “imaginárias” para modelagem de “travessuras”:
  - ” demônios/anjos que surgem e danificam partes/deixam inoperantes;
  - ” usualmente, apresentam múltiplas realizações;
- pode haver diferentes entidades concorrentes.

# Partes de Um Modelo de Simulação

(cont.)

## ” atributos

- . características de todas as entidades: descrevem, diferenciam;
- . todas as entidades possuem atributos básicos, mas diferentes valores para diferentes atributos, p.e.:
  - ” tempo de chegada;
  - ” prazo para entrega;
  - ” prioridade;
  - ” cor;
- . valores são característicos de uma entidade específica;
- . semelhantes às variáveis locais;
- . no Arena, algumas são automáticas, outras, você define.

# Partes de Um Modelo de Simulação

(cont.)

## “variáveis (globais)

- . reflete a característica de um sistema, independente das entidades;
- . cujo nome e valor só existe um para o modelo inteiro;
- . não acopladas às entidades;
- . podem ser acessadas e modificadas pelas entidades;
  - ” tempo de viagem entre estações;
  - ” número de partes no sistema;
  - ” cronômetro da simulação;
- . escritas nas “paredes” do sistema, não nas partes;
- . algumas, próprias no Arena, mas outras podem ser definidas por você.

# Partes de Um Modelo de Simulação (cont.)

- ” recursos
  - . sobre o que as entidades competem:
    - ” pessoal,
    - ” equipamentos,
    - ” espaço físico;
  - . as entidades tomam um recurso, utilizam-no e liberam-no;
  - . pense em um recurso como sendo atribuído a uma entidade, em lugar de uma entidade pertencendo a um recurso;
  - . “um” recurso pode ter diversas unidades de capacidade:
    - ” assentos em uma mesa de restaurante;
  - . número de unidades de um recurso podem mudar ao longo da simulação.

# Partes de Um Modelo de Simulação (cont.)

## „ filas

- locais de espera das entidades, quando não podem se mover (possivelmente porque o recurso que pretendem tomar não está disponível);
- têm nomes, freqüentemente ligados ao recurso respectivo;
- podem ter capacidade finita para modelar limitação de espaço—tem que ser modelado o que fazer quando uma entidade surgir para entrar em uma fila que já estiver cheia;
- usualmente, monitora-se o seu tamanho e tempo de espera.

# Partes de Um Modelo de Simulação (cont.)

- ” acumuladores de estatísticas
  - . variáveis que “monitoram” o que está ocorrendo;
  - . dependentes das medidas de desempenho de interesse;
  - . “passivos” no modelo—não participam, apenas monitoram;
  - . muito, são automáticos no Arena, mas alguns precisam ser inicializados e manipulados por você durante a simulação;
  - . ao final da simulação, são utilizados para calcular as medidas de desempenho finais.

# Partes de Um Modelo de Simulação (cont.)

- ” acumuladores de estatísticas para o sistema simplificado:
  - . número de partes produzidas até então;
  - . total de tempos gastos na fila até então;
  - . número de partes que passaram pela fila;
  - . maior tempo na fila já visto;
  - . total de tempos no sistema;
  - . maior tempo no sistema já visto;
  - . área sob a curva de tamanho da fila  $Q(t)$  até então;
  - . maior  $Q(t)$  até então;
  - . área sob a curva de ocupação do servidor  $B(t)$  até então;

# Eventos

“algo” que acontece em um instante de tempo (simulado) e pode mudar:

- . atributos,
- . variáveis,
- . acumuladores de estatísticas;

# Eventos para o Sistema Simplificado

- ” chegada de uma nova parte no sistema:
  - . atualizar os acumuladores de estatísticas,
    - ” área sob  $Q(t)$ ,
    - ” máximo  $Q(t)$ ,
    - ” área sob  $B(t)$ ;
  - . marcar a parte com o tempo corrente
  - . se a máquina está desocupada:
    - ” iniciar processamento (programar partida), tornar a máquina ocupada, inicializar o tempo na fila (0);
  - . senão (máquina ocupada):
    - ” colocar parte no final da fila, incrementar tamanho da fila;
  - . programar próximo evento de chegada.

# Eventos para o Sistema Simplificado (cont.)

- ” partida (quando o serviço é completado):
  - . incrementar o acumulador do unidades produzidas;
  - . calcular e acumular o tempo no sistema (agora - tempo de chegada);
  - . atualizar as estatísticas acumuladas;
  - . se a fila é não-vazia:
    - ” tome a primeira parte da fila, calcule e acumule o tempo na fila, inicie serviço (programar evento de partida);
  - . senão (fila está vazia):
    - ” faça a máquina desocupada (note que neste caso não haverá programação de evento de partida).

# Eventos para o Sistema Simplificado (cont.)

## ” Final

- . atualize as estatísticas acumuladas (até o final da simulação);
- . calcular as medidas de desempenho finais, utilizando as estatísticas acumuladas.

# Especificações para o Sistema Simplificado

- „ relógio da simulação (interno ao Arena)
- „ calendário de eventos (lista de eventos):
  - . [número da entidade, tempo do evento, tipo de evento];
  - . mantidos na ordem segundo a qual ocorrerão;
  - . próximo evento sempre no topo;
  - . inicialmente, programar a primeira chegada;
  - . final, corresponde à última partida;
- „ variáveis de estado (descrevem o status atual):
  - . status do servidor,  $B(t) = 1$ , se ocupado, 0, se livre;
  - . número de usuários na fila,  $Q(t)$ ;
  - . tempos de chegada de cada usuário (lista de tamanho aleatório).

# Simulação Manual

- ” acompanhar manualmente as variáveis de estados e os acumuladores;
- ” utilizar os tempos de chegada e de serviço;
- ” acompanhar o calendário de eventos;
- ” adiantar o relógio de um evento ao próximo;
- ” por simplicidade, omitir cálculos dos tempos no sistema e valores máximos;

# Implementação Computacional da Simulação Orientada a Eventos

- ” claramente adequada;
- ” frequentemente utiliza bibliotecas para:
  - . gerenciamento de listas;
  - . geração de números aleatórios;
  - . coleta de dados;
  - . gerência de lista de eventos e relógio;
  - . conclusões e saídas;
- ” programa principal agrupa bibliotecas e executa os eventos em ordem.

# Comparação de Alternativas

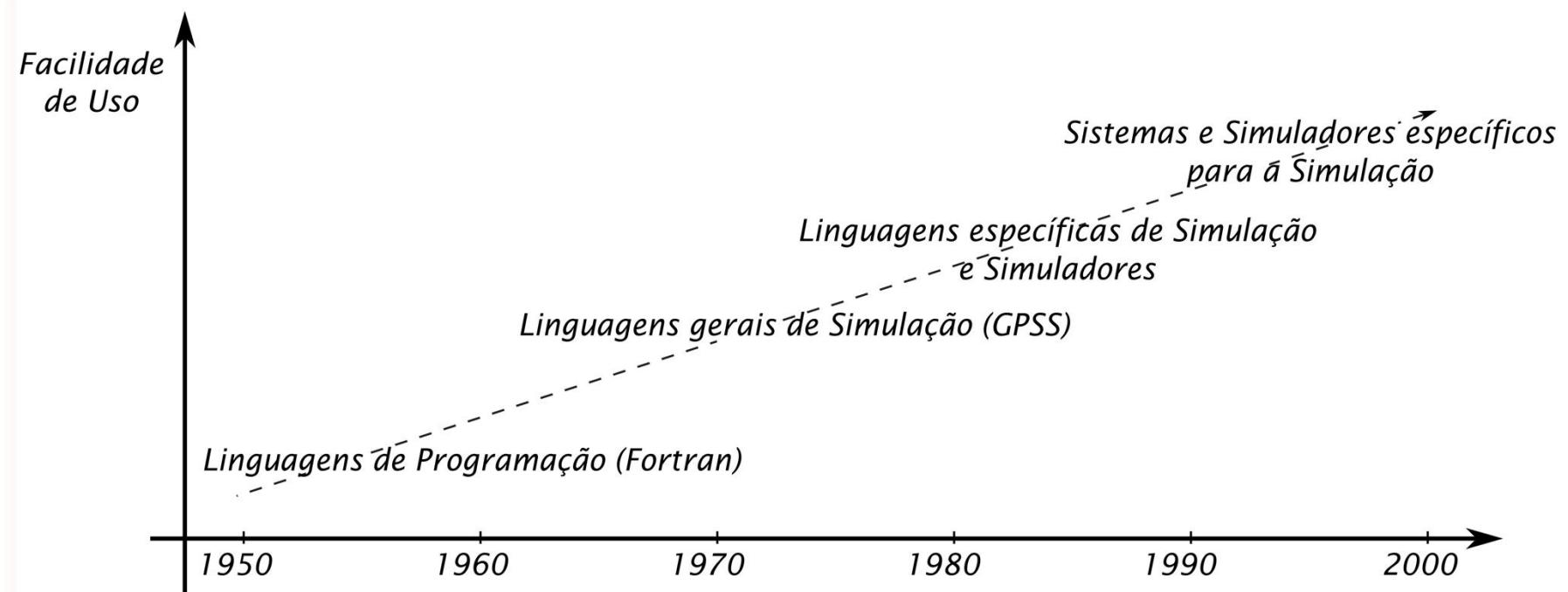
- ” Normalmente, simulação é utilizada para mais de uma simples configuração;
- ” freqüentemente, deseja-se comparar alternativas, selecionar ou procurar pela melhor (segundo algum critério);
- ” sistema simplificado: o quê aconteceria se a taxa de chegada dobrasse?
  - . reduzir o tempo entre chegadas à metade;
  - . simular novamente o modelo;
  - . realizar cinco replicações.

# Revisão do Estudo de Simulação

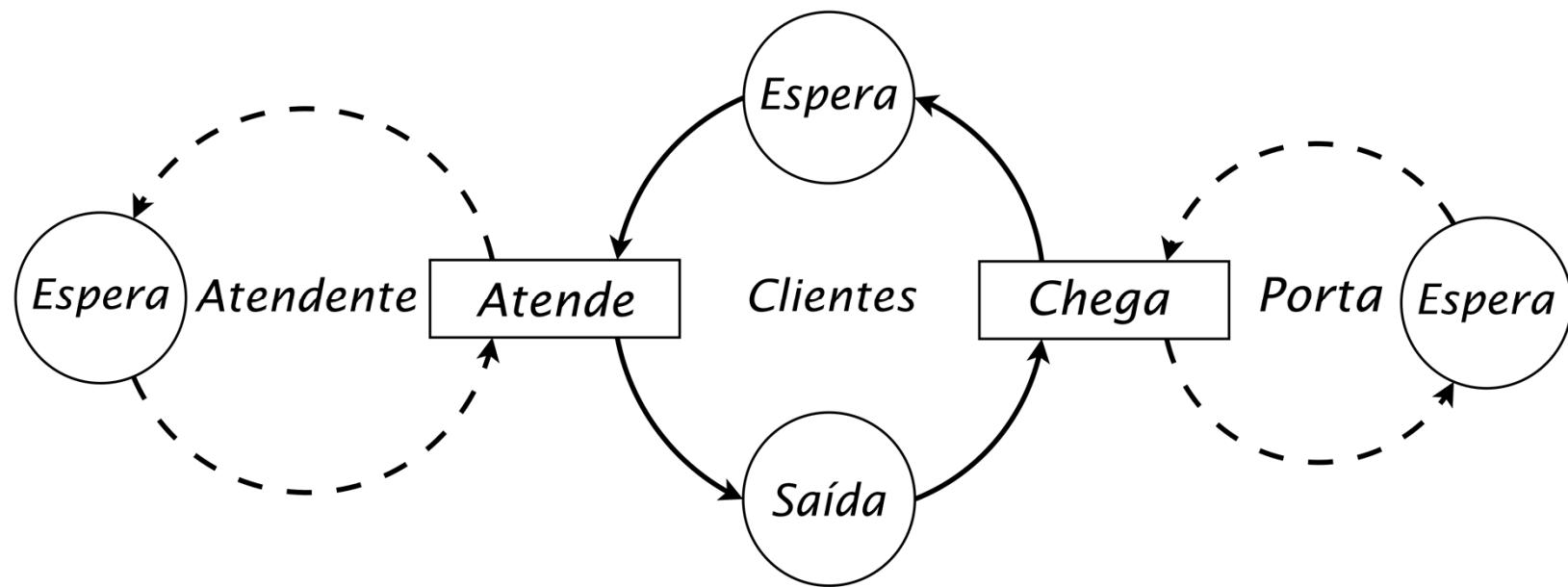
- “ entenda o sistema;
- “ tenha objetivos claros;
- “ formule um modelo de representação;
- “ traduza o modelo para um *software* de simulação;
- “ verifique o “programa”;
- “ valide o modelo;
- “ planeje os experimentos;
- “ faça simulações;
- “ analise, compreenda, documente os resultados.

# Softwares de Simulação

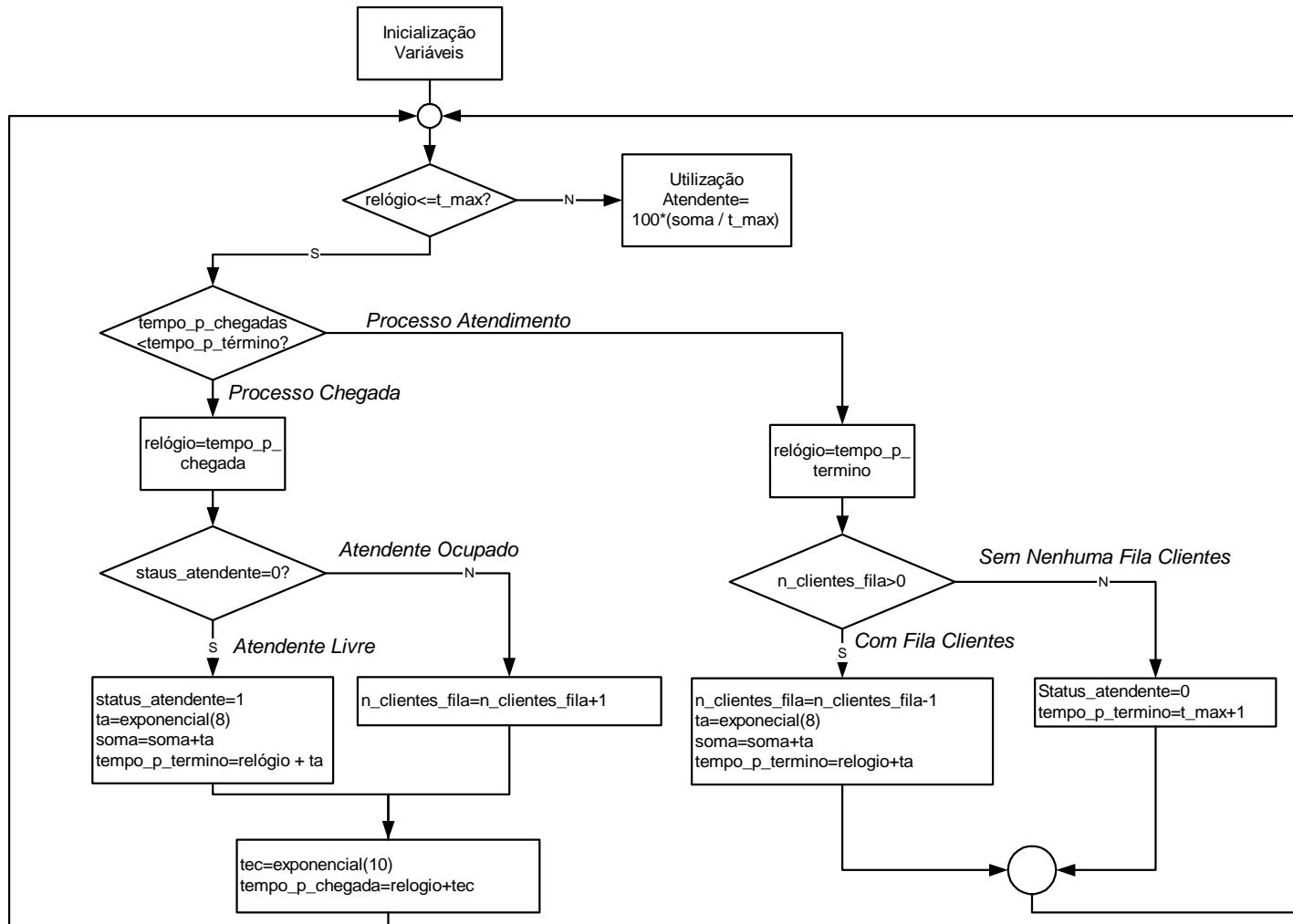
## Uma Visão Geral



# Softwares de Simulação



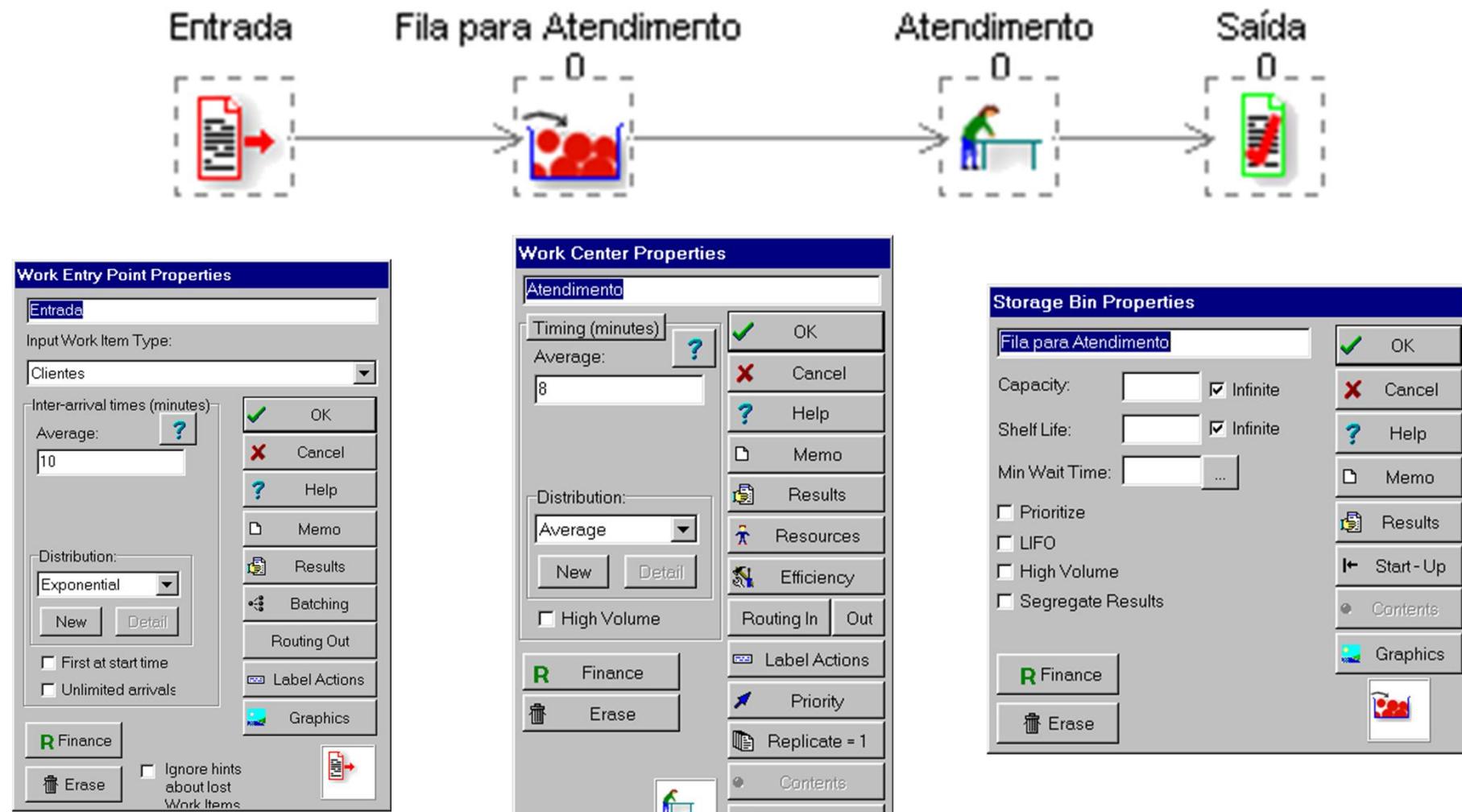
# Softwares de Simulação



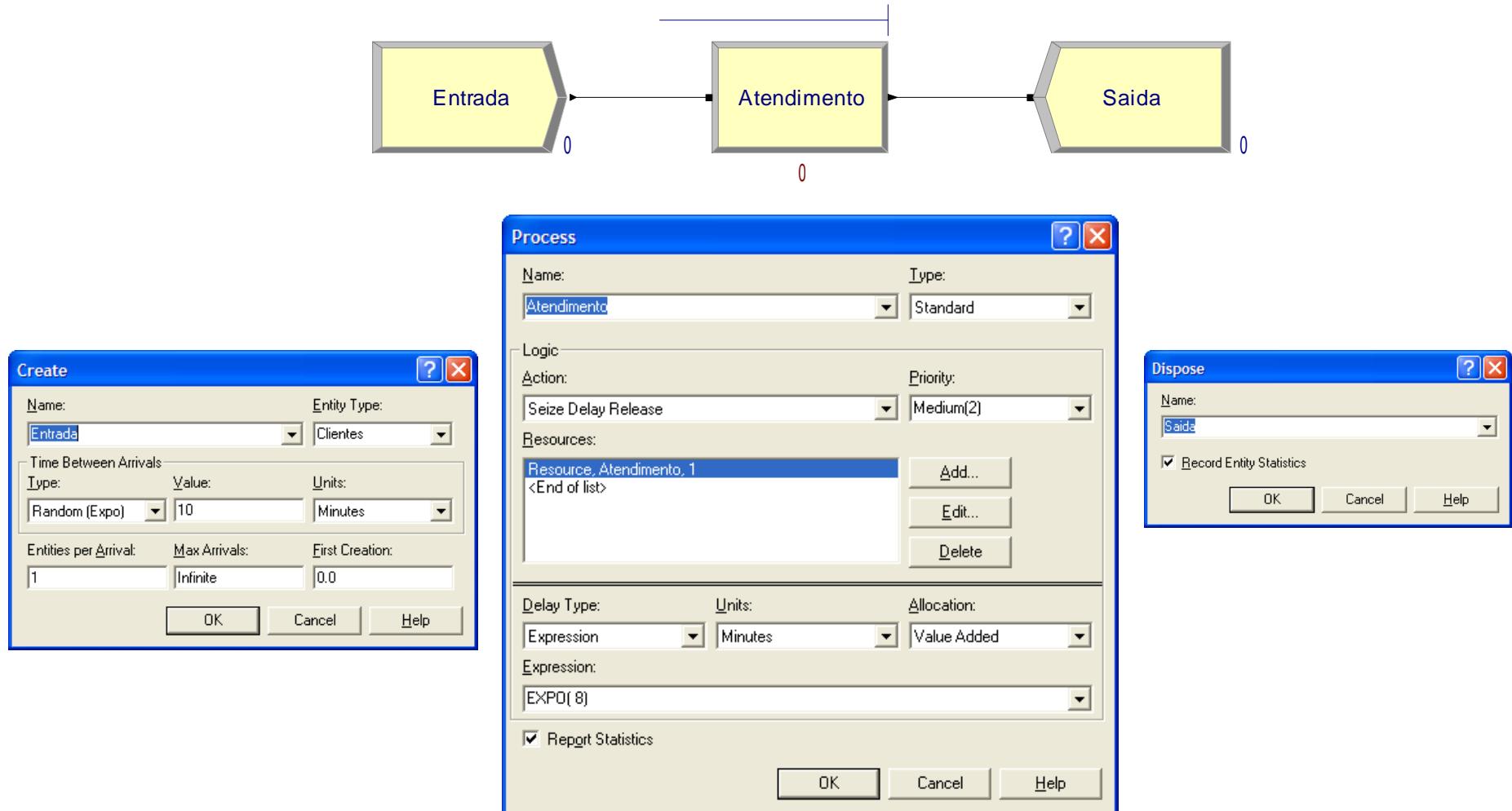
# Softwares de Simulação

```
Initialization()
CreateQ(Q_sai, Infinite)
CreateQ(Q_esp,0)
CreateQ(Q_Atend,1)
CreateQ(Q_Porta,1)
Create_Var(tempo1)
Create_Var(tempo2)
End
C_event(Chegada, Qsize(Q_Sai)>=1 and Qsize(Q_Porta)>=1) {Chegada}
    Set_Var(tempo1,E(10))
    Schedule(B1, Dequeue(Q_porta),tempo1)
    Schedule(B2, Dequeue(Q_sai), tempo1)
End
B_event(B1) {Porta acaba}
    Enqueue(Q_Porta, Current, true)
End
B_event(B2)
    Enqueue(Q_Esp, current, true)
End
C_event(atendimento,Qsize(Q_esp)>=1) and Qsize(Q_Atend)>=1)
Set_Var(tempo2,E(8))
    Schedule(B3,Dequeue(Q_esp), tempo2)
    Schedule(B4,Dequeue(Q_Atend),tempo2)
End
B_event(B3) { clientes saem do sistema}
    Enqueue(Q_sai, current, true)
End
B_event(B4) {atendente termina atendimento}
    Enqueue(Q_atend,current, true)
End
Termination(S_time>=10000)
    Message("utilização do atendente = ", utilization(Q_attend))
```

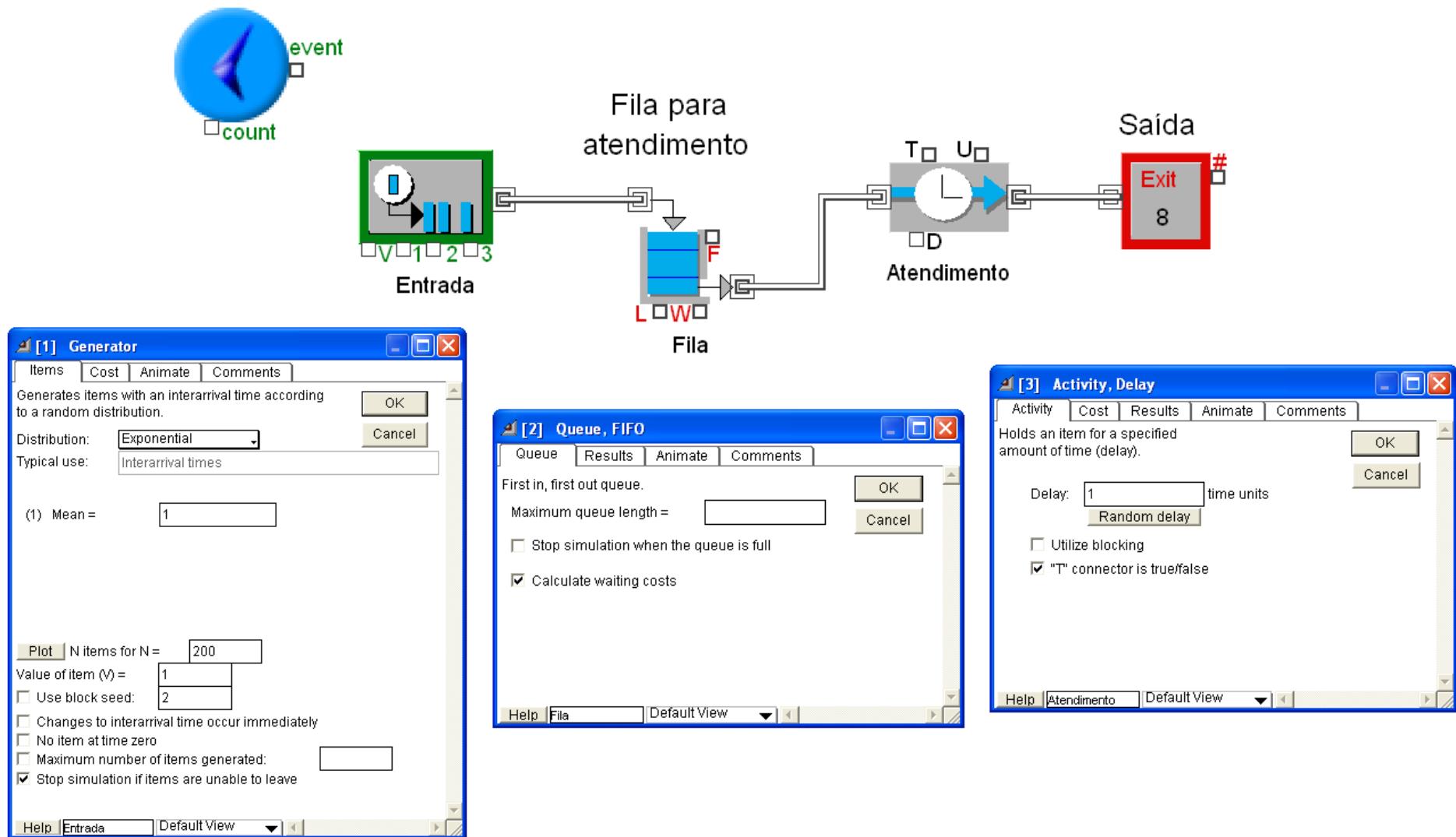
# Softwares de Simulação - Simula



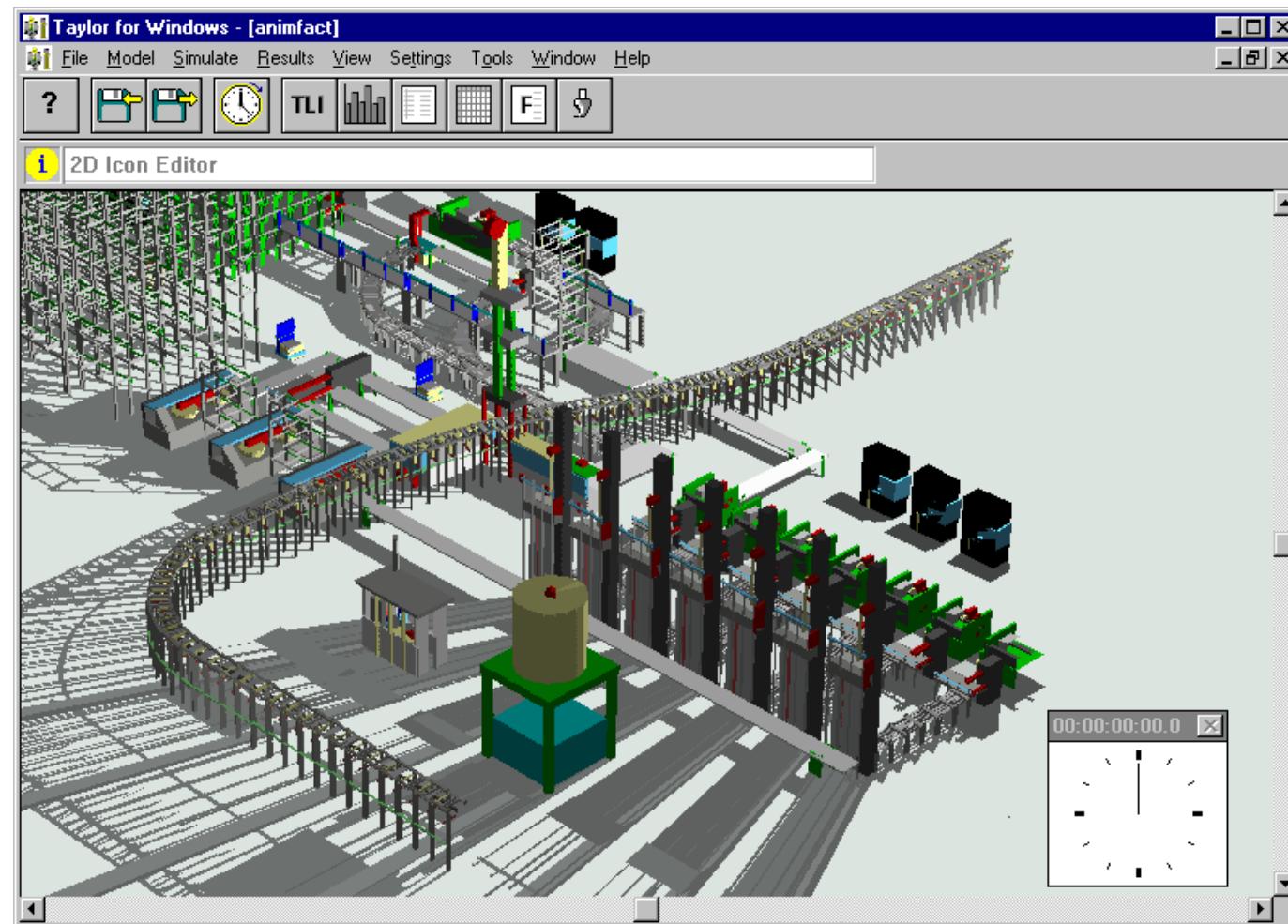
# Softwares de Simulação - ARENA



# Softwares de Simulação - Extend



# Softwares de Simulação . Efeitos 3D em Modelos 2D

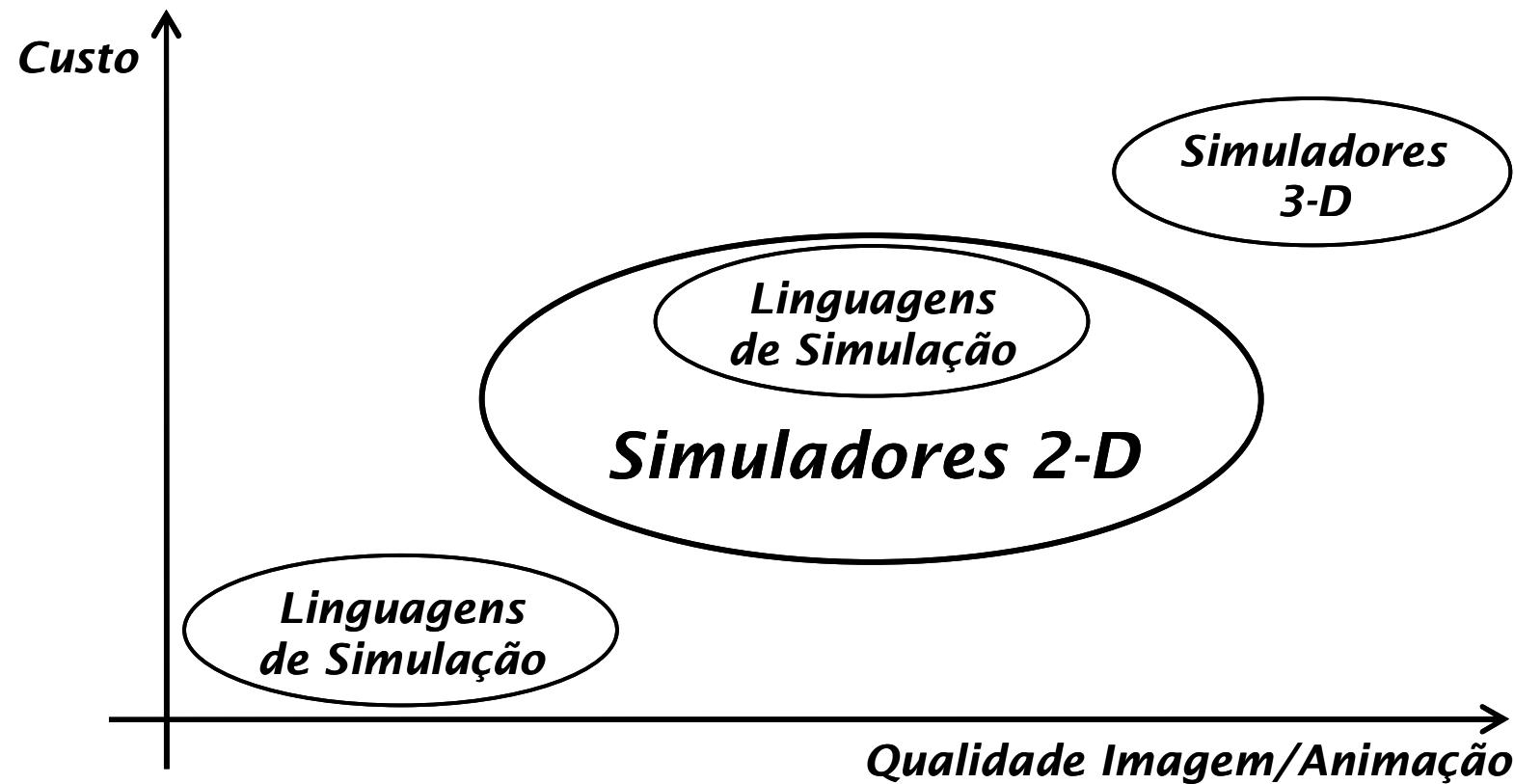


# Softwares de Simulação

” Observe algumas animações de simulações complexas usando diferentes ferramentas no link:

<http://www.youtube.com/watch?v=fDr6ZAhTuqE>

# Softwares de Simulação



# Modelagem - Conceitos

- *Problema do Mundo Real*
  - ✚ *Modelo Abstrato*
  - ✚ ***Modelo Conceitual***
  - ✚ *Modelo Computacional*
  - ✚ *Modelo Operacional*
  - ✚ *Resultados Experimentais*
- *Conclusões*

# Modelagem

- ” Entendimento
- ” Facilitar Criação do Modelo Computacional
- ” Melhorar Validação do Modelo
- ” Meio de Comunicação

# Modelagem

Analistas “**pulam**” a fase de modelagem conceitual indo diretamente para o modelo computacional:

- Falha de comunicação do modelo entre as pessoas – obrigação de “**consertar**” após implementar o modelo
- Inviabiliza a participação de **muitas pessoas** no processo de modelagem
- Causa um maior esforço na **validação do modelo**

# Modelagem

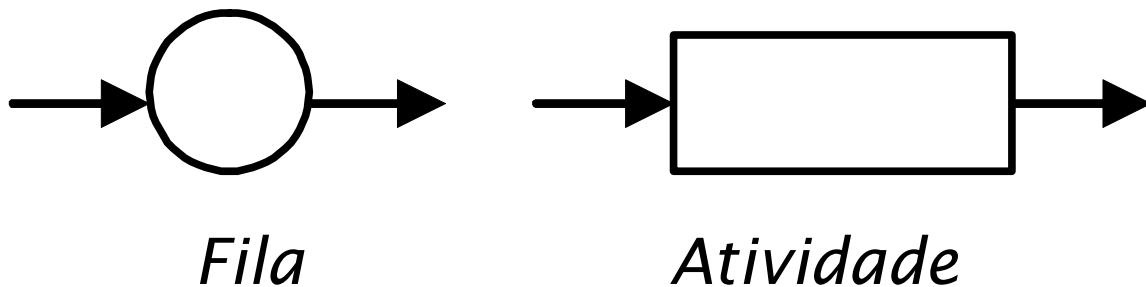
“Baseado na idéia das “Engrenagens Estocásticas”

“**Simples:** constituído de 2 símbolos básicos

“Indica explicitamente as interações entre os objetos do sistema e seus fluxos

“**Fácil** de entender e utilizar

# Modelagem



- **Entidade** = qualquer componente no modelo que retém sua *identidade* ao longo do tempo
- **Fila** = elemento **passivo** (uma *fila* para cada tipo de entidade)
- **Atividade** = elemento **ativo** , possível cooperação entre diferentes entidades.

# Modelagem

- ”Importante para o analista **aumentar sua sensibilidade** em relação a execução da simulação.
- ”Uma ferramenta de **verificação se a lógica** do modelo está coerente.
- ”Permite **estabelecimento de prioridades** onde existem.
- ”Promove um **melhor entendimento** do mecanismo de simulação

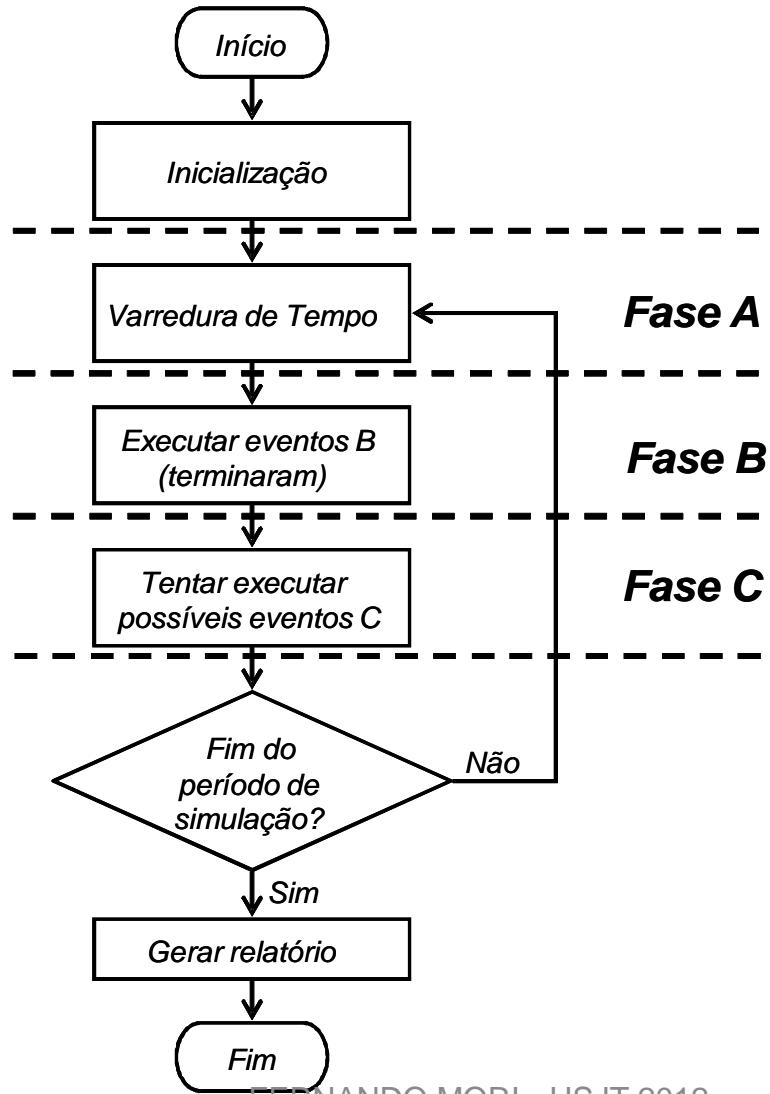
# Modelagem

““A”: Checar o tempo de todas as atividades em progresso. Determinar a que ocorre primeiro. **Avançar** o relógio de simulação.

““B”: Para as atividades que **terminaram**, mover as entidades para as respectivas filas.

““C”: Procurar as atividades em uma ordem determinada e **iniciar as que tem condição de começar**. Mover as entidades das filas para a atividade. Amostrar e calcular o tempo de término da atividade.

# Modelagem



# Modelagem

A	B	C
<ul style="list-style-type: none"> <li>- checar o tempo de todas atividades</li> <li>- determinar quem ocorre primeiro</li> <li>- avançar o relógio</li> </ul>	<ul style="list-style-type: none"> <li>- determinar as atividades que TERMINARAM e mover as entidades para as respectivas FILAS</li> </ul>	<ul style="list-style-type: none"> <li>- Procurar as entidades e iniciar as que tem condição de COMEÇAR</li> <li>- Mover as entidades das filas para ATIVIDADES</li> <li>- Amostrar e calcular o tempo de TÉRMINO da atividade</li> </ul>
0	-x-	chegada começa, termina em 1
1	termina chegada	chegada começa, termina em $1+20=21$ encher começa ( $N=4$ ), termina em $1+5=6$
6	termina encher	beber começa, termina em $6+7=13$
13	termina beber	beber começa, termina em $13+7=20$ lavar começa, termina em $13+5=18$
•	•	•
•	•	•
•	•	•

# Modelagem

“Um modelo é uma **representação do mundo real**, ou pelo menos de parte dele. Portanto, a validação de um modelo é realmente muito direta – em princípio. Tudo o que devemos fazer é checar **se o modelo comporta-se como o mundo real** sob as mesmas condições. Se ele se comporta, então o modelo é válido, caso contrário, não é válido.”

# Modelagem

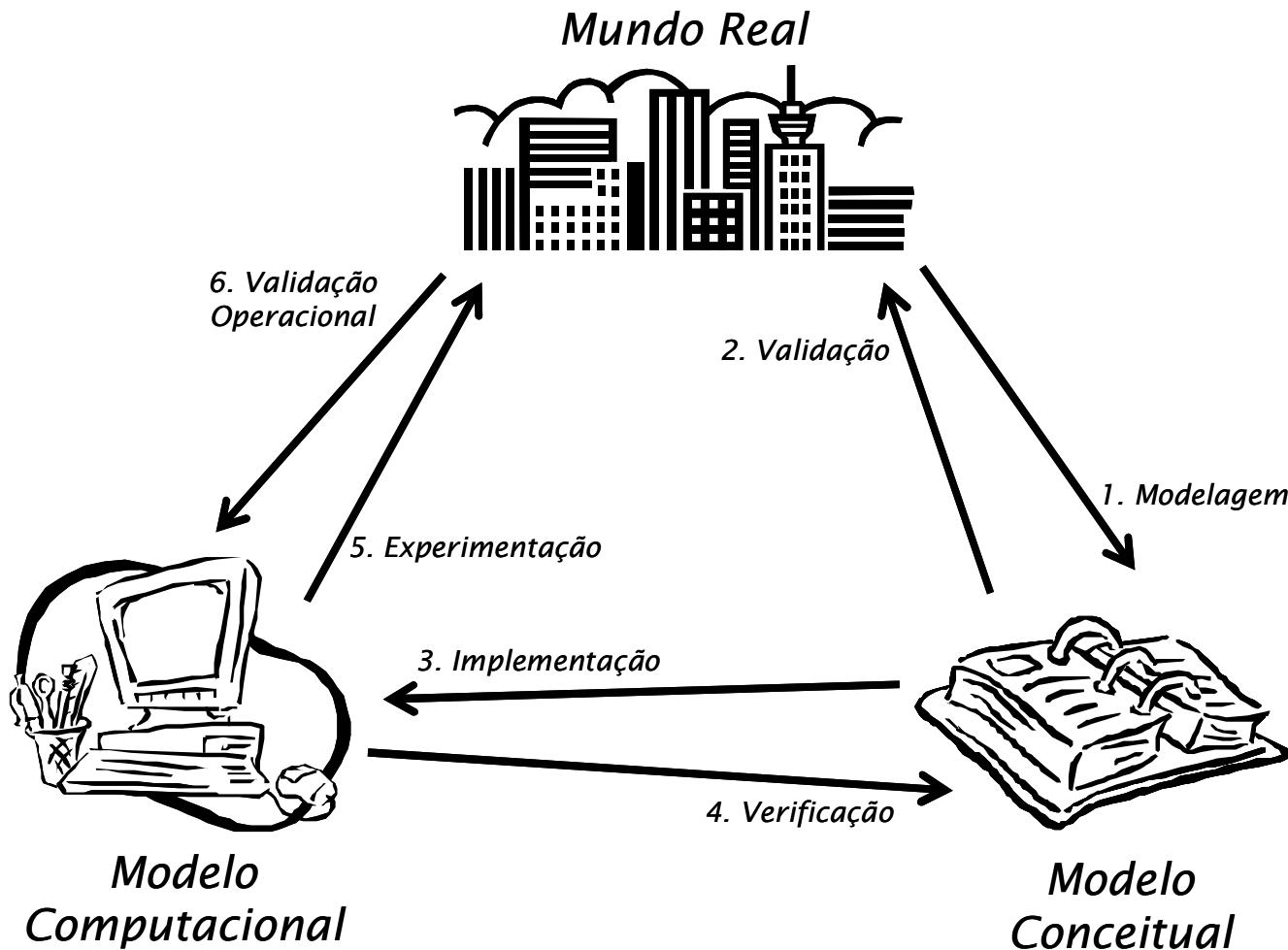
## ” VERIFICAÇÃO

- . O Modelo faz o que **eu** quero?

## ” VALIDAÇÃO

- . O Modelo **funciona** como no mundo **real**?

# Modelagem



# Modelagem

- Relação entre o **modelo conceitual** e o **modelo computacional**
- Consiste em assegurar que o modelo computacional funcione conforme o **cliente** deseja.
- A verificação de modelos de simulação é equivalente a retirar os %Bugs+de programas. (debugging)

# Modelagem

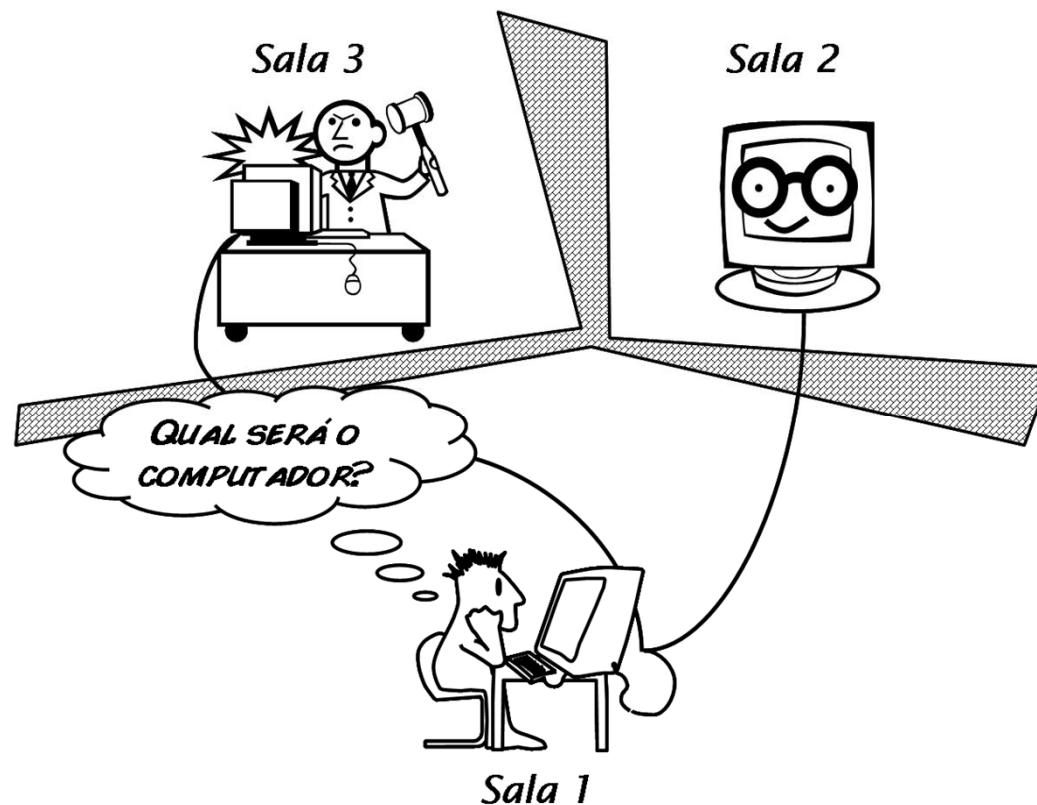
- *Implementação Modular / Verificação Modular*
- *Valores constantes/simplificados + cálculos manuais*
- *Utilização do %Debugger+(Trace)*
- *Simulação Manual*
- *Animação Gráfica*
- *Revisão em grupo*

# Modelagem

- A Validação é um processo muito **mais difícil e sofisticado** que a verificação
- Não há como validar um modelo na prática** e sim como aumentar sua confiança com que ele representa a realidade
- Como validar **sistemas novos**?

# Modelagem

- *Teste de Turing ou validação black-box.*



# Modelagem

- *Duplicação de modelos*
- *Comparação com modelos anteriores*
- *Análise de sensibilidade*
- *Validação % face a face+*

# Modelagem

- *Etapa importante do processo de modelagem. Muitos se esquecem dela, devido a grande dificuldade.*
- *Não há como garantir que o modelo é 100% livre de bugs+e sim minimizá-los.*
- *Não há como validar 100% um modelo e sim aumentar sua confiança.*
- *Deve ser um processo contínuo (envolve vários ciclos).*

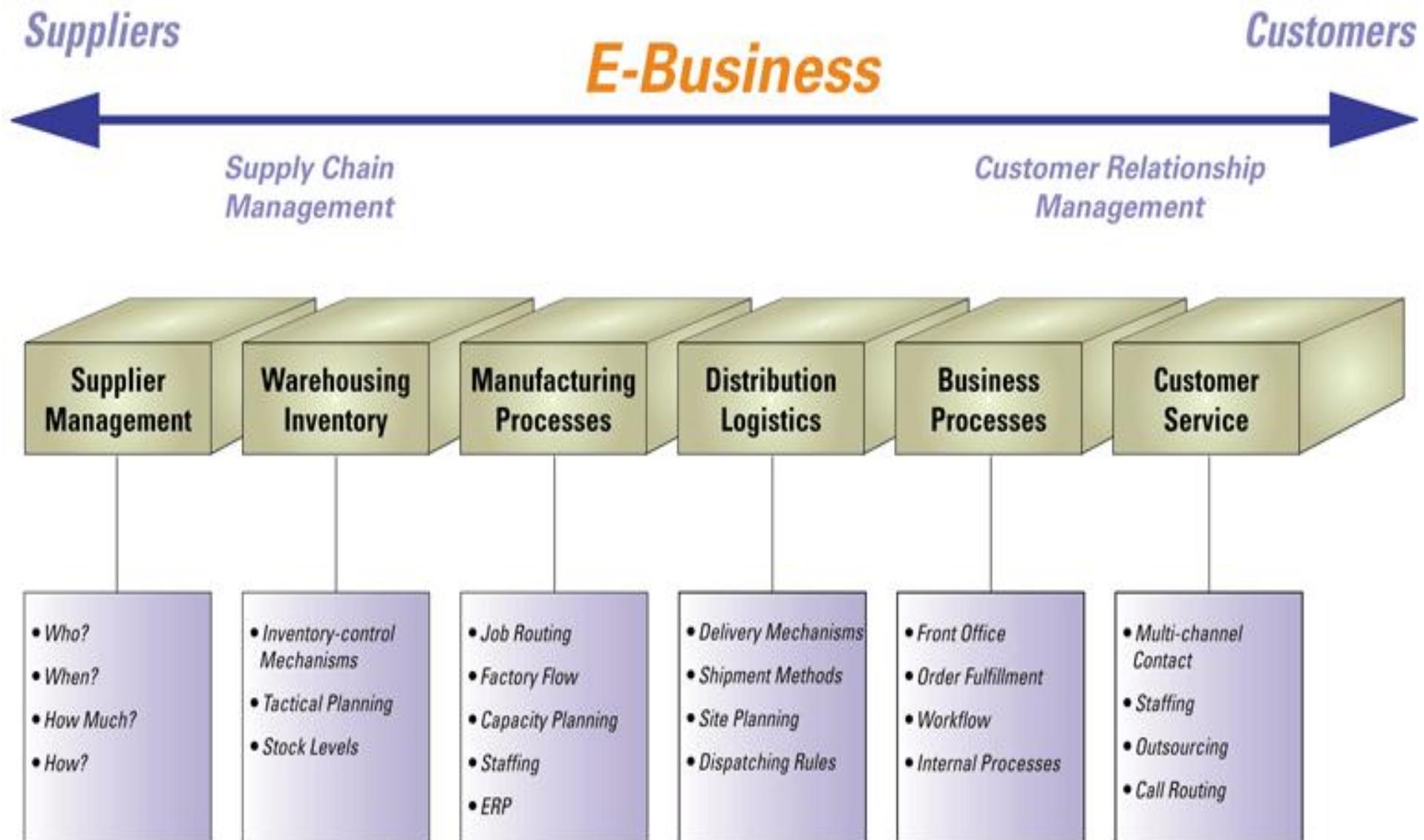


# *Arena* Enterprise Simulation Software

<http://www.rockwellautomation.com/>



# Modelos Arena



# Usando o ARENA em Simulação

- ” Origem
- ” Empresa Rockwell – sediada na Inglaterra
- ” Lançado em 1993
- ” Atualmente estamos na versão 11.0
  - . construção de modelos de simulação
  - . Analisador de dados de entrada
  - . Analisador de resultados
  - . Visualizador da simulação
  - . Execução em lotes
  - . Recursos de otimização
  - . Animação em 3D

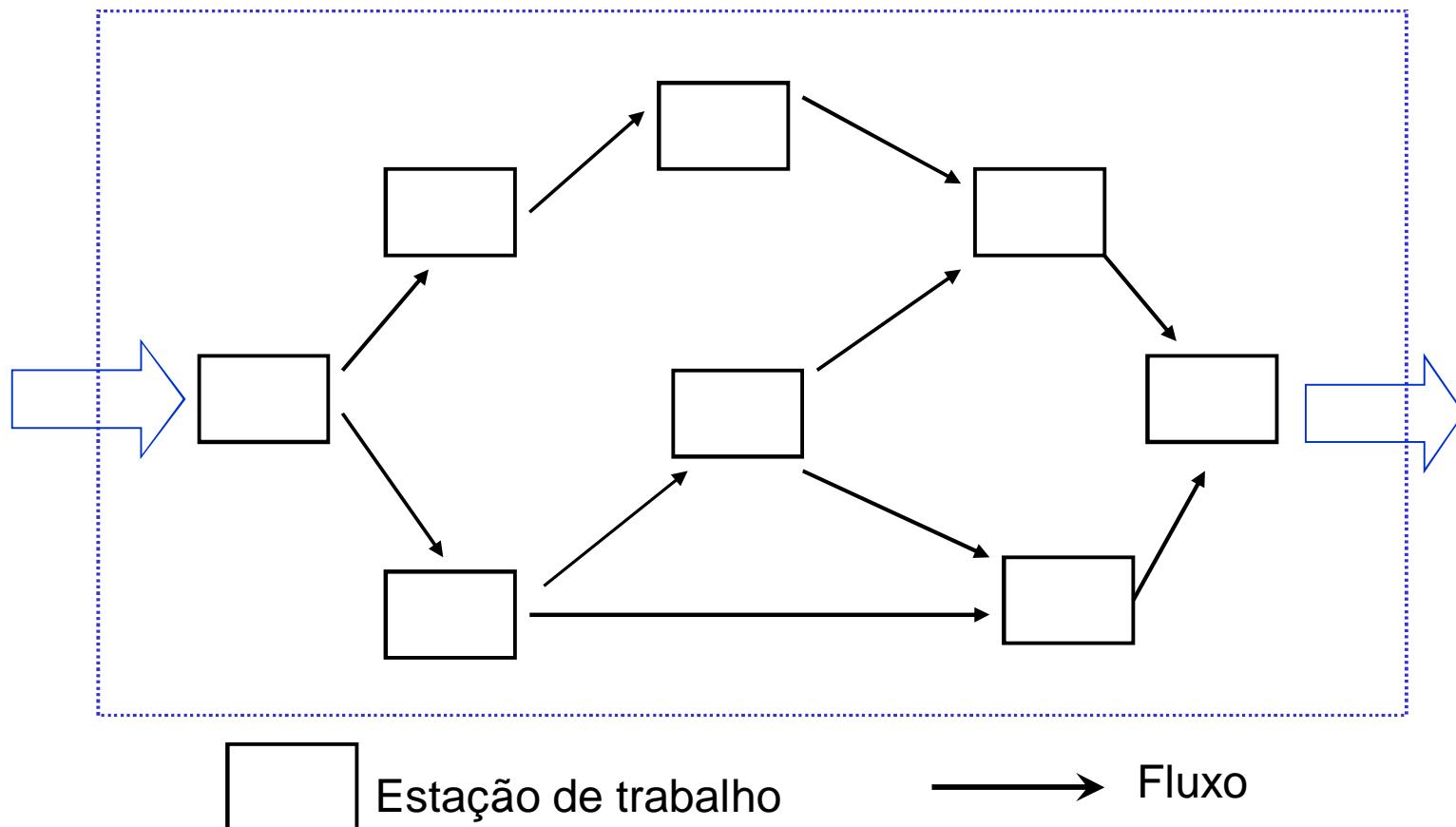
## A VISÃO DE MUNDO DO ARENA

O arena visualiza o sistema a ser modelado como um conjunto de estações de trabalho que prestam serviços a clientes (também chamados de entidade ou transações), que se movem através do sistema. O movimento pode ser feito pela própria entidade ou por transportadores.

Esta característica básica pode ser usada de inúmeras maneiras, por exemplo:

- Pessoas (entidades) percorrendo as diversas seções (stations) de um supermercado e,m que efetuam compras.
- Um automóvel (entidade) sendo fabricado nas diversas seções (stations) de uma fábrica.
- Uma apólice de seguro (entidade) sendo processada nas diversas seções (stations) de uma seguradora.

- Clientes (entidades) chegam a um banco e utilizam os serviços dos diversos departamentos (stations) do banco.



Para efetuar o dimensionamento de um sistema sempre nos referimos a variáveis como o tempo de espera do cliente na fila, a quantidade de atendentes, etc.

Em simulação estas variáveis são aleatórias, ou seja são descritas por uma distribuição de probabilidades.

Considerando um exemplo simples no qual os clientes chegam e entram na fila, existindo **M** servidores para atendê-los.

As principais variáveis usadas no dimensionamento são:

- Variáveis referentes ao processo de atendimento

TA = tempo médio de atendimento ou serviço  
(Delay time ou Process time)

C = quantidade de atendentes

NA= número médio de clientes que estão sendo atendidos

$\mu$  = rítmico médio de atendimento de cada cliente

Por definição:  $TA = \frac{1}{\mu}$

Relações básicas: NS = NF + NA

TS = TF + TA

- Taxa de utilização dos atendentes

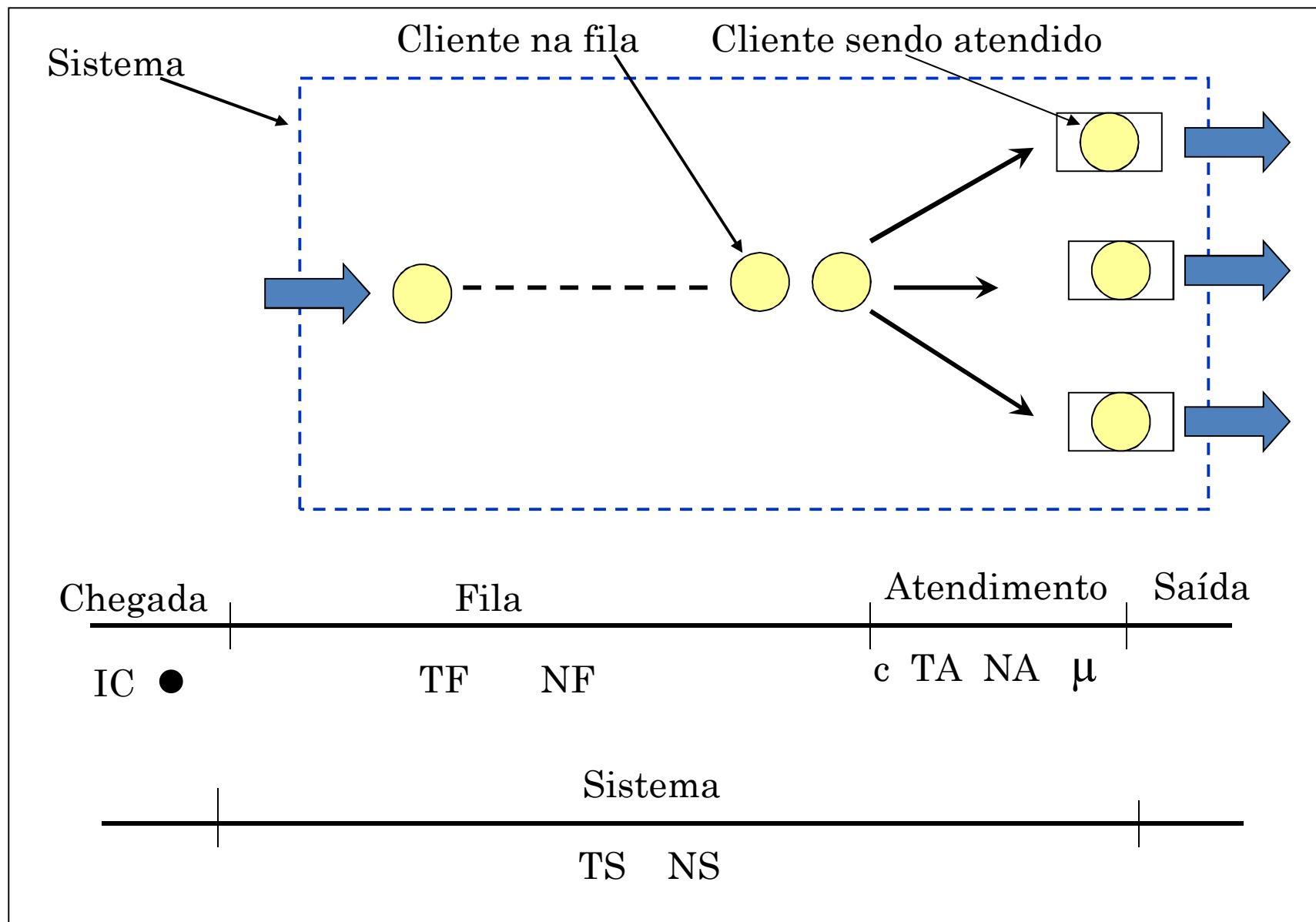
Para o caso de uma fila/um atendente chamamos de taxa de utilização do atendente a relação entre seu tempo total ocupado e o tempo total disponível.

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

No caso de uma fila com vários atendentes:

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

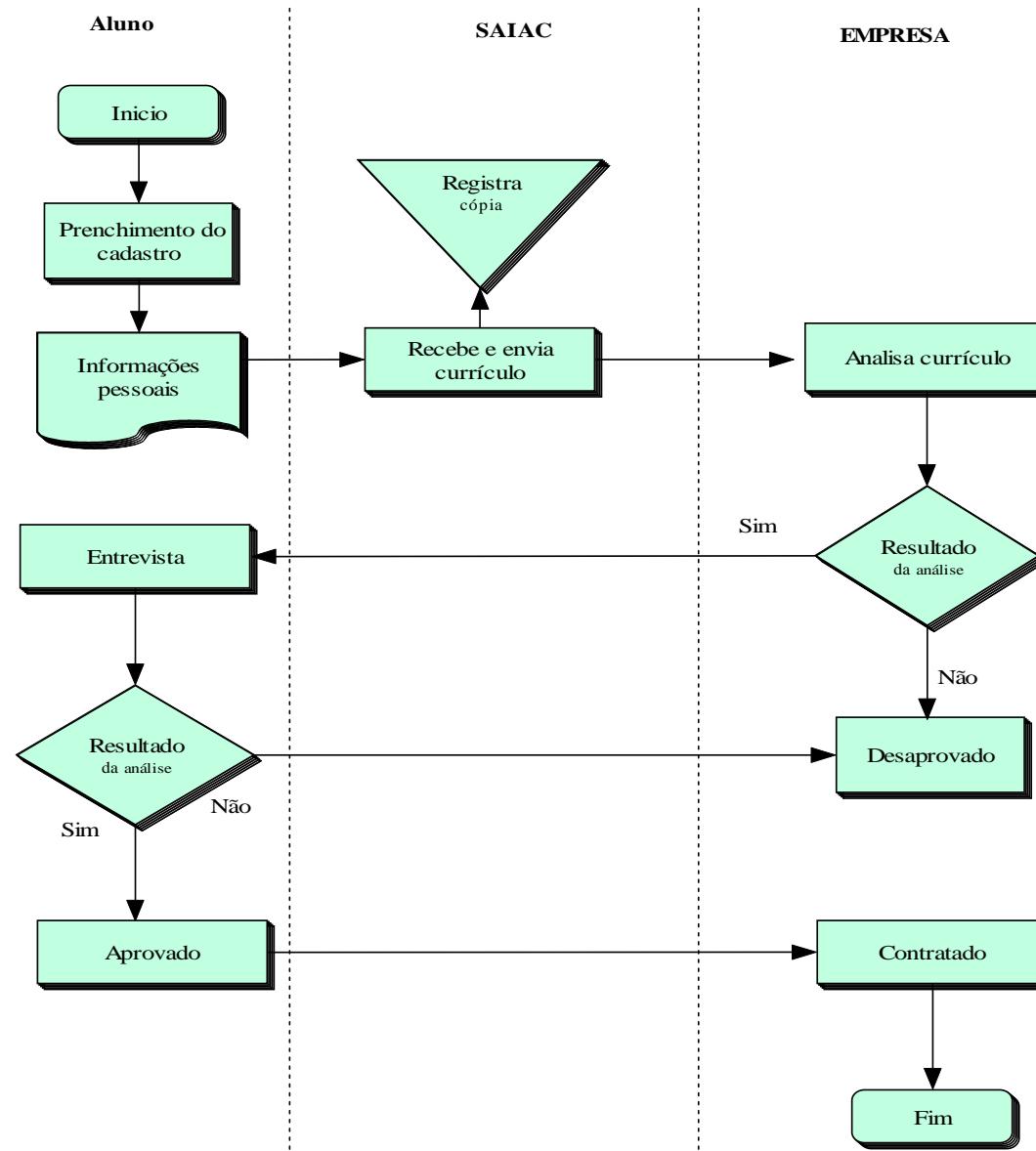
onde C é o número de atendentes.



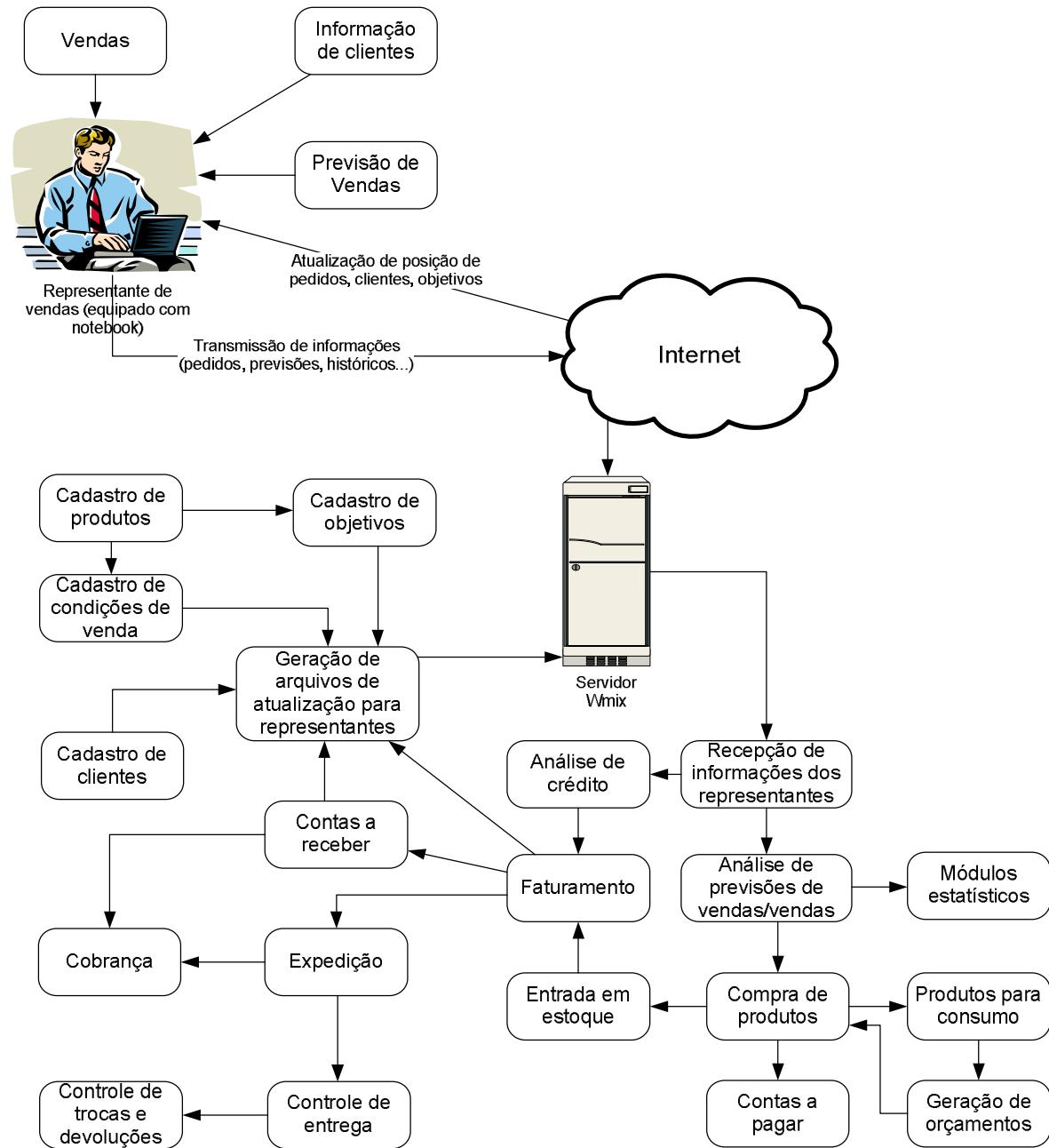
# Alguns Exemplos de Modelagem

# 1) Fluxograma de encaminhamento de alunos para estágio

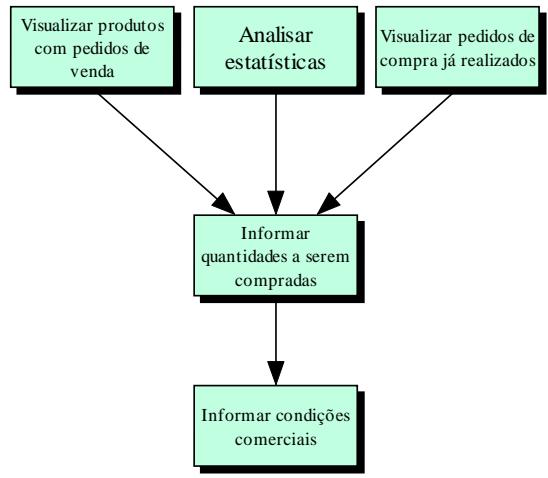
**Encaminhamento de alunos para estágio**



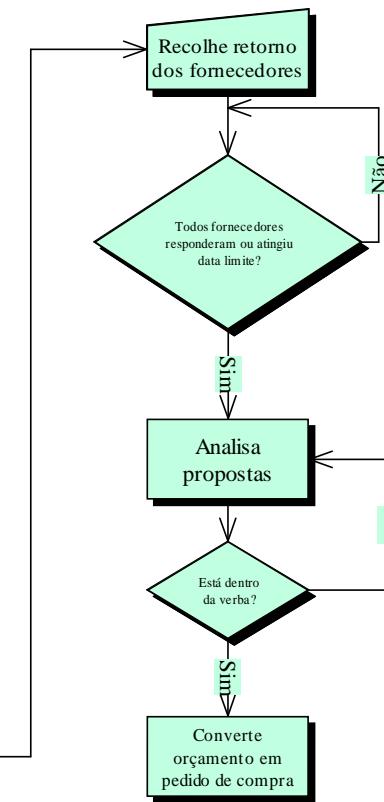
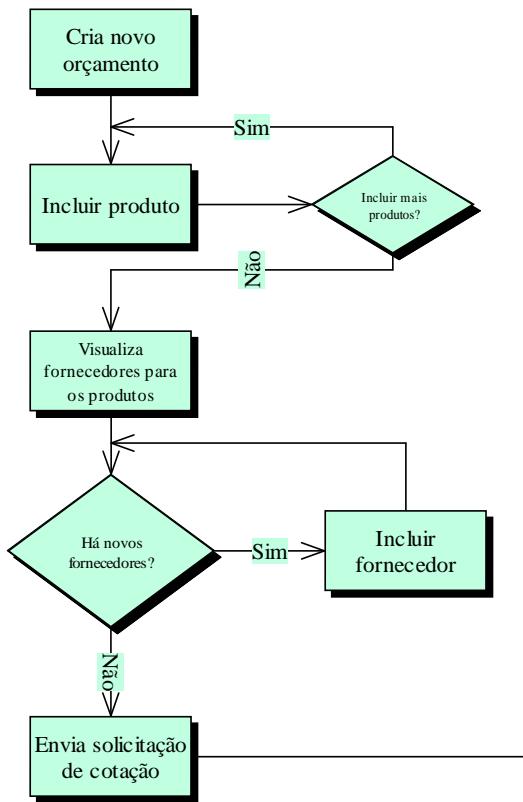
## 2) Fluxo de Integração de Atividades de uma pequena Empresa.



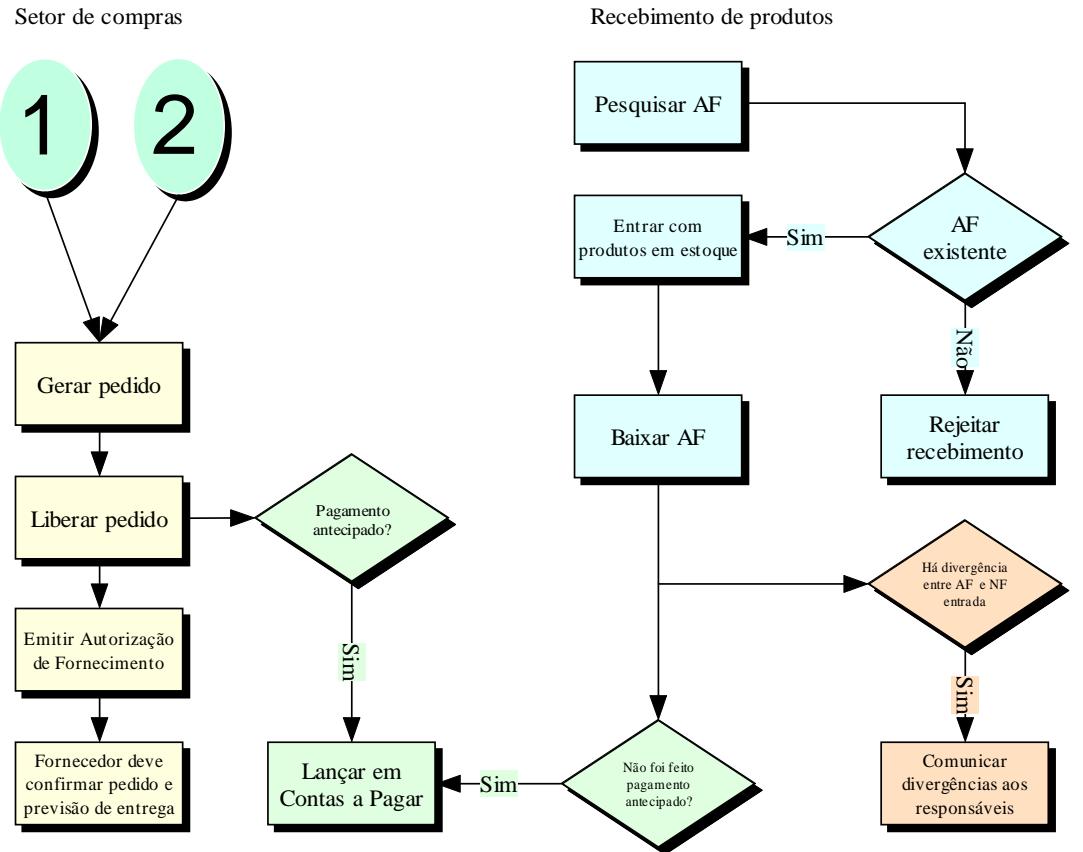
3) Fluxo de pedido de compras e integração com outras atividades.



1



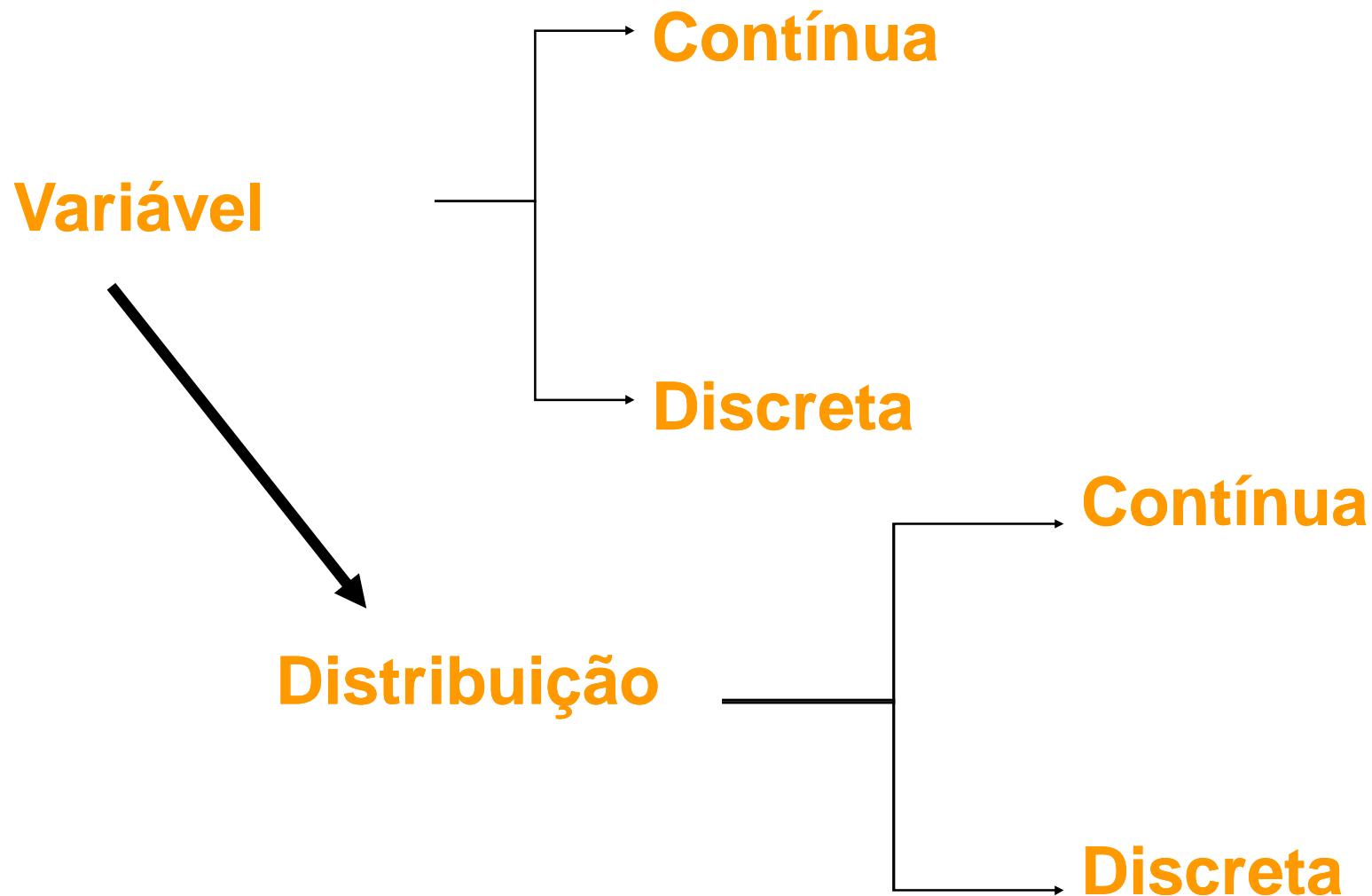
2



# ARENA

- ” Para a construção de um modelo no ARENA são necessários fornecer dados que devem ser baseados na realidade.
- 1. Processo de Chegadas.
- 2. Processo de Atendimento.
- 3. Deslocamento entre Estações de Trabalho.
- ” Todos esses dados são dados na forma de distribuições de probabilidades.

# Distribuições de Probabilidade



# Distribuições de Probabilidade

## ***Normal***

A distribuição Normal, descreve fenômenos regidos por variáveis aleatórias que possuem variação simétrica acima e abaixo da média. Muito utilizada em tempos de processo como tempos de máquina.

Sua mais importante contribuição é o fato de que os possíveis valores de uma variável aleatória, que são resultantes da soma ou da média, de um grande número de outras variáveis aleatórias, resulta em uma curva cuja forma pode ser aproximada por uma Normal.

## ***Beta***

Devido a sua capacidade de se adequar a várias formas, esta distribuição é usada como uma aproximação, quando houver ausência de dados.

## ***Uniforme***

A distribuição Uniforme especifica que cada valor entre um mínimo e um máximo especificado, tenham igual probabilidade de acontecer. Costuma-se utilizar esta distribuição quando pouco ou quase nada se sabe a respeito do comportamento da variável aleatória que estamos tratando, a exceção de seus pontos extremos.

# Distribuições de Probabilidade

## *Triangular*

A distribuição Triangular não é identificada com nenhum tipo de operação específica, mas é útil quando se deseja uma primeira aproximação na falta de dados específicos. Além dos valores mínimo e máximo característico da distribuição uniforme, o conhecimento de um valor mais provável, valor modal, permite o uso desta distribuição, no lugar da uniforme. É muito utilizada quando não existem dados suficiente e é necessária uma estimativa.

## *Exponencial*

A distribuição exponencial é uma das mais utilizadas em modelo de simulação.

O principal uso é da modelagem de períodos de tempos entre dois acontecimentos quaisquer.

## *Erlang*

Utilizada na simulação de alguns tipos de processos, muitas vezes em situações em que uma entidade entra em uma estação para ser servida, seqüencialmente, por uma série de postos.

## *Gamma*

Esta função costuma ser aplicada para representar tempo de completar alguma tarefa.

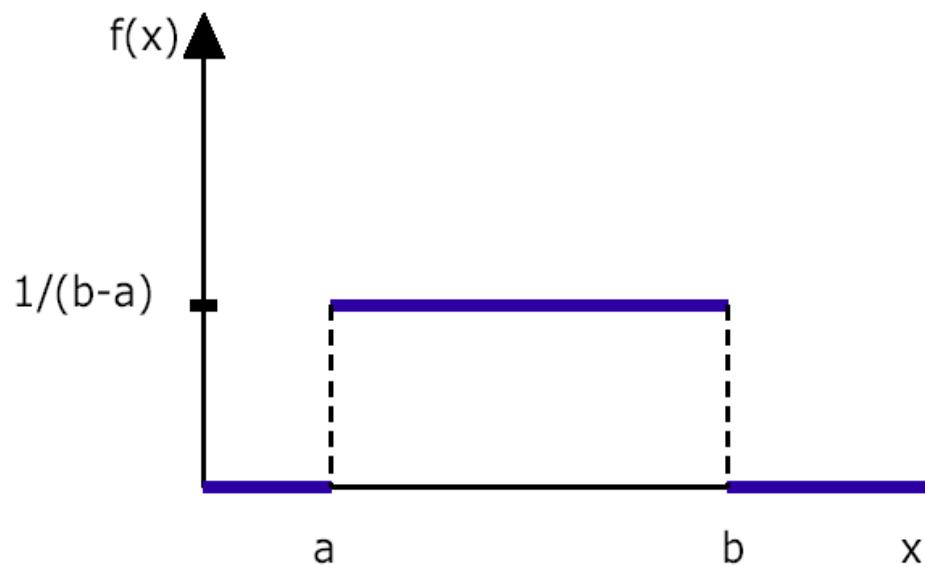
## *Log Normal*

A distribuição Log-Normal é empregada em situações onde a quantidade é o produto de um número grande de quantidades aleatórias. É freqüentemente utilizada para representar tempos de atividades com distribuição não simétrica.

# Distribuições de Probabilidade

## Distribuições Contínuas Uniforme

Primeira tentativa  
em casos em que  
apenas os limites  
dos dados são  
conhecidos.



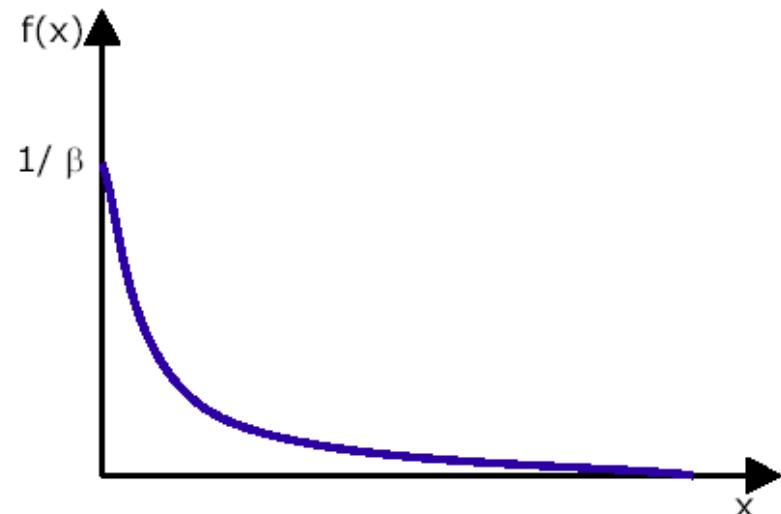
$$f(x) = \begin{cases} 1/(b-a) & \text{se } a \leq x \leq b \\ 0 & \text{caso contrário} \end{cases}$$

# Distribuições de Probabilidade

## Distribuições Contínuas Exponencial

Intervalos de tempo de chegada de clientes a um sistema, cuja chegada ocorre com uma determinada taxa constante.

$$f(x) = \begin{cases} \frac{1}{\beta} e^{-x/\beta} & \text{se } x \geq 0 \\ 0 & \text{caso contrário} \end{cases}$$



# Distribuições de Probabilidade

## Distribuições Contínuas

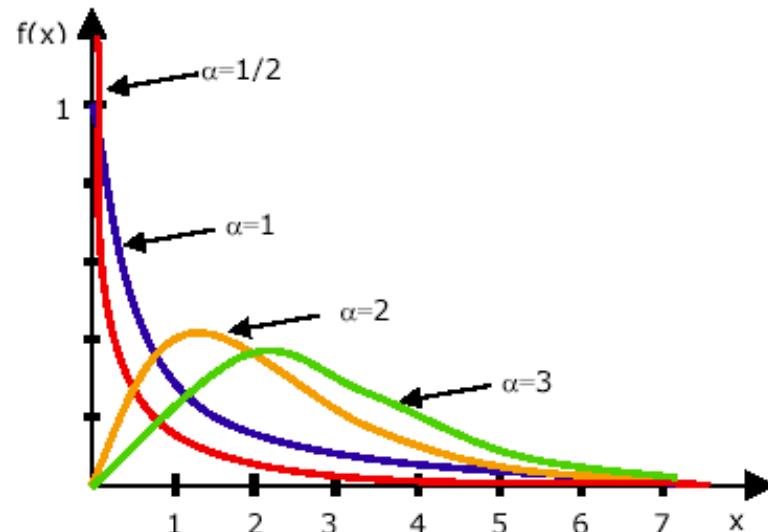
### Gama

Tempo para  
realizar alguma  
tarefa.

$$f(x) = \begin{cases} \frac{\beta^{-\alpha} x^{\alpha-1} e^{-x/\beta}}{\Gamma(\alpha)} & \text{se } x > 0 \\ 0 & \text{caso contrário} \end{cases}$$

sendo  $\Gamma(\alpha)$  a função Gama definida como

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt \quad \text{para } z > 0$$



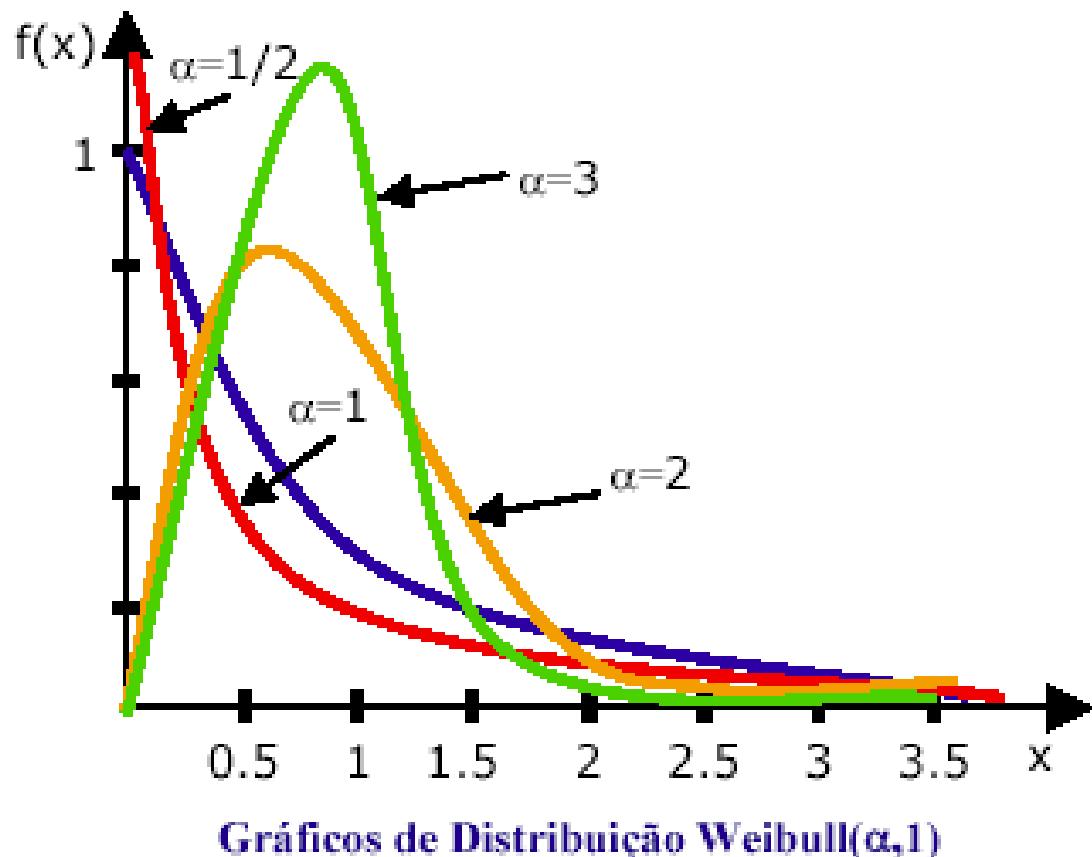
Gráficos da Distribuição Gama( $\alpha, 1$ )

# Distribuições de Probabilidade

## Distribuições Contínuas

### Weibull

Tempo para  
realizar alguma  
tarefa.

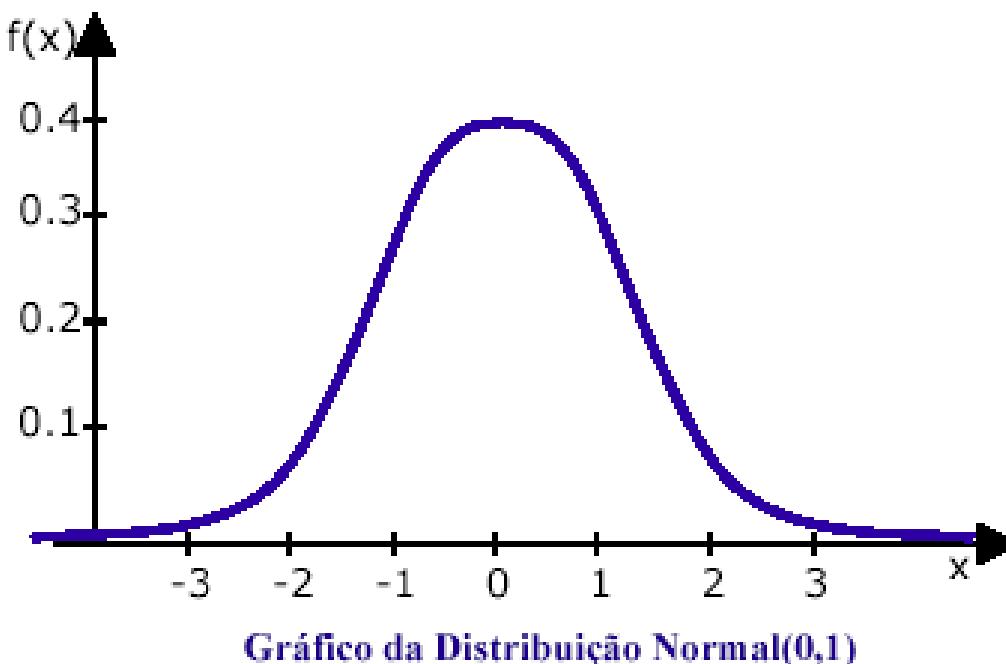


# Distribuições de Probabilidade

## Distribuições Contínuas Normal

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x-\mu)^2/(2\sigma^2)}$$

Valores que são a  
soma de grande  
número de outros  
valores.

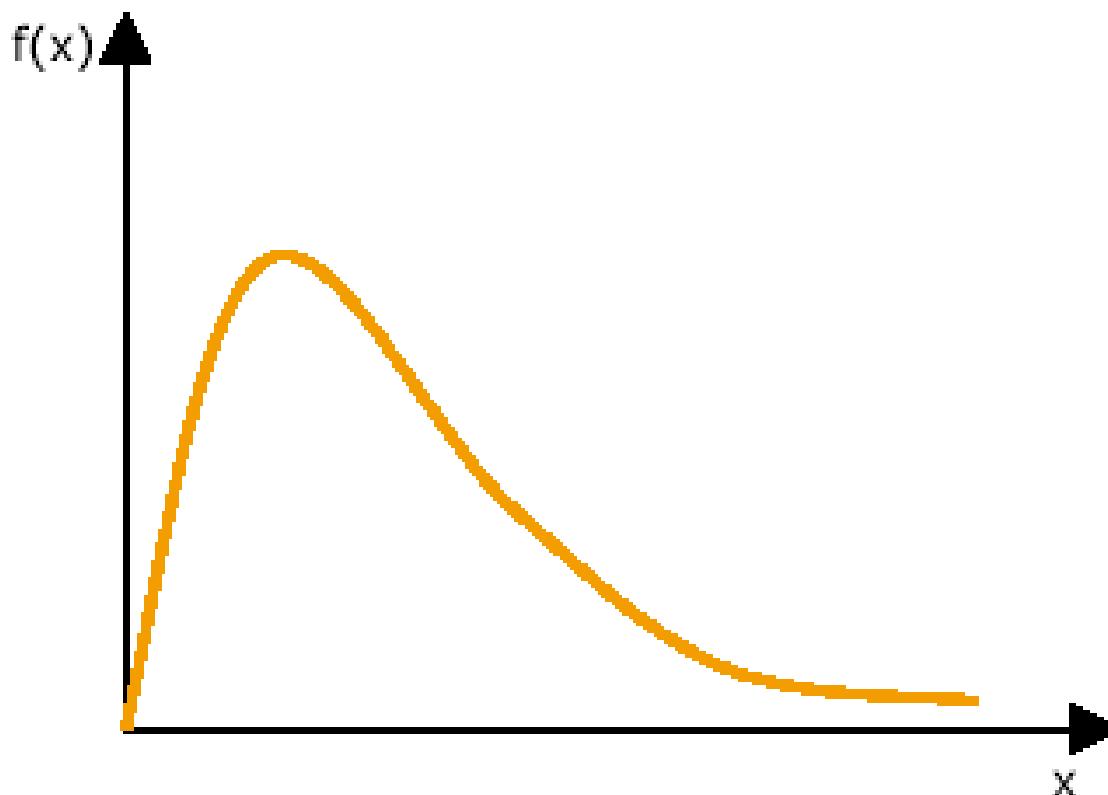


# Distribuições de Probabilidade

## Distribuições Contínuas

### Lognormal

Valores que são o resultado da multiplicação de grande número de outros valores.

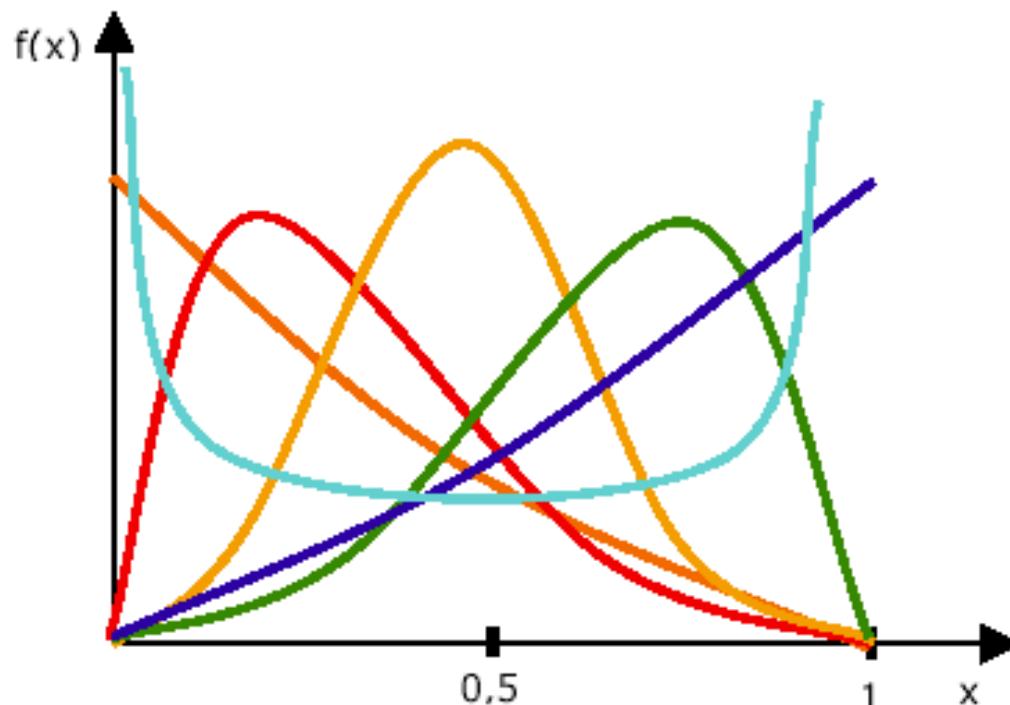


# Distribuições de Probabilidade

## Distribuições Contínuas

### Beta

Distribuição de proporções aleatórias.

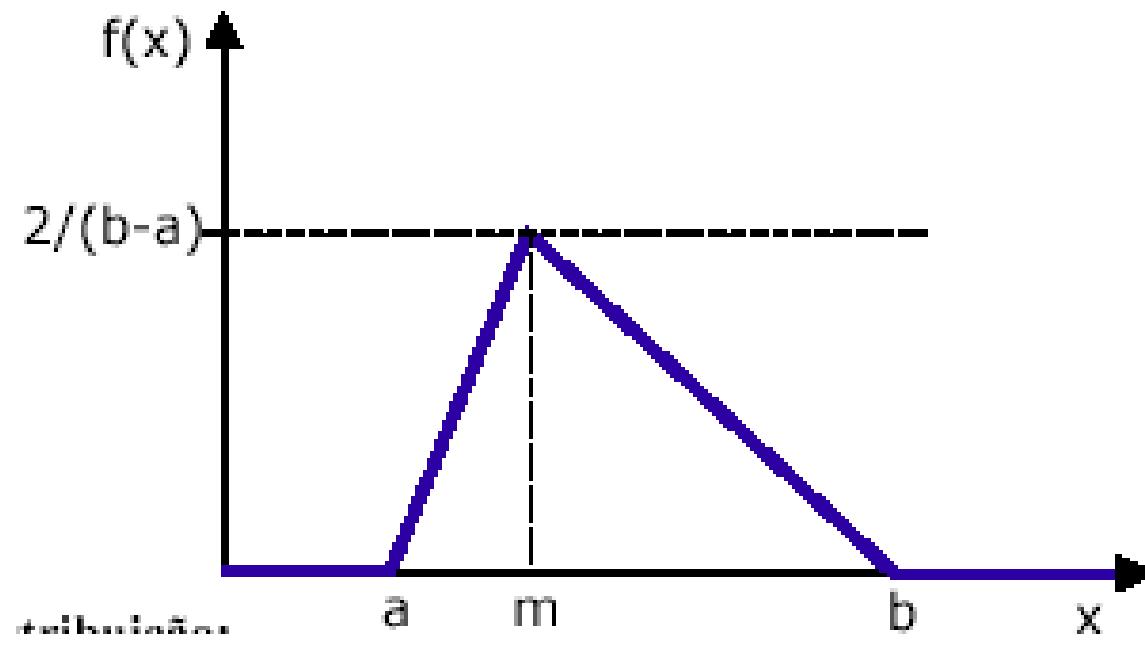


Gráficos da Distribuição Beta( $\beta, \alpha$ )

# Distribuições de Probabilidade

## Distribuições Contínuas Triangular

Aproximação de  
dados que  
permitam obter  
uma distribuição  
adequada.



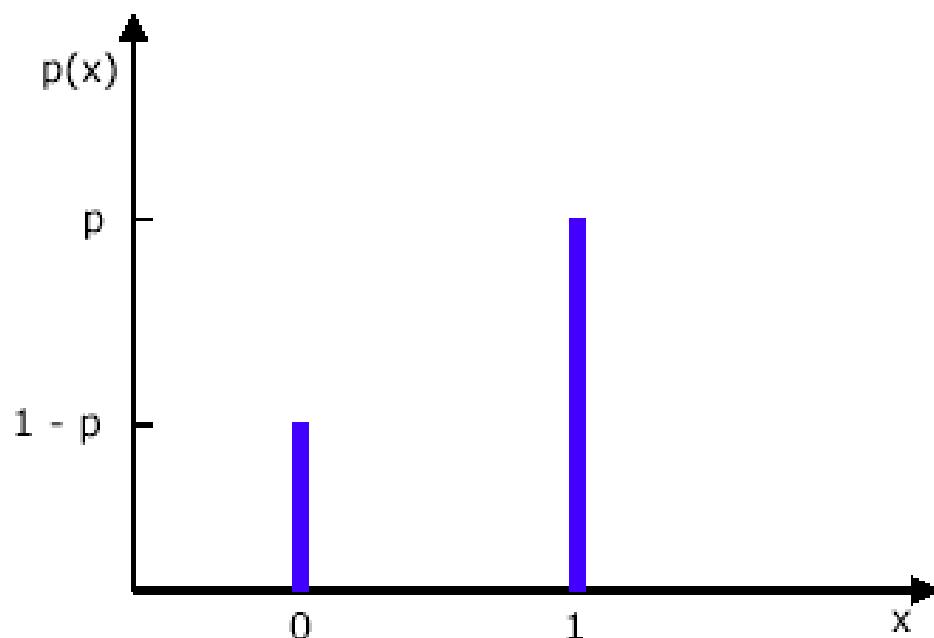
# Distribuições de Probabilidade

## Distribuições Discretas

### Bernoulli

$$p(x) = \begin{cases} 1 - p & \text{se } x = 0 \\ p & \text{se } x = 1 \\ 0 & \text{caso contrário} \end{cases}$$

Ocorrência aleatória onde são possíveis apenas dois resultados.



# Distribuições de Probabilidade

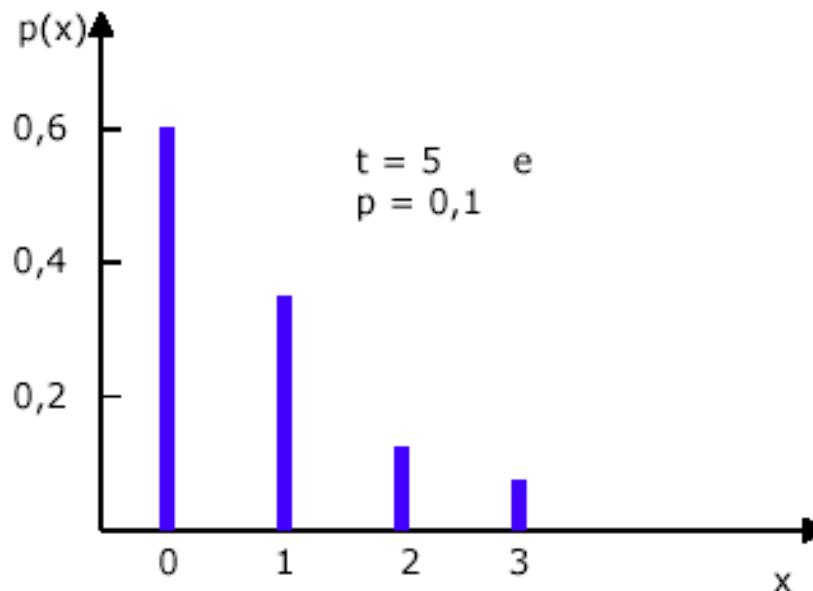
## Distribuições Discretas

### Binomial

$$p(x) = \begin{cases} \binom{t}{x} p^x (1-p)^{t-x} & \text{se } x \in \{0,1,2,\dots,t\} \\ 0 & \text{caso contrário} \end{cases}$$

Número de  
sucessos em t  
tentativas  
independentes.

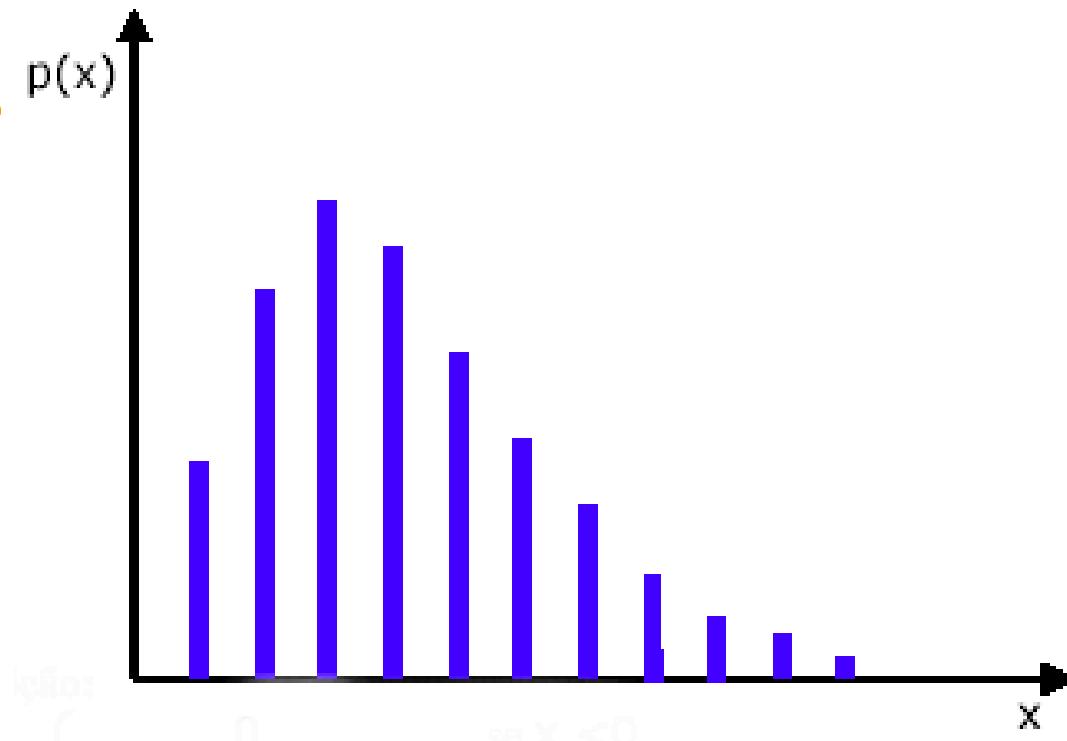
$$\text{onde } \binom{t}{x} = \frac{t!}{x!(t-x)!}$$



# Distribuições de Probabilidade

## Distribuições Discretas Poisson

Eventos aleatórios  
que ocorrem com  
uma freqüência  
média  $\lambda$   
conhecida.



# Tipos de Sistemas

- ” Dois tipos de sistemas (modelos)
  - . Sistemas terminantes
    - ” Condição de término bem definida (o sistema deixa de operar)
    - ” Corrida definida por N tentativas
  - . Sistemas não terminantes
    - ” O sistema, em teoria, poderia operar indefinidamente
    - ” Estuda-se o comportamento estacionário o sistema
    - ” Corrida definida por uma (longa) duração T
    - ” Estimativas resumem valores observados durante o período simulado.

# Elementos do modelo

## ” Entidade:

- . Elemento básico
- . Identificado e processado individualmente
- . Pode ser
  - ” PERMANENTE - sempre presente durante o período simulado
  - ” ou TEMPORÁRIO - definido por uma fonte ou origem

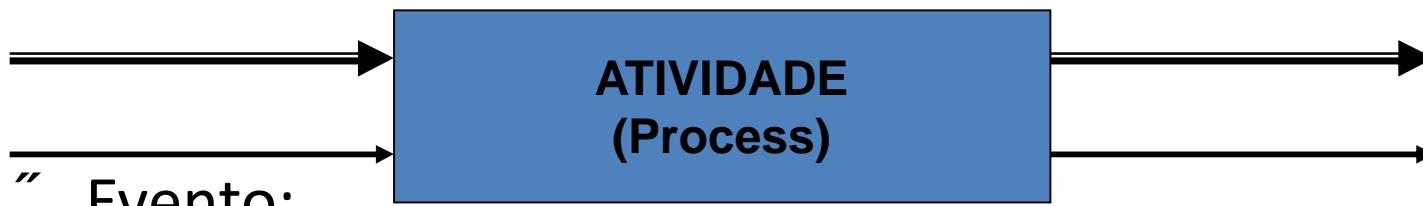
## ” Atributo:

- . Informação associada a uma entidade
- . Pode ser modificado ao longo do tempo!
  - ” Os atributos são acionados no Bloco Assign
    - . Tipo de assing – attribute
    - . Define-se o nome do atributo e o seu “valor”.

# Elementos do modelo

## ” Atividade:

- . Ocupa temporariamente uma ou mais entidades.
- . A duração é conhecida *a priori*



## ” Evento:

- . Momento de mudança do estado do sistema
  - ” Início de atividade e Término de atividade – são as especificações do Delay

# Elementos do modelo

- ” Fila e disciplina de atendimento:
  - . Estado passivo de uma entidade aguardando o início de uma atividade
  - . Disciplinas: FIFO, LIFO, SPT (shortest processing time, ...)
  - . O tempo de permanência na fila não é pré-determinado, é o resultado de uma simulação.
- ” Fonte/sumidouro:
  - . Local de criação – Bloco Create
  - . Destrução de entidade temporária – Bloco Dispose

# Elementos do modelo

- ” Roteamento da entidade:
  - . Caminho seguido por uma entidade ao término de uma atividade;
  - . Decisão por atributo ou de caráter aleatório;
- ” Relógio da Simulação:
  - . Avanço do tempo a intervalos variáveis;
- ” Histograma:
  - . Variáveis observadas (coleta de estatísticas)

## ” FORNECENDO DADOS AO ARENA

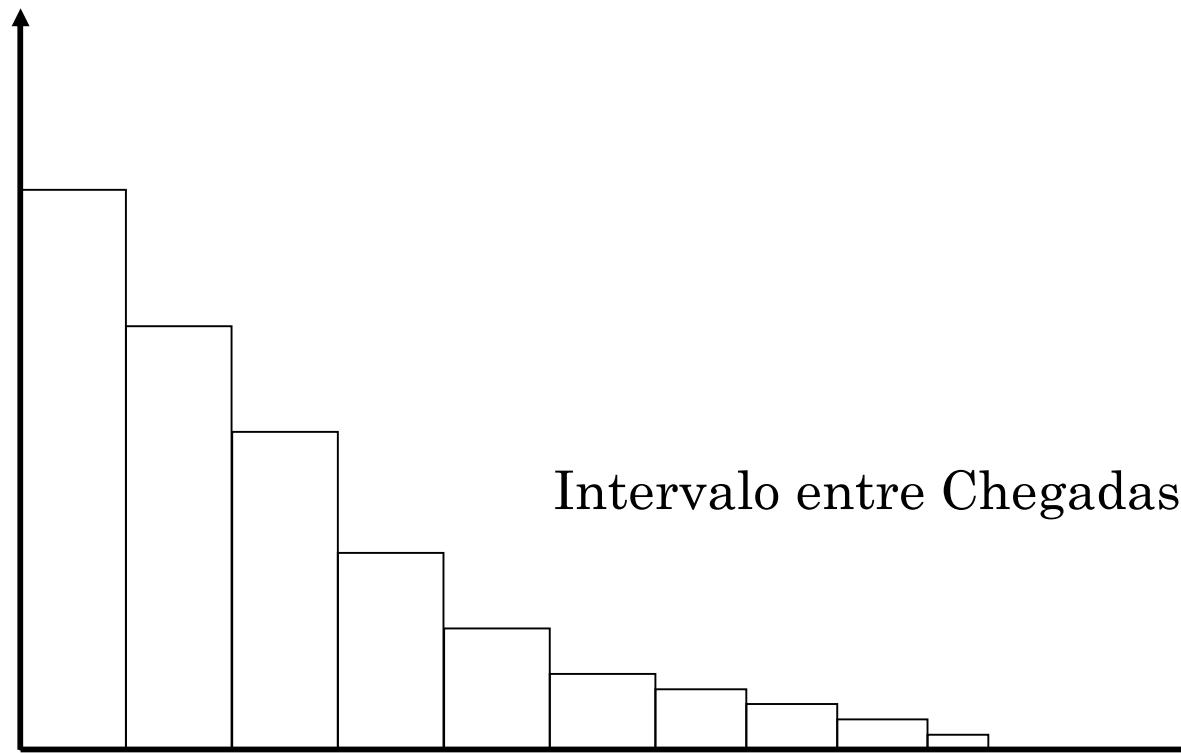
Para montarmos um modelo em Arena, devemos fornecer informações sobre o que acontece em cada estação de trabalho, sobre o deslocamento entre as estações, etc.

### 1. O processo de Chegada.

O estabelecimento do processo de chegada de entidades ao sistema que está sendo simulado é uma etapa muito importante na criação do modelo. Por processo de chegada geralmente estaremos nos referindo a uma distribuição de probabilidades que descreve corretamente a chegada de clientes ao sistema. Cada caso deve ser analisado individualmente, mas os mais comuns são:

- a) Os intervalos entre chegadas seguem a distribuição exponencial negativa.
- b) Os intervalos entre chegadas seguem uma tabela que descreve o processo.

O primeiro caso é bastante comum: dentre as distribuições estatísticas, a exponencial negativa se adapta a quase todos os processos de chegada.



Assim, para expressar no *Arena* que clientes chegam a um sistema a cada 5 minutos, segundo a distribuição exponencial negativa, diremos que o valor para Time Between Arrivals é  $EXP(5)$ .

## 2. O Processo de Atendimento

Ao chegar a uma estação de trabalho, a entidade sofre um atendimento durante um período de tempo. Para este caso não existe uma distribuição estatística que se adapte a todos os cenários. Pelo contrário, cada cenário deve ser analisado individualmente. As possibilidades teóricas são:

- A distribuição de Erlang.
- A distribuição exponencial negativa.
- A distribuição triangular.
- A distribuição retangular ou uniforme.
- Etc.

Assim, por exemplo, para expressar no Arena que o atendimento em uma dada estação de trabalho segue a distribuição triangular, podemos dizer que o valor para o *Process Time* é TRIA(10,15,25), em que:

- Valor mínimo = 10
- Moda = 15
- Valor máximo = 25

Além disso, é muito comum o caso em que nenhuma das distribuições teóricas se adapta a um caso real. Então devemos utilizar os próprios dados reais para a simulação.

### 3. O Deslocamento entre estações

Os dados de durações do deslocamento podem contemplar:

- deslocamento efetuado pelo próprio cliente.
- deslocamento efetuado por um equipamento (como uma ponte rolante ou uma esteira).

No primeiro caso geralmente devemos fornecer uma distribuição de probabilidades, semelhante às utilizadas no processo de atendimento. No segundo caso devemos fornecer os dados de funcionamento do equipamento.

## 4. A Programação Visual

Um modelo em *Arena* é constituído de duas partes:

- Lógica
- Animação

### LÓGICA:

Nesta parte montamos um programa, utilizando comandos (também chamados de blocos ou módulos) do *Arena*.

### ANIMAÇÃO:

Nesta parte são colocados desenhos e símbolos para representar as estações de trabalho e os caminhos por onde passa a entidade. O *Arena* simula a evolução do tempo e movimenta a entidade pelos caminhos e estações.

## 5. A Execução do Modelo

Quando um modelo é executado, o *Arena* vai criando entidades e movimentando-as entre as estações de trabalho. **O *Arena* simula e gerencia o transcorrer do tempo:** a cada instante, no tempo que está sendo simulado, algum evento pode acontecer, tal como:

- um novo cliente chega ao sistema.
- uma entidade inicia o deslocamento entre duas estações de trabalho.
- um servidor de uma estação de trabalho inicia o atendimento a um cliente.

Para execução, o *Arena* se baseia na lógica da programação fornecida para o modelo. Ele se encarrega de manusear todos os dados surgidos na própria simulação, tais como tempo de espera na fila, taxa de utilização de atendentes, etc. Além disso, ele faz com que a animação na tela tenha um aspecto próximo à realidade. Ao final da simulação, ele disponibiliza relatórios que mostram os principais resultados do processo.

## O Método de Monte Carlo

Quando uma cliente chega a uma estação de trabalho, a duração do atendimento daquele cliente geralmente é diferente do atendimento dos outros clientes, visto que o processo é randômico e descrito por uma distribuição de probabilidades. Para descobrir qual a duração do atendimento daquele cliente específico, o *Arena* utiliza o Método de Monte Carlo. Por este método é efetuado um sorteio para se encontrar a duração do atendimento. O mesmo método é efetuado para o processo de chegada, para o deslocamento entre estações de trabalho, etc.

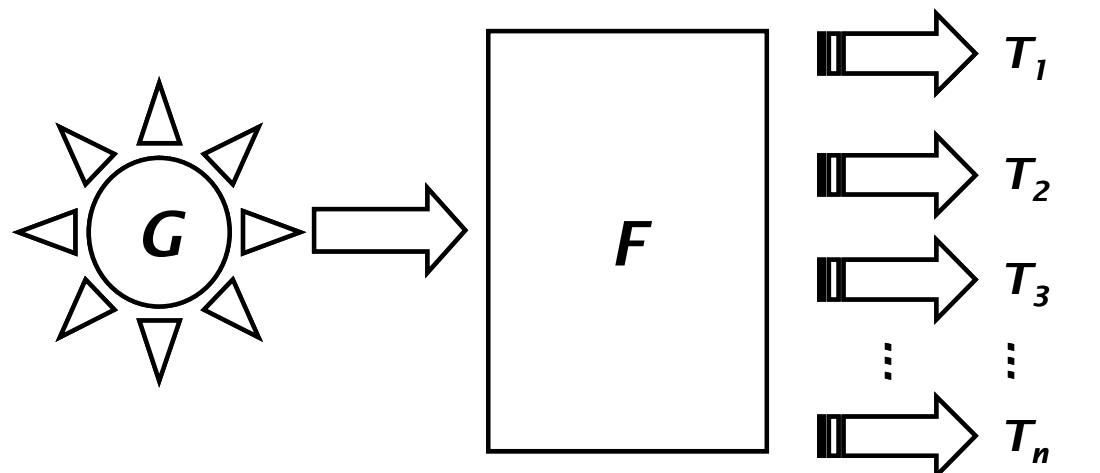
O Método de Monte Carlo utiliza números aleatórios e a função de probabilidades que descreve o fenômeno real. Para que a simulação forneça resultados confiáveis, é necessário que a amostra simulada seja de tamanho adequado, o que significa que, se o tamanho da amostra for aumentado, os resultados não se alteram. O uso de amostras de pequeno tamanho pode fornecer resultados diferentes do real.

# Geração de Números Aleatórios

- ” Um número é aleatório pode representar decisões arbitrárias ou servir como entrada para geração de tempos segundo várias distribuições.
- ” Como produzir números aleatórios ?
  - . Dispositivos físicos (Ex. dados, roleta, moeda etc.)
  - . Tabela de números aleatórios (livros)
  - . Processos matemáticos
- ” No Excel: “=ALEATORIO()” (gera um número aleatório maior ou igual a 0 e menor do que 1)

# Geração de Números Aleatórios

- Sorteia-se um número aleatório uniformemente distribuído entre 0 e 1.
- Utilizando-se a função repartição da distribuição de probabilidade desejada, transforma-se o número aleatório uniformemente distribuído em um valor segundo a distribuição probabilística desejada.



*Gerador de  
número aleatório*

*Função de repartição:  
Exponencial, Normal,  
etc*

*Tempos  
segundo as  
distribuições*

# Geração de Números Aleatórios . Método do Meio Quadrado

” Von Neumann (1946)

- .  $r_1 = 76 \Rightarrow 76^2 = 5776$
- .  $r_2 = 77 \Rightarrow 77^2 = 5929$
- .  $r_3 = 92\dots$

” Seqüência gerada (76,77,92,46,11,12,14, ...)

” Quando resultar em 0, deve-se utilizar outra semente.

# Geração de Números Aleatórios - Congruência

$$x_{i+1} = (ax_i + c) \bmod m$$

***gera números  
inteiros entre 0 e m-1***

- $x_0$  é a semente do número aleatório
- “mod” é a função módulo = mostra o resto da divisão inteira. Ex.:  $10 \bmod 6 = 4$

# Geração de Números Aleatórios

Passo 1: *Escolher os valores  $a$ ,  $c$  e  $M$ . Usualmente,  $M$  é escolhido o maior possível.*

Passo 2: *Escolher a semente  $r_0$ , tal que:  $1 \leq r_0 \leq M$ .*

Passo 3: *Calcular o próximo número aleatório pela expressão:*

$$r_1 = (a \cdot r_0 + c) \bmod M$$

*onde:  $x \bmod y$  é o módulo da divisão de  $x$  por  $y$  (por exemplo:  
 $10 \bmod 6 = 4$ ).*

Passo 4: *Substitua  $r_0$  por  $r_1$  e volte ao passo anterior, de modo a construir a seqüência de números aleatórios desejada.*

# Geração de Números Aleatórios

Gerar números aleatórios pelo método da congruência, com  $a = 9$ ,  $c = 1$ ,  $m = 17$  e  $x_0 = 7$ .

<b><i>n</i></b>	<b><i>X<sub>n</sub></i></b>	<b><i>y=9x<sub>n</sub>+1</i></b>	<b><i>y mod 17</i></b>	<b><i>x<sub>n+1</sub>/17</i></b>
0	$X_0=7$	$9*7+1=64$	13	$13/17 = 0.7647$
1	$X_1=13$	118	16	$16/17 = 0.9412$
2	$X_2=16$	145	9	0.5294
3	$X_3=9$	82	14	0.8235
4	$X_4=14$	127	8	0.4706

*números pseudo-aleatórios inteiros entre 0 e 16 (=17-1)*

*números pseudo-aleatórios inteiros entre 0 e 1*

# Geração de Números Aleatórios .

## Transformação Inversa

Passo 1: *Obtenha a função de repartição da variável aleatória através da expressão (Apêndice I):*

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

Passo 2: *Gere um número aleatório r entre 0 e 1.*

Passo 3: *Faça  $F(x)=r$  e resolva em x. A variável x é uma variável aleatória cuja distribuição é dada pela função densidade de probabilidade  $f(x)$ .*

# Trabalhando com o Arena

# Trabalhando com o Arena

- ” inicie o Arena
- ” carregue, explore e rode um modelo existente
  - . basicamente, o mesmo modelo da simulação manual do capítulo anterior;
- ” Arena é um aplicativo do Windows
  - . aparência padrão;
  - . operação, funções são todos padrões;
  - . inter-operacionalidade como outros *software*, p.e., MS Office;
  - . gerência de memória própria;

# Começando

- ” localize o ícone ou atalho; duplo clique;
  - . chave de *hardware* vs. Versão educacional/avaliação;
- ” janela de abertura (welcome):
  - . dica do dia (forma de conseguir ajuda do sistema);
  - . entradas alternativas para:
    - ” introdução- visão rápida, inicio rápido, assistente, bibliotecas, exemplos, representante comercial;
    - ” versão;
  - . dicas no futuro? (usuário experiente? nenhuma?);
  - . botão para fechar
- ” finalizando Arena: File/Exit, Alt+F4 ou botão 

# Abrindo um Modelo Existente

- ” File/Open ... ou botão 

  - . navegue até o disco/diretório desejado;
  - . clique Open ou dois cliques em **Mod\_03\_1.doe**

- ” janela do modelo:
  - . onde o modelo é desenvolvido;
  - . redimensione, maximize, minimize, movimente, *zoom*;
  - . suporta múltiplos modelos de uma vez;
- ” barra de *templates* à esquerda:
  - . disponibiliza modelos;
  - . alterna níveis de modelagem;
  - . anexa outros painéis;

# Visualizando a Janela do Modelo

- ” disponibiliza *espaço* para o modelo;
- ” *movimenta* com as barras e setas do teclado;
- ” *reduz* vista com o botão  ou a tecla -
- ” *aumenta*, com o botão  ou a tecla +
- ” apresenta no maior tamanho possível com o botão  ou a tecla \*
- ” *rótulos das vistas*:
  - . salva vistas de diferentes partes do modelo;
  - . atribui uma tecla rápida (*hot key*);
  - . acessível via View/Named Views ou tecla ?
- ” apresenta grade com o botão



# Interação Básica

## ” aplicação Windows :

- . comportamento usual do mouse, teclado, seleção, arrasto, ajuste de tamanho, botões, menus;
- . operações usuais de janela - redimensionamento, camadas, maximização, minimização, fechar;

## ” operações usuais de arquivos:

- . novo, abrir, salvar, salvar como, imprimir, visualização de impressão;

## ” cortar/copiar/colar:

- . dentro de uma janela do Arena;
- . entre as janelas do Arena;
- . entre o Arena e outros aplicativos (quando ativo).

# Interação Básica (cont.)

” opções gerais:

- orientação por objetos: selecione um objeto, então atue sobre o mesmo;
- desfaça/refaça 
- botão direito do *mouse*:
  - ” repete última ação;
  - ” abandona ação corrente;
- desenhando: segure a tecla Shift para restringir linhas a 45°, retângulos a quadrados, elipses em círculos, etc.;
- percorrendo as janelas do Arena: Ctrl+Tab
- duplicando uma seleção: Ctrl+D — então movê-la e edita-la.

# Menus

- ” File, Edit, View Tools, Arrange, Module, Run, Window, Help
  - . se a janela de modelo não estiver aberta: apenas File, View, Tools, Help
- ” operações padrão do Windows nos menus usuais:
  - . menu File tem Open, Save, Save As, etc.;
  - . menu Edit tem Cut, Copy, Paste, etc.;
  - . menu Help;
  - . etc.;
  - . assim, serão abordas características nos menus específicas do Arena;
- ” muitos itens do menu possuem sub-itens;

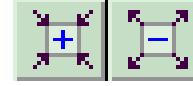
# Menu File

- ” New, Open..., Save, Save As... 
- ” Importação de arquivos CAD DXF;
- ” paleta de cor;
- ” Print, Print Preview 
- ” Send mail
  - . anexa arquivos com extensão .doe (extensão própria do Arena);
- ” Recent models
- ” Exit Arena

# Menu Edit

- ” Undo/Redo 
- ” Cut, Copy, Paste 
- ” Paste Link
- ” Duplicate, Delete — duplica e apaga objetos;
- ” Select All, Deselect All
- ” Find — procura, em todos os módulos do modelo corrente, por seqüências (para “esquecidos” ou datilógrafos com dificuldades);
- ” Links — informações e ligações com outros tipos de documentos;
- ” Insert — edita objetos de outras aplicações;

# Menu View

- ” aumentado/diminuindo tamanho 
- ” Named views
  - . pré-definidas pelo Arena,
  - . definidas por você;
- ” Grid, Smap, Setteings 
- ” Layers — o quê os objetos mostrarão em qual modo; 
- ” Toolbars — qual conjunto de “botões” serão mostrados;
- ” Status bar — estado, coordenadas.

# Menu Tools

- ” separa aplicações do Arena para construção e análise;
  - . Input Analyzer
  - . Output Analyzer
- ” Scenario Manager
- ” License
- ” editor Visual Basic
- ” Options... — controla aspectos do Arena, tais como aparência, etc.

# Menu Arrange

- ” desenho de objetos 
- ” traz objeto selecionado à frente e atrás; 
- ” agrupar/desagrupar (Group/Ungroup) objetos;
- ” efeito espelho no objeto (Flip); 
- ” gira objeto; 
- ” alinha objeto com a rede (Grid);
- ” muda pontos em objetos;

# Menu Module

“ controles de conexão: 

- . ferramenta de conexão;
- . conexão automática de módulos recentemente adicionados ao modelo (botão liga-desliga—toggle on/off);

# Menu Module (cont.)

- ” Template Panel
  - . Attach (conecta); 
  - . Detach (desconecta);
- ” Show — alterna entre o nome do módulo ou simplesmente a sua animação;
- ” Breakpoint — conectar ao módulo para interromper durante simulação (para correção de erros, “debugging”). 

# Menu Run

- ” entradas para rodar, checar, pausa, execução um passo à frente;
- ” alternativas para visualizar a execução, resultados (ou erros);
- ” controla como a execução será feita e apresentada;

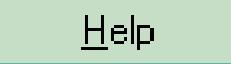
# Menu Window

- ” modos de exibição de janelas;
- ” arranjar ícones;
- ” cor de Background;
- ” lista de janelas de modelos abertas;

# Barra de Ferramentas

- ” coleção de botões para operações freqüentes:
  - . muitas (mas não todos) são duplicações de itens dos menus;
- ” Standard, View, Arrange, Draw, Color, Animate, Template, Run, Run Interaction, Integration
- ” *View/Toolbars* (ou botão direito em uma área de barra de ferramentas) para decidir quais serão apresentadas e escondidas;
- ” o Arena se lembrará das configurações na próxima chamada;

# Help

- „ extensivo e comprehensivo sistema *online* — substitui os manuais tradicionais;
- „ interligado via hipertexto, para referência cruzada;
- „ múltiplas entradas;
- „ tela de entrada (*Welcome*);
- „ botão  para ajuda dependente do contexto;
- „ botão  em muitas caixas de diálogo;
- „ botão  (O quê é isto?) para informações sobre itens;
- „ dicas — permaneça com o *mouse* imóvel e apanhe um pequena nota de ajuda;

# Janela de Modelo

- “ onde o modelo é construído, editados e rodado;
- “ pode haver várias aberta simultaneamente;
- “ “janela do mundo” — coordenadas (x, y);
- “ abrir um modelo existente (extensão .doe);
- “ criar modelos novos (em branco);
- “ salvar e salvar como (Save, Save As...);
- “ adiciona módulos ao modelo — clique na barra de *template* e leve ao modelo;

# Desenho

” desenhar somente via botões:



- ” linhas, poligonais, arcos, curvas, etc.;
- ” caixas, polígonos, elipses;
- ” texto (fonte, tamanho, estilo, cor);
- ” camadas visíveis/invisíveis — mostradas durante animação?
- ” muda posição, tamanho, cor;
- ” melhor forma de aprender: experimentando!

# Impressão

- ” direta do Arena;
- ” suporta cor;
- ” modelos grandes — muitas páginas;
- ” também imprime vistas parciais do modelo:
  - . visualize a impressão e selecione somente aquelas de interesse;
- ” tecla PrintScreen — envia à área de transferência (*clipboard*), para ser anexada a outra aplicação:
  - . Alt+PrintScreen — envia somente a janela ativa.

# Execução

- ” Menu Run; barra de ferramentas Run;
- ” *Go* (▶) — executa a simulação normalmente (depende das opções selecionadas no item Options... do menu Tools);
- ” *Step* (▶) — um passo por vez (verificação, correção);
- ” *Fast-Forward* (▶) — desabilita a animação (rapidez);
- ” *Pause* (⏸) — congela a execução, retorna com Go;
- ” *Start Over* (⏮) — recomeça simulação do início;
- ” *End* (⏹) — sai do modo de execução.

# Execução (cont.)

- ” *Check Model* (  ) — como um compilação;
- ” *Review Errors* — visualizar erros da compilação mais recente;
- ” *View Results* — sumário da última execução;
- ” SIMAN — visualiza, salva arquivos extensão .mod e .exp, gerados pelo Arena para seu modelo;
- ” *Command* (  ) — ativa janela de linha de comando;
- ” *Break...* (  ) — especifica tempos, condições para interrupção, para conferir resultados, ilustração;

# Execução (cont.)

- ” *Trace...* (  ) — geração de informações detalhadas (correção de erros);
- ” *Watch...* (  ) — ativa janela para monitoramento de uma variável ou expressão durante execução;
- ” *Report...* (  ) — visualiza situação numérica;
- ” *Setup...* — altera condições de execução (liga/desliga animação, tela cheia durante execução);
- ” *Speed* — controla a velocidade da animação

## “Construindo Modelos no *Arena*

Neste tópico, apresenta-se uma visão geral de como são construídos os modelos de sistema, para posterior simulação, utilizando-se o ambiente do *Arena*.

O *Arena*, e a linguagem ou ambientes de modelagem e simulação de propósito geral, isto é, os sistemas a serem modelados podem pertencer a qualquer área de aplicação, tais como: manufatura, serviços, sistema de transportes, logística, redes de computador, etc.

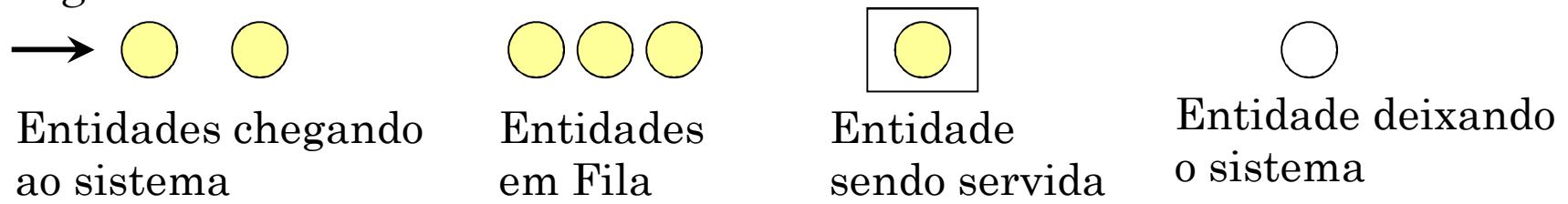
### Projeto 1

Para a introdução dos conceitos básicos de construção de modelos no *Arena*, aborda-se, inicialmente, um exemplo genérico. O exemplo trata de um sistema com um único servidor no qual as entidades chegam, aderem à fila do servidor (se o mesmo estiver ocupado), realizam um processo (despendendo algum tempo com o servidor) e deixam o sistema.

Durante o desenvolvimento do modelo, serão discutidos conceitos e terminologias já referenciadas, de maneira que se possa fazer ligações e analogia entre a teoria, e a prática de modelagem.

A seguir podemos ver a representação esquemática do sistema exemplo que será modelado.

Figura 1



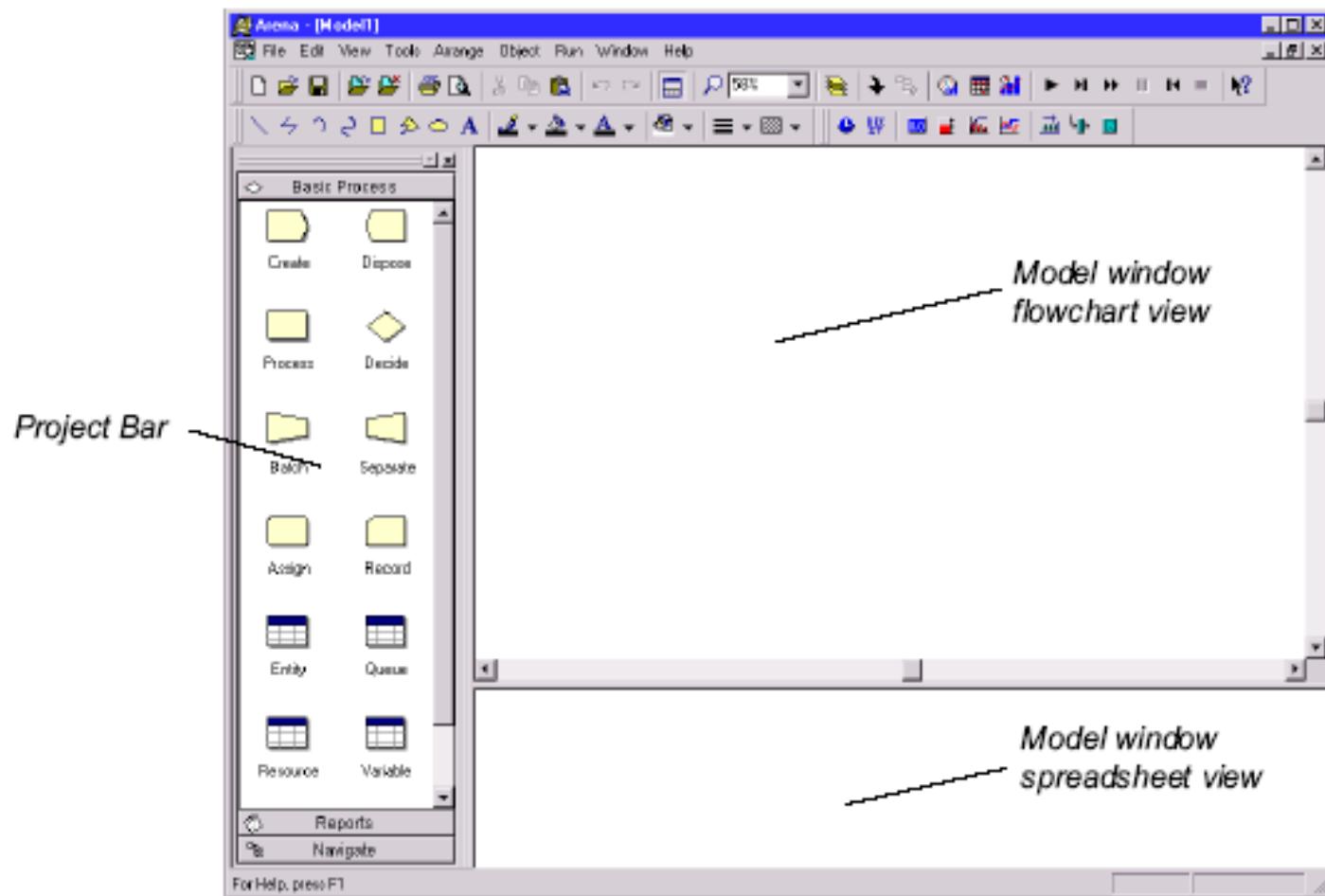
Para modelar este Sistema no *Arena*, as seguintes tarefas devem ser executadas:

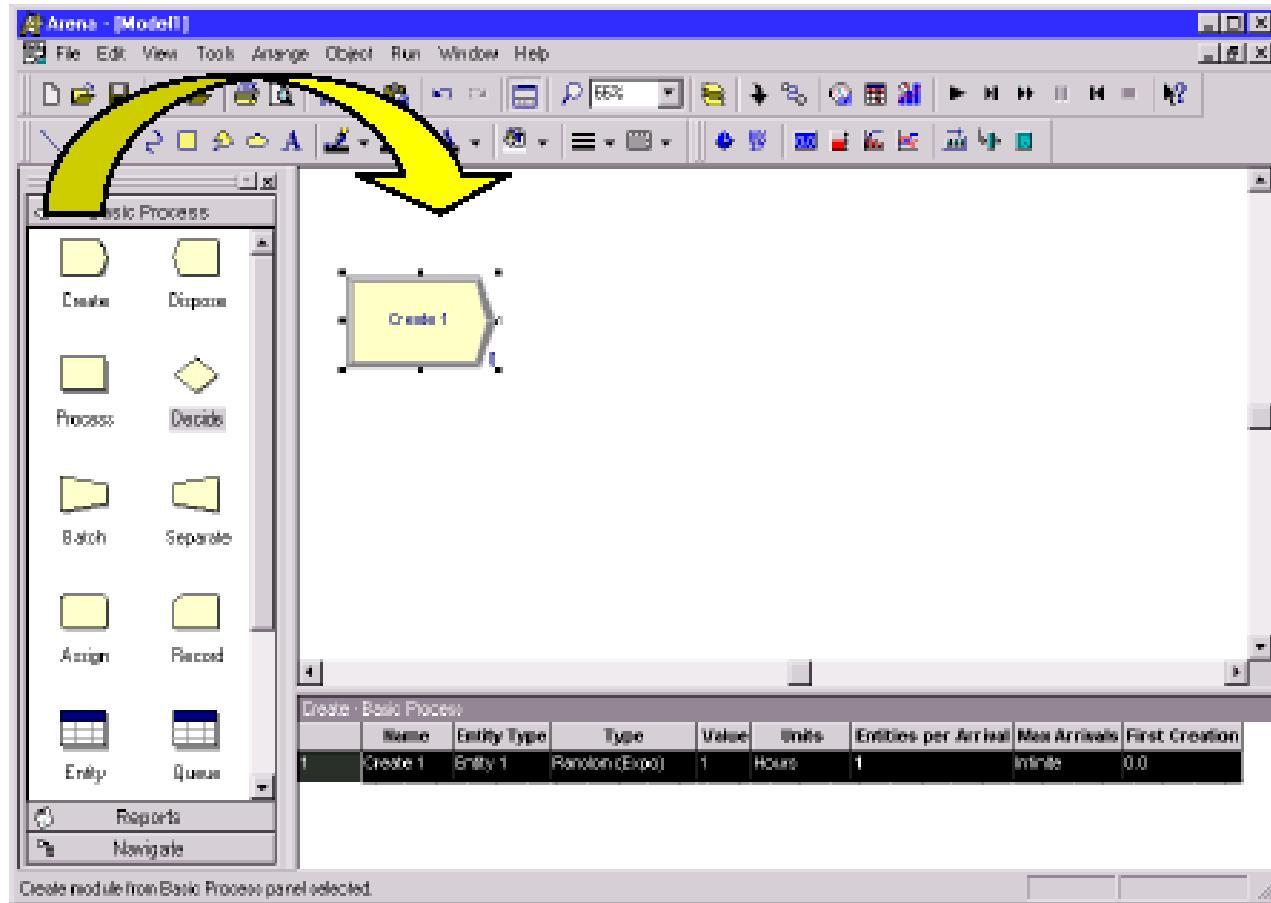
1. Colocar e interconectar módulos para representar o sistema de acordo com sua lógica de operação, obtidos a partir do *template Basic Process*;
2. Fornecer dados para o modelo;
3. Executar a simulação do modelo.

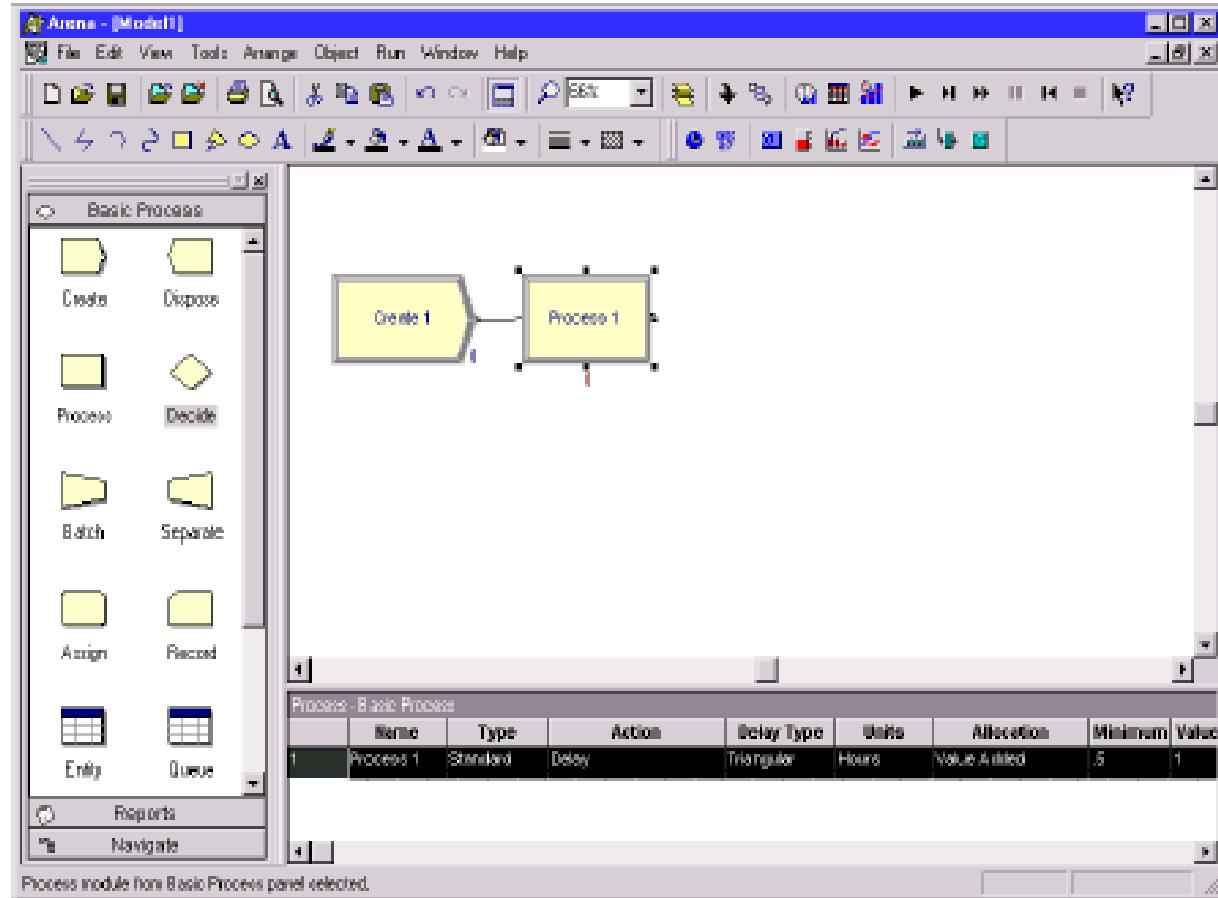
## **1. Anexando os Módulos na Área de Trabalho**

No Arena, a modelagem é feita considerando uma visão dos processos do sistema. A maneira mais simples de entender esta forma de modelar é assumir o lugar de uma entidade dinâmica, e entender como esta interage com o sistema. A Figura 1 permite ter esta visão. Observa-se, que apenas um tipo de entidade ou cliente entra no sistema. Além disso, verifica-se que todos eles realizam apenas uma atividade: um processo ou serviço junto a um único servidor do sistema. Finalmente, nota-se que uma vez, terminado o processo, todas as entidades deixam o sistema.

Simplificadamente pode-se simbolizar tudo o que pode acontecer com uma entidade neste sistema, através de três processos: a criação ou chegada da entidade ao sistema, o serviço ou processamento que esta realiza ou recebe de um recurso e, finalmente, sua partida.

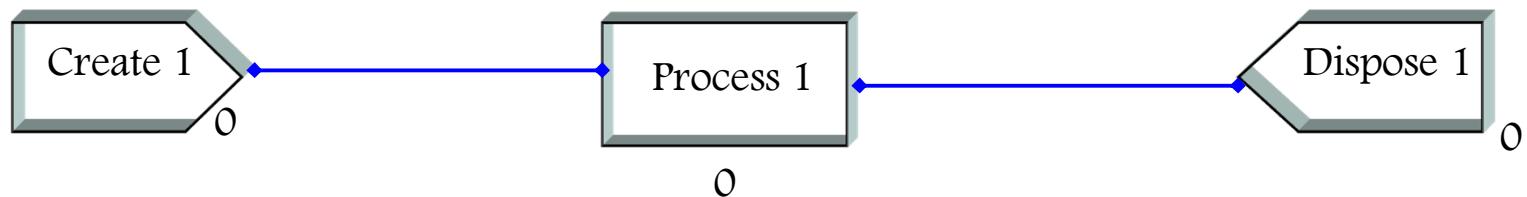






Pensando desta forma, é possível encontrar junto aos painéis ou templates de modelagem do *Arena*, alguns módulos (este é o nome dado aos “comandos” desta linguagem de simulação) que permitem simular estes processos. Na Figura 1, a seguir, é possível observar três dos módulos presentes no templates Basic Process, que permitem a implementação da lógica presente neste sistema.

Figura 2 – A lógica do sistema implementada no *Arena*



Para acrescentar cada um deles no modelo que está sendo criado, basta clicar, arrastar e soltar cada módulo desejado sobre a área de trabalho. Para retirar um módulo, basta marcá-lo (clicando sobre o módulo) e eliminá-lo (tecla *Delete*).

Os três módulos principais anexados ao modelo são:  
*Create, Process e Dispose*.

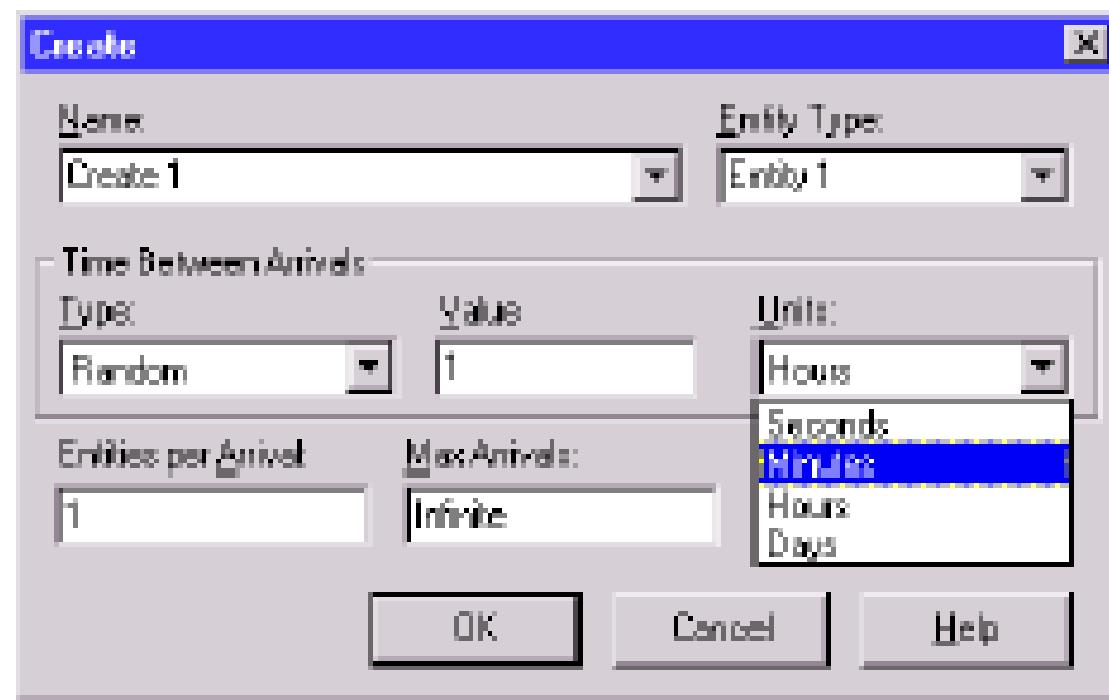
O primeiro é responsável pela criação de entidades no modelo, o segundo pelo processo ou serviço efetuado pelo servidor e o último, pela retirada das entidades do sistema.

Na medida em que os módulos são acrescentados, uma conexão (linha) une os módulos ao outro automaticamente (esta ação automática pode ser desativada). Se uma destas conexões deve ser eliminada, basta marcá-la e excluí-la (tecla *Delete*). Para estabelecer uma conexão entre dois módulos, pode-se usar a função *Connect*, ↗, pr [ ] te na barra de ferramentas.

## 2. Fornecendo Dados ao Modelo

Uma vez que os módulos necessários ao modelo estejam colocados na área de trabalho, o próximo passo é fornecer dados do sistema para que o modelo seja representativo e possa ser válido como ferramenta de análise. Para editar os dados de qualquer dos módulos do *Arena*, basta que se dê um duplo clique sobre a figura do módulo. Inicia-se a edição, executando um duplo clique sobre o módulo *Create*. Uma janela de diálogo se abre. A Figura 3, a seguir, mostra a janela de diálogo do módulo *Create* com seus diversos campos apresentando um preenchimento padrão (*Default*).

Figura 3 – Janela de diálogo padrão (*Default*) do módulo *Create*.



A informação fornecida ao módulo *Create* controla o processo de chegadas. O primeiro campo permite personalizar o módulo, mediante a atribuição de um nome ao mesmo (campo *Name*). As entidades criadas pelo módulo também podem receber nomes que as identifiquem. O segundo campo, denominado *Entity Type*, especifica o nome de um atributo. Assim, cada entidade criada pelo módulo terá um atributo especificando seu tipo.

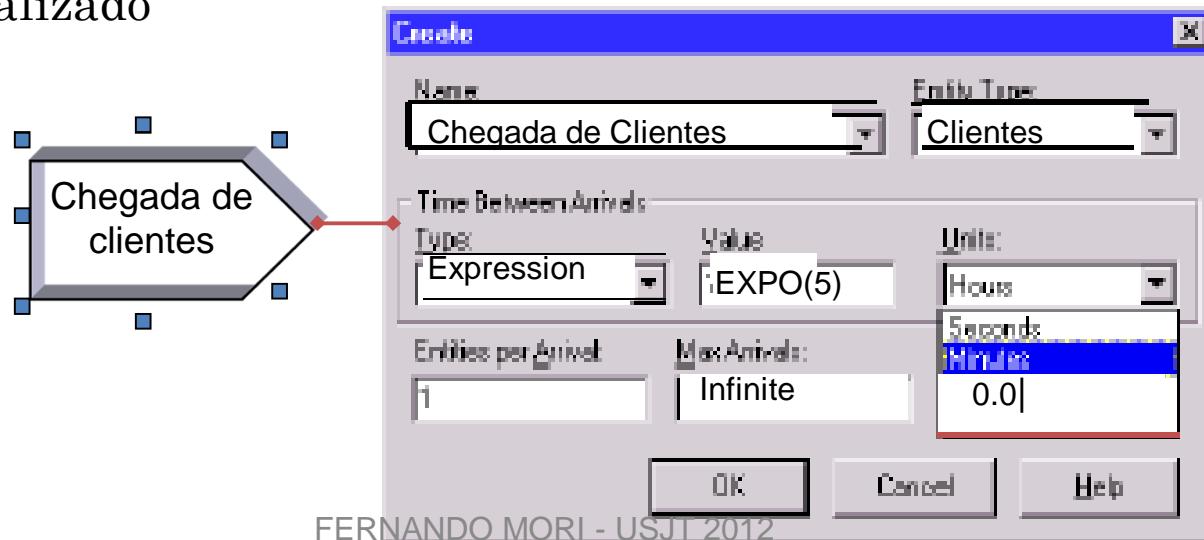
Na Figura 4, a seguir, observa-se que as entidades criadas pelo módulo acima serão classificadas como *Tipo 1 (Entity 1)*. Os nomes podem ser alterados a conveniência de quem está modelando.

Os seis campos restantes tratam efetivamente do controle de chegadas das entidades. Destes, os três primeiros, sob o título *Time Between Arrivals*, que pode ser entendido como a variável *TEC* (*Tempo Entre Chegadas*) anteriormente adotada, especificam, respectivamente, a expressão que define o processo de chegadas, seus parâmetros e a unidade de tempo a ser adotada.

Na Figura 4, *Random* pode ser entendido como sendo uma distribuição *Exponencial*. A média desta distribuição (seu único parâmetro) foi determinada (padrão) como 1 e, a unidade adotada foi *horas*. Já os três campos restantes, definem, respectivamente, quantas entidades serão criadas por cada chegada (padrão é 1), se existe ou não restrição quanto ao número de entidades que podem ser criadas (*Infinite* significa que não há restrição) e quando ocorrerá a primeira criação (no caso, a primeira criação ocorrerá ao início da simulação, tempo zero).

Fig.4 – O módulo *Create*

personalizado

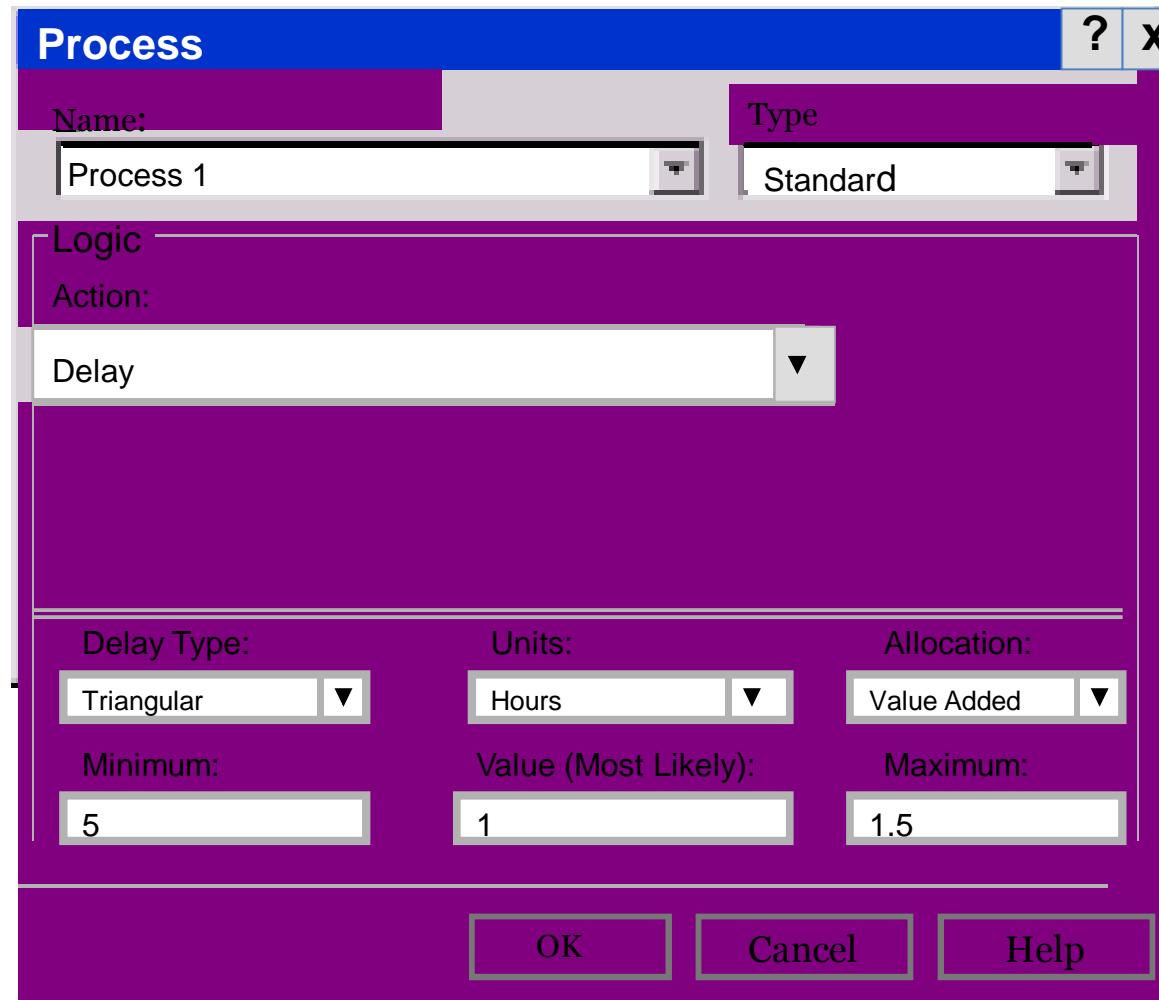


Observe na Fig.4, que o módulo agora possui um nome. Neste caso, o nome utilizado reflete sua função, isto é, gerar a chegada de clientes (entidades). O atributo tipo também reflete à categoria de entidades (clientes) que estão sendo criadas. Nos campos que definem os tempos entre chegadas, vê-se que uma expressão foi empregada. Neste caso, uma distribuição *Exponencial*, com seu parâmetro, média, valendo 5.

O próximo módulo a ser editado é o *Process* (Figura 5). Os procedimentos são semelhantes. Dá-se um duplo clique sobre o módulo para abrir a janela de diálogo.

Da mesma forma que no módulo anterior, o nome padrão *Process 1* pode ser alterado para melhor refletir o processo que será efetivado no módulo. Isso será realizado na seqüência (Figura 6), com o nome sendo alterado para “*Posto de Lavagem*”. No campo *Type*, será aceito o que o programa está sugerindo, *Standard*. Na lista disponível para este campo, a alternativa é criar um submodelo.

Figura 5 – Janela de diálogo padrão (*Default*)  
do módulo *Process*.

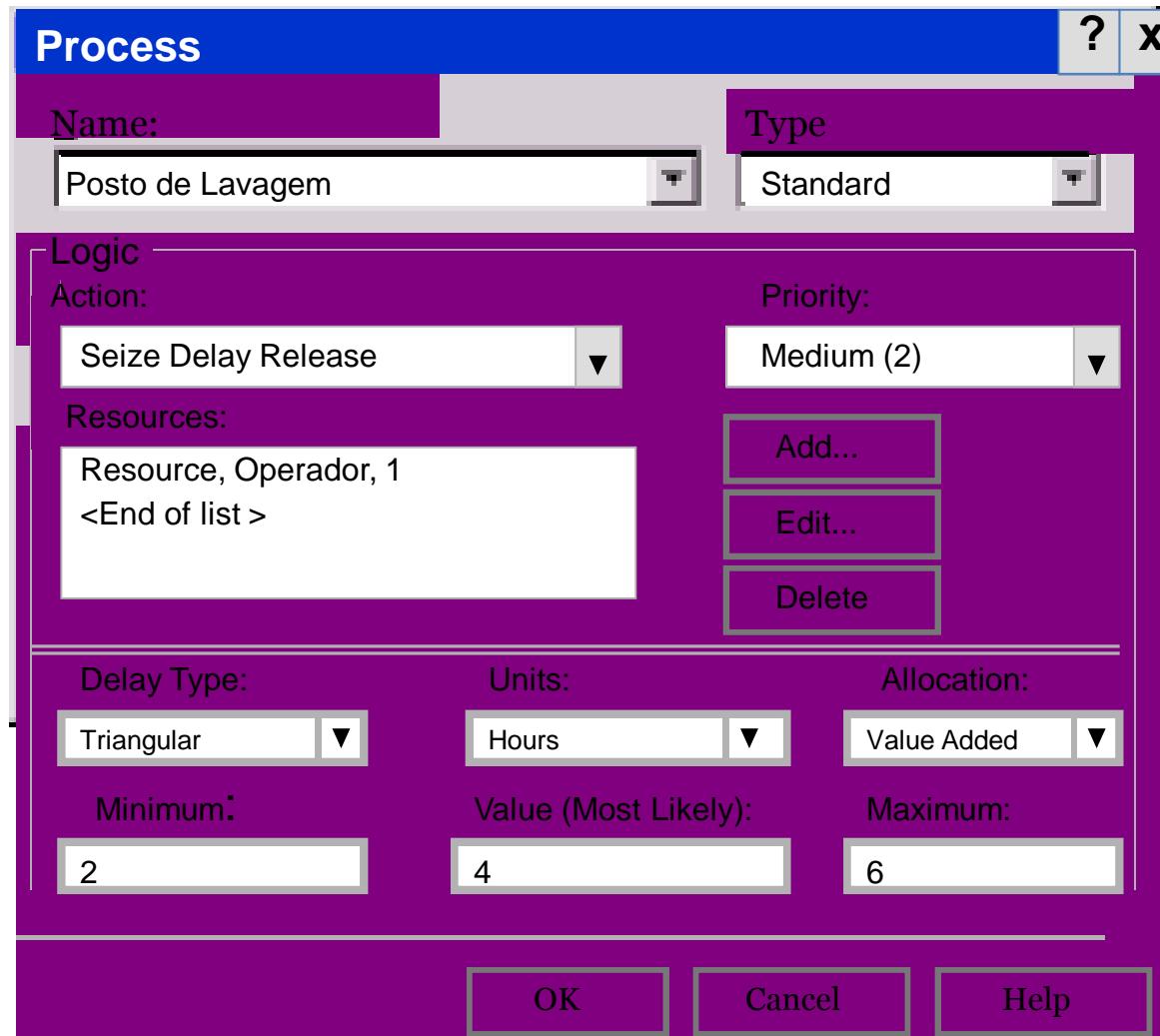


O campo *Action* apresenta como proposta padrão, um *Delay*. Esta ação implica em que a entidade que estiver passando por este módulo realiza uma parada ou espera, de acordo com a informação que é apresentada nos campos seguintes, na parte debaixo do módulo. Nesta forma de modelagem, assume-se que a entidade não necessita de nenhum recurso (uma máquina ou um operador, por exemplo) para realizar esta atividade. Como se o serviço fosse do tipo *self-service*.

No caso de se desejar indicar ao programa que tal processo ou serviço necessita de um recurso do sistema para ser realizado, o campo *Action* precisa ser editado, como se pode ver na Figura 6. Observa-se que o campo *Action* apresenta agora a seqüência *Seize-Delay-Release*. Estas ações indicam que, pela ordem, um recurso será tomado pela entidade ou cliente (*Seize*), este permanecerá com a posse deste recurso por um determinado tempo (*Delay*) e, finalmente, o cliente liberará o recurso (*Release*), uma vez transcorrido o tempo indicado.

Verifica-se, também, que dois novos campos apareceram quando se alterou o campo *Action* (de *Delay* para *Seize-Delay-Release*). O primeiro relativo ao recurso que será utilizado no processo e o segundo, relativo à prioridade na tomada deste recurso. Para o recurso, foi utilizado o nome *Operador*. O número 1, após o nome, indica que, para esta operação, apenas uma unidade deste recurso é necessária. Observe-se que outros recursos poderiam ser também empregados na operação. O campo *Resources* é, na verdade, uma lista. Para determinadas operações, uma troca de óleo de um carro, por exemplo, à lista poderiam ser acrescentados um elevador hidráulico, um recipiente para coleta de óleo usado, ferramentas, etc.

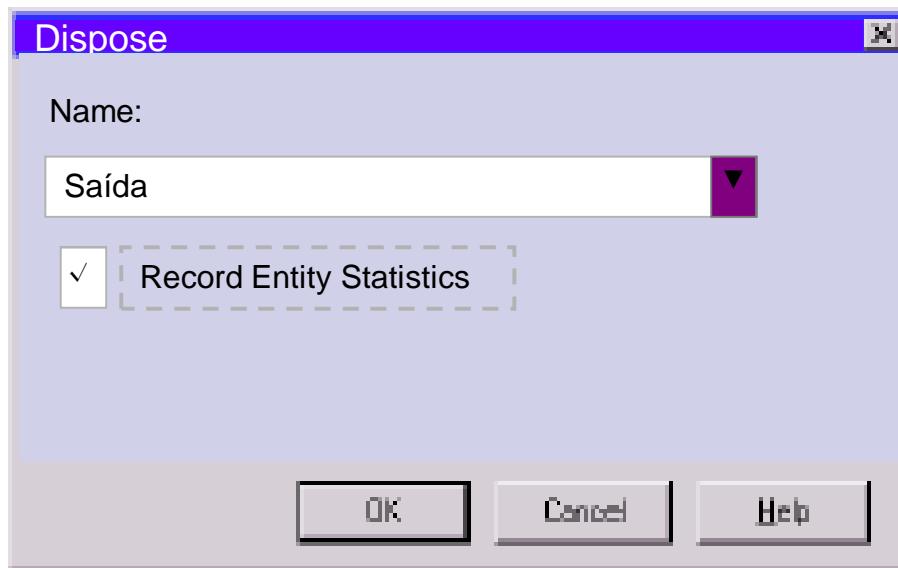
Figura 6 - Janela de diálogo do módulo *Process* editada.



Finalmente, nos campos relativos ao tempo de processamento, adotou-se uma distribuição *Triangular* com parâmetros 2,4 e 6, para os valores mínimos, mais provável (*moda*) e máximo, respectivamente.

No sistema, os clientes, uma vez servidos, deixam o mesmo. No modelo, representa-se tal ação fazendo com que as entidades “deixem” o modelo. Quem realiza este procedimento é o módulo *Dispose*. A figura 7 mostra o mesmo editado. A única modificação realizada foi alterar o nome do módulo para *Saída* (sem acentos). Observe-se também, que o campo *Record Entity Statistics* está habilitado. Isto faz com que algumas estatísticas básicas sejam realizadas pelo simulador (por exemplo, o número de entidades atendidas pelo servidor durante o período de tempo simulado).

Figura 7 – O módulo *Dispose* editado



No procedimento de modelagem aqui descrito primeiro posicionamento, na área de trabalho, os módulos necessários ao modelo para, subsequentemente, editá-los um a um. Esta não é uma regra rígida de modelagem. Os módulos poderiam ter sido editados na medida em que fossem posicionados.

### 3. Executando a Simulação

Neste ponto, tem-se o sistema modelado, mas ainda sem as especificações para o controle da execução da simulação.

Para este propósito, deve ser selecionado o comando *Run/Setup*, localizado no menu principal do *Arena*. A Figura 8 mostra a janela de diálogo.

São várias as alternativas a serem preenchidas. No painel *Replication Parameters*, deve ser alterado o campo *Replication Length* (de *Infinite* para *8 horas*). Além deste, vê-se que o campo *Base Time Units* foi alterado de horas para minutos. Isto faz com que as estatísticas apresentem seus resultados nesta unidade de tempo. Os demais campos deste painel podem permanecer inalterados. Observa-se que, para o campo *Number of Replications*, foi aceito o valor 1, significando que apenas uma rodada de simulação será executada.

Figura 8 – Janela de diálogo *Run/Setup*, localizada no menu principal do *Arena*.

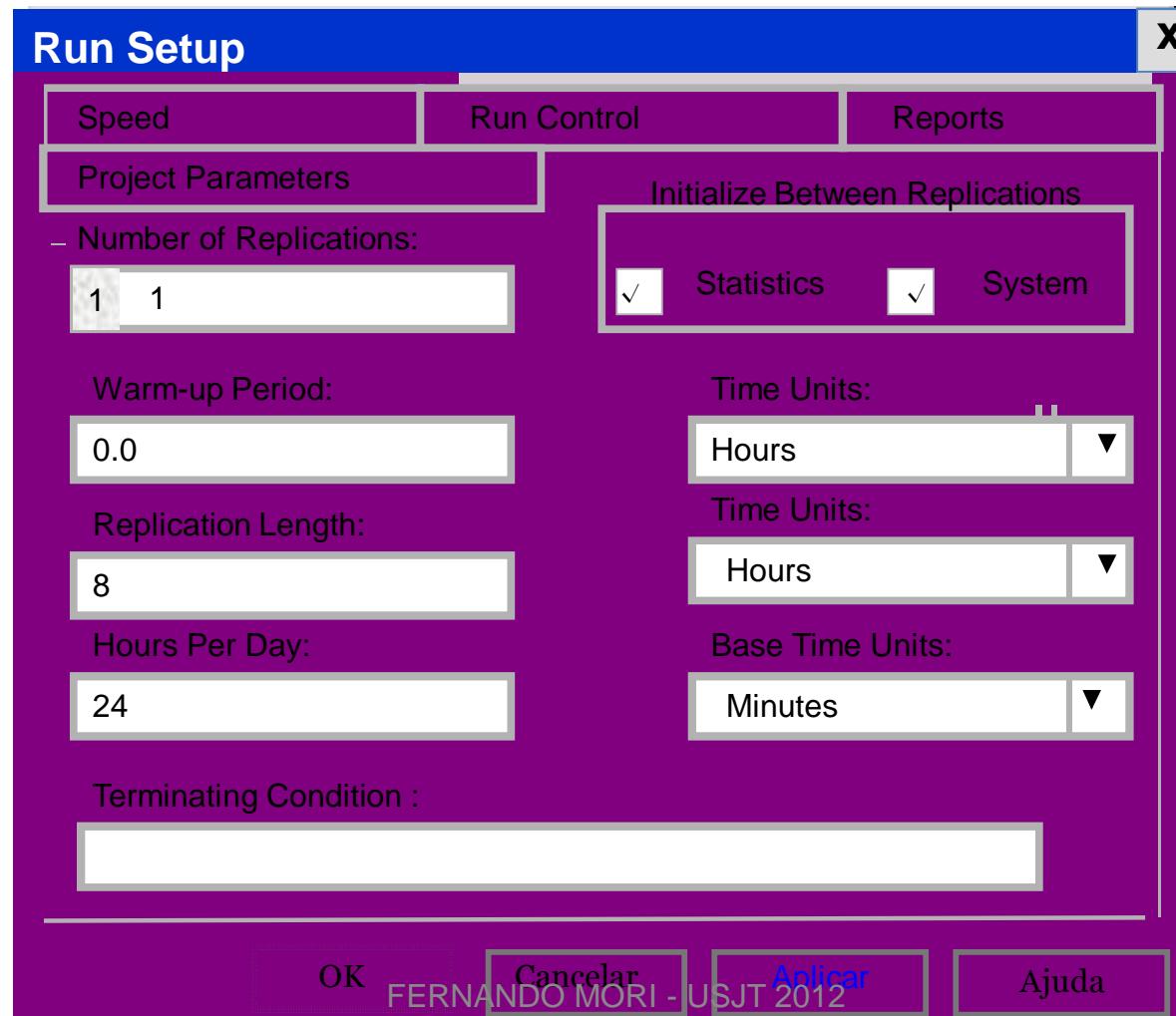
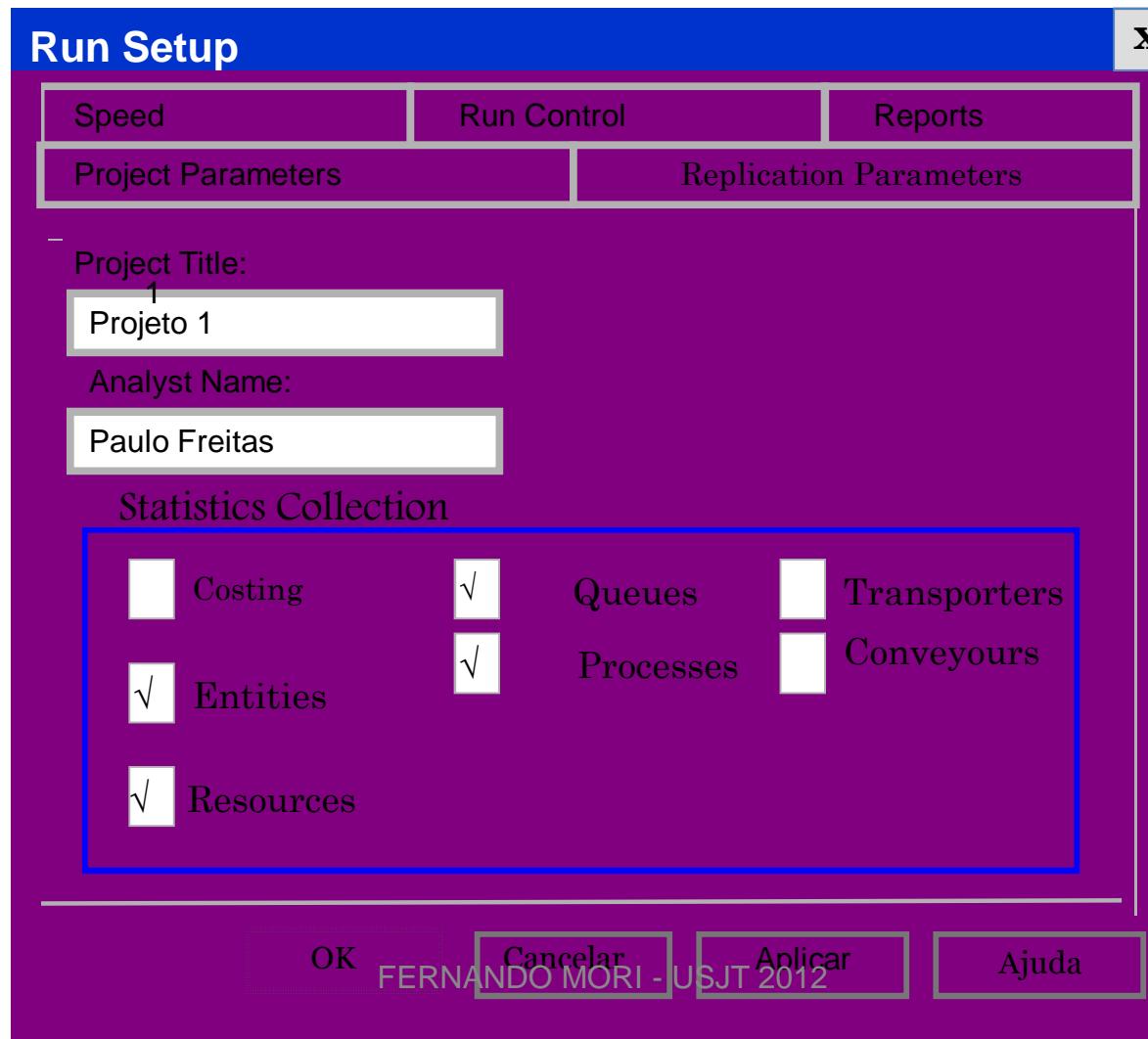
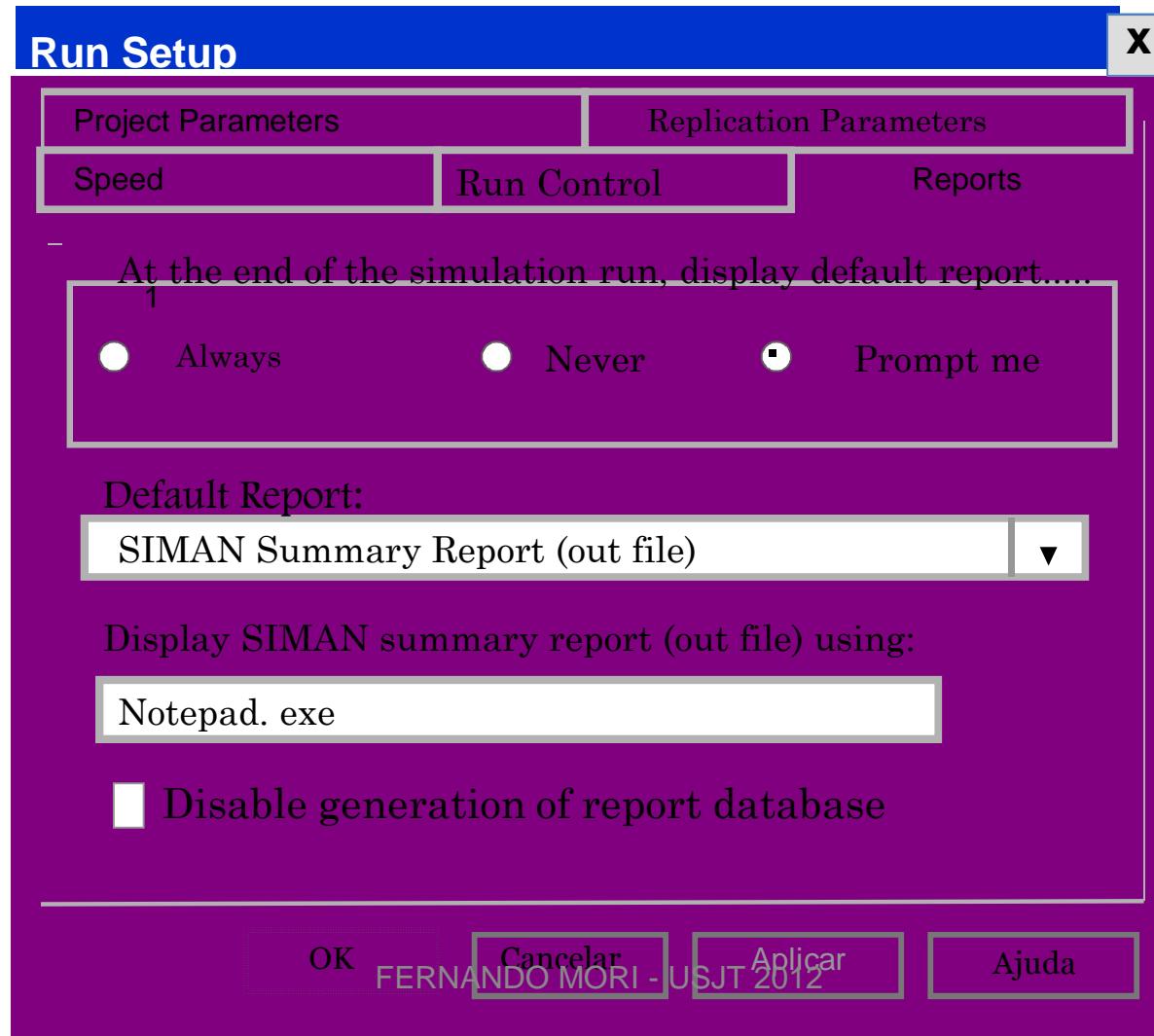


Figura 9 – Janela de diálogo do painel Project Parameters. Vê-se que estatísticas sobre filas (*Queues*), processos (*Processes*), entidades (*Entities*) e recursos (*Resources*) serão realizadas.



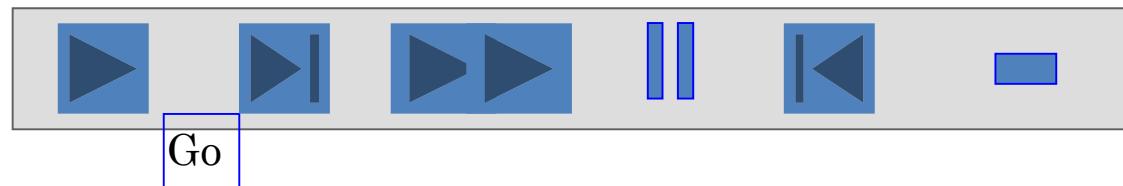
Finalmente, examine-se o painel *Reports*. A *Figura 10* indica que, ao término da simulação, o relatório com os resultados poderá ou não ser apresentado e aberto pelo usuário. O programa questionará o usuário (opção *Prompt me*). Uma vez que o relatório seja aceito, algumas alternativas se apresentam. O padrão é *Category Overview*. Esta forma de apresentação mostra cada uma das estatísticas autorizadas em detalhes. Um opção mais simplificada é solicitar um relatório resumo (padrão até a versão anterior do Arena) ativando a alternativa *SIMAN Summary Report. (out file)*, a qual é vista na *Figura 10*, a seguir.

*Figura 10 – Janela de diálogo do painel Reports.*



Para rodar a simulação, isto é, executar o programa, basta clicar sobre o botão Go (*Executar* – seta apontando para a direita) do painel de controle de execução da simulação da interface básica do *Arena*. Este painel (*Fig. 11*) contém uma série de botões com a aparência de um painel de controle de um videocassete. Os botões representam os seguintes comandos (da esquerda para a direita): *Executar*, *Executar passo-a-passo*, *Execução acelerada*, *Pausa*, *Reiniciar a simulação* e *Encerrar a simulação*.

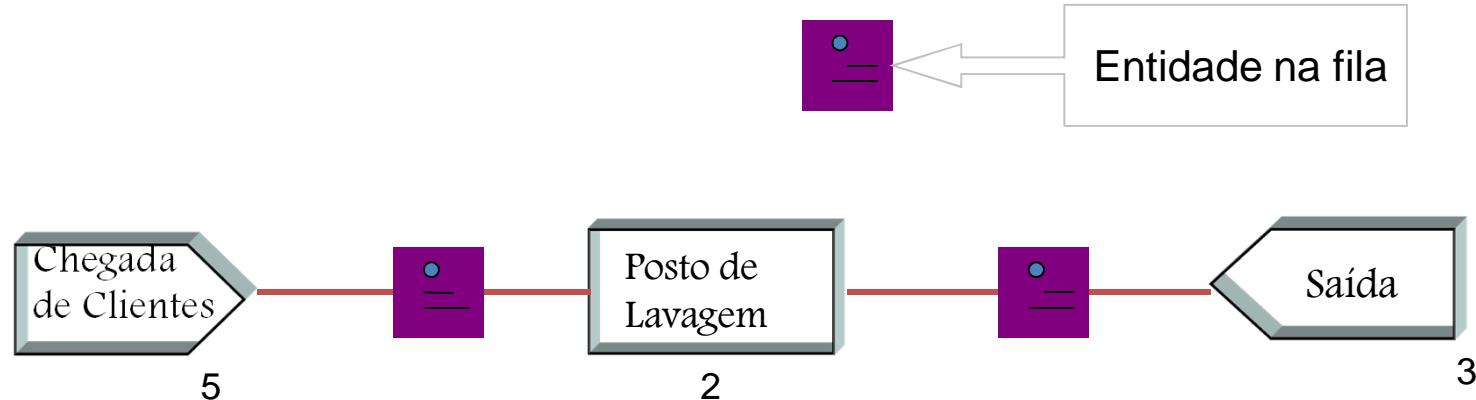
*Figura 11- Painel de controle de execução de simulação*



Após o início da execução da simulação, o Arena passa a verificar o modelo na procura de eventuais erros de sintaxe e/ou lógica. Uma vez que o modelo esteja isento de erros, a simulação tem início. O estado da execução é mostrado na barra de estados (abaixo e à direita da tela principal do ambiente *Arena*).

Durante a execução, as entidades se “movimentam” da estação *Chegada de Clientes* para o *Posto de Lavagem* e deste para a estação *Saída* (*Figura 12*). Estas são representadas por figuras semelhantes a um relatório. Esta é uma figura padrão que pode, naturalmente, ser alterada para melhor representar o tipo de entidade que está circulando pelo sistema.

*Figura 12 – Entidades circulando pelo modelo durante a simulação.*



Quando uma ou mais entidades necessitam esperar na fila do Operador, estas ficam alinhadas sobre o local onde se encontra sua fila, uma linha azul que também fica invisível durante a simulação. Na figura acima, vê-se que 5 entidades deram entrada no modelo, 2 estão no Posto de Lavagem (uma na fila) e 3 já deixaram o sistema.

A simulação pode ser interrompida (pausa) a qualquer momento. Uma pausa não implica em finalizar a simulação. A sessão de simulação ainda permanecerá ativa. A pausa pode ser obtida com um clique sobre o botão de pausa ou pressionando-se a tecla *Esc*.

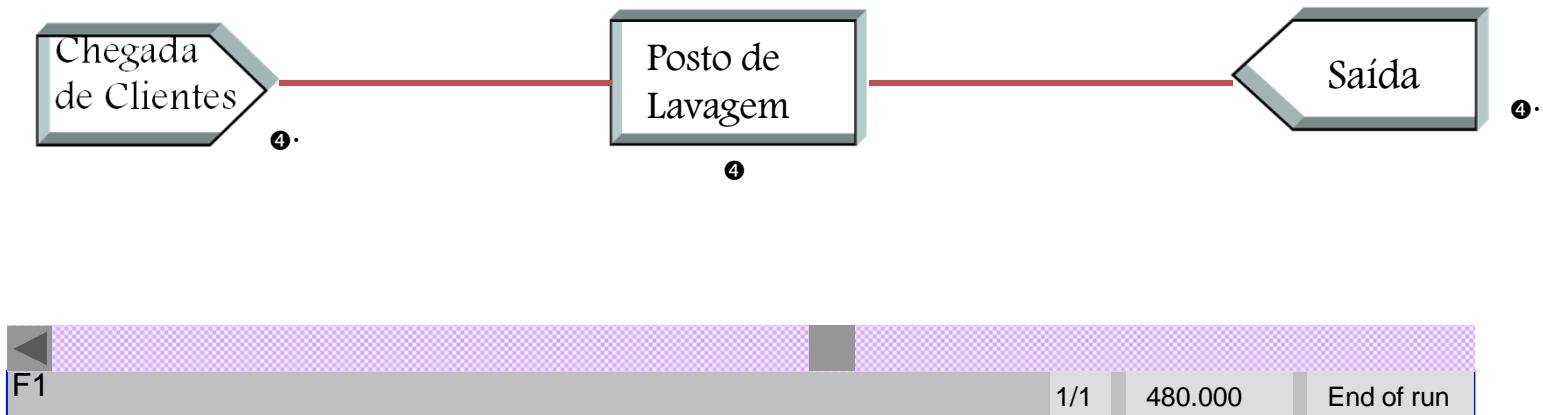


Figura 13 – Encerramento de uma rodada de simulação

Ao final da simulação, o valor do contador, próximo ao módulo *Saída*, mostrará o número de entidades que foram servidas durante as 480 unidades de tempo (48 horas). Durante a simulação, este valor vai se alterando na medida em que as entidades vão sendo processadas e deixam o sistema.

Quando o tempo final de simulação é atingido, a simulação se encerra e uma mensagem aparece na tela perguntando se o usuário deseja ver os resultados. Se a resposta for afirmativa, abre-se um arquivo do aplicativo *Bloco de Notas do Windows* (*Notepad.exe*), com várias estatísticas que o programa acumulou durante a simulação. Os resultados são mostrados como um arquivo texto, podendo ser facilmente editados para servir como parte da documentação do sistema modelo.

## Exercícios

- 1) Navios chegam a um porto a intervalos de EXPO(8) horas e gastam TRIA(3,5,10) horas para descarregar. Faça o diagrama de blocos e submeta-o ao Arena. Simule 8.760horas (1ano). Determine os valores para:
- Taxa de ocupação do porto;
  - Tamanho médio da fila;
  - Tempo médio na fila.
- 2) Em uma fábrica de geladeiras, na seção de colocação de motores, a chegada de uma geladeira sem motor ocorre a intervalos de EXPO(50)minutos e gastam-se TRIA(25,35,50) minutos para o serviço. Determine o tamanho médio da fila. Faça o diagrama de blocos e submeta-o ao Arena. Simule 480 minutos (8 horas).

## ■ RELATÓRIOS DO ARENA

Os relatórios do *arena* são os seguintes:

- “ Categoria Overview: Visão Global da categoria.
- “ Categoria Replication: Semelhante ao anterior, mas dividido pelas aplicações.
- “ Entities: Entidades
- “ Frequencies: Frequências
- “ Process: Processos
- “ Queues: Filas
- “ Resources: Recursos
- “ Transfers: Transportadores
- “ User Specified: Variáveis ou atributos especificados pelo usuário.
- “ Agents: Agentes
- “ Call times: Chamadas e contadores

Alguns relatórios importantes:

- a) Relatório sobre filas (Queues)

Campo	Significado
Waiting Time	Tempo médio na fila (TF)
Number Waiting	Tamanho médio da fila (NF)
Waiting Time Average	Tempo médio na fila (TF)
Waiting Time Halfwidth	Tempo na fila - meia largura
Waiting Time Minimum	Tempo na fila – valor mínimo
Waiting Time Maximum	Tempo na fila – valor máximo
Number Waiting Average	Tamanho médio da fila
Number Waiting Halfwidth	Tamanho da fila - meia largura
Number Waiting Minimum	Tamanho da fila - mínimo
Number Waiting Maximum	Tamanho da fila - máximo

## b) Relatórios sobre Recursos (Resources)

Campo	Significado
Number Busy	Quantidade em uso
Number Scheduled	Disponibilidade
Utilization	Taxa de utilização
Number times used	Número de vezes que o recurso foi utilizado

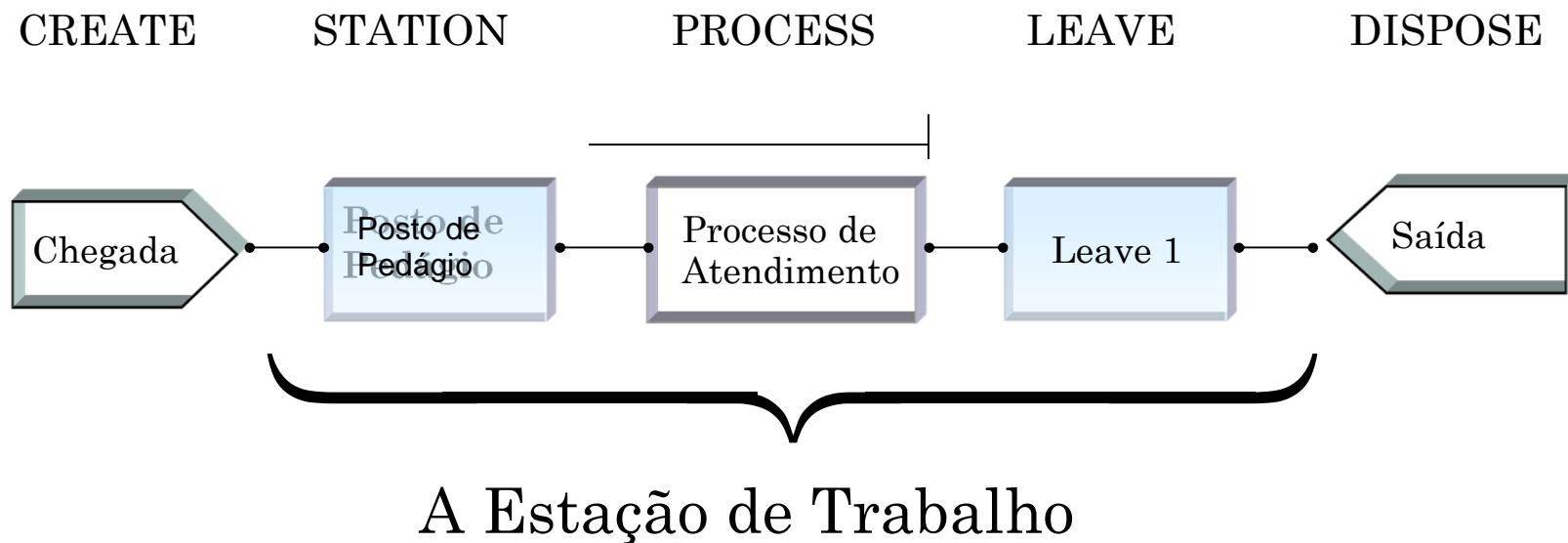
# AULA 2

- ” Conceito de Estação de Trabalho
- ” Uso do Bloco Decide
- ” Uso do Bloco Leave
- ” Uso do Bloco Station

## ■ ESTAÇÃO DE TRABALHO

A estação de trabalho no Arena é composta de 3 módulos.

Considerando o modelo do posto de lavagem teríamos: Fig. 14



O bloco ***Station*** informa que o veículo ocupou a estação.  
O bloco ***Leave*** para informar que o veículo desocupou a estação.

Para acessar estes dois blocos precisamos do template ***Advanced Transfer***.

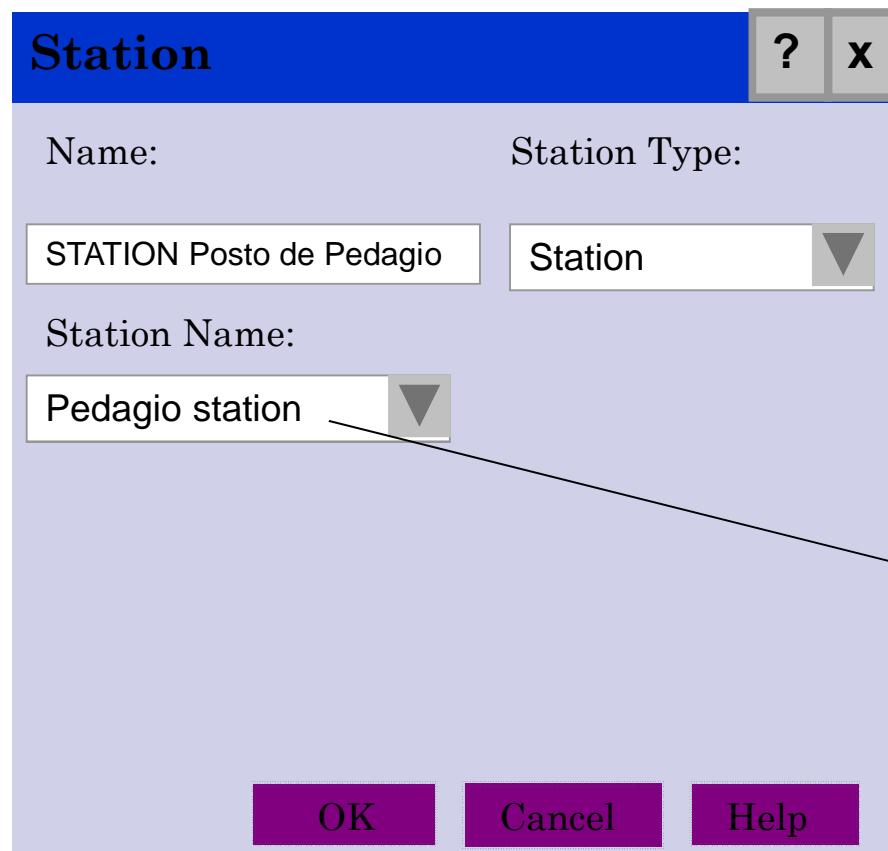
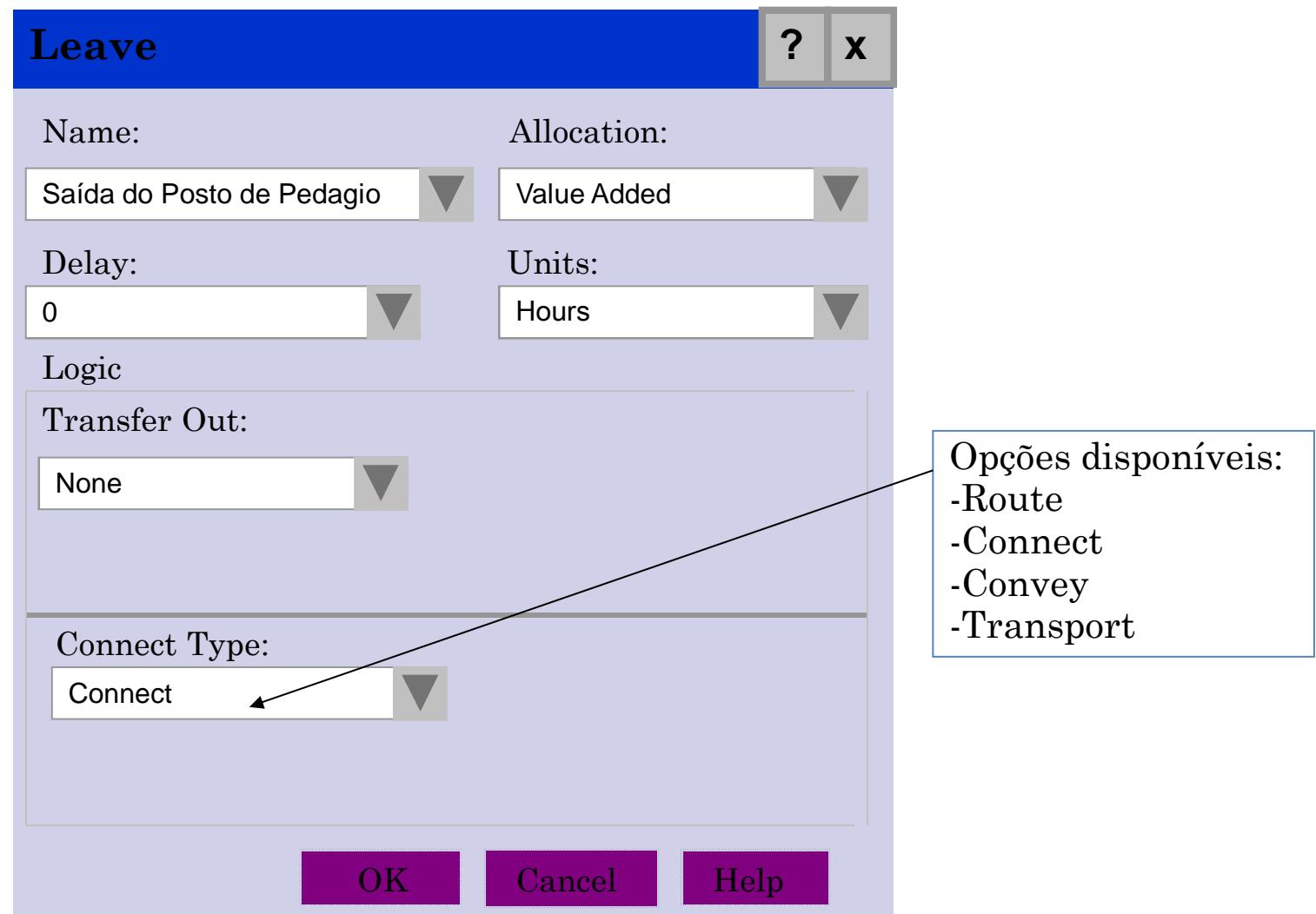


Figura 15

# O MÓDULO ENTER OU LEAVE

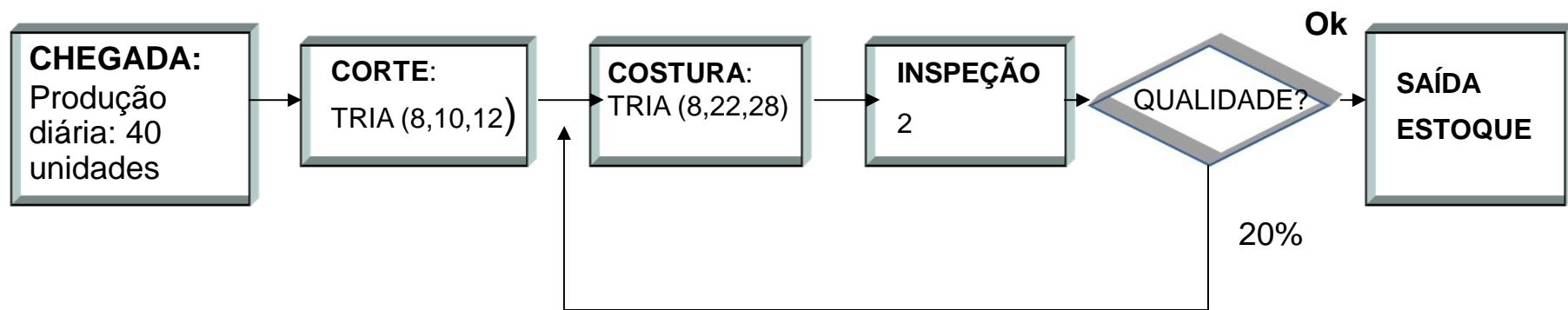
Figura 16



## ■ MÓDULO DECIDE

- Suponha que uma fábrica de roupas deseja analisar seu processo de produção. Os dados são:
  - Produção diária: 40 unidades
  - Tempos de produção:
    - 1) Corte: TRIA (8, 10, 12) minutos
    - 2) Costura: TRIA (18, 22, 28) minutos
    - 3) Inspeção: igual a 2 minutos
  - Índice de rejeição na inspeção: 20%
  - Tempos de deslocamento:
    - 1) entre corte e costura: 2 minutos
    - 2) entre inspeção e costura: 2 minutos
    - 3) entre costura e inspeção: 2 minutos
    - 4) entre inspeção e estoque: 2 minutos

*Figura 16 – Fluxo na fábrica de roupas*



Desejamos saber:

- O dimensionamento adequado (capacidade de atendimento em cada estação de trabalho).
- O tempo médio de confecção de uma peça de roupa.
- A produção em 600 minutos (10 horas)

O processo de inspeção é executado com o auxílio do módulo DECIDE que é obtido no template ***Advanced Transfer***.

Neste processo devemos criar 4 estações por meio do módulo ***Station***:

Estação	Nome da Estação	Nome do Processo	Nome da Fila
Corte	Corte. Station	Processo Corte	Processo Corte.Queue
Costura	Costura. Station	Processo Costura	Processo Costura.Queue
Inspeção	Inspecao. Station	Processo Inspecao	Processo Inspecao.Queue
Estoque	Estoque. Station		

## A produção diária: ***Módulo Create***

- First Creation = 0.0
- Entities per Arrival = 40 : são criadas 40 entidades
- Max Arrival = 1: não serão criados mais entidades  
Isto quer dizer que serão criadas 40 transações no momento zero.

*Fig.17*

## Deslocamento entre estações: ***Módulo Leave***

O deslocamento entre estações é fornecido no módulo Leave da seguinte forma:

- Connect Type = Route → este tipo de conexão envia a entidade para a estação “Corte” gastando 2 minutos. Ao ativar esta opção, o desenho do módulo não possui mais sinal de conexão. *Fig.18*

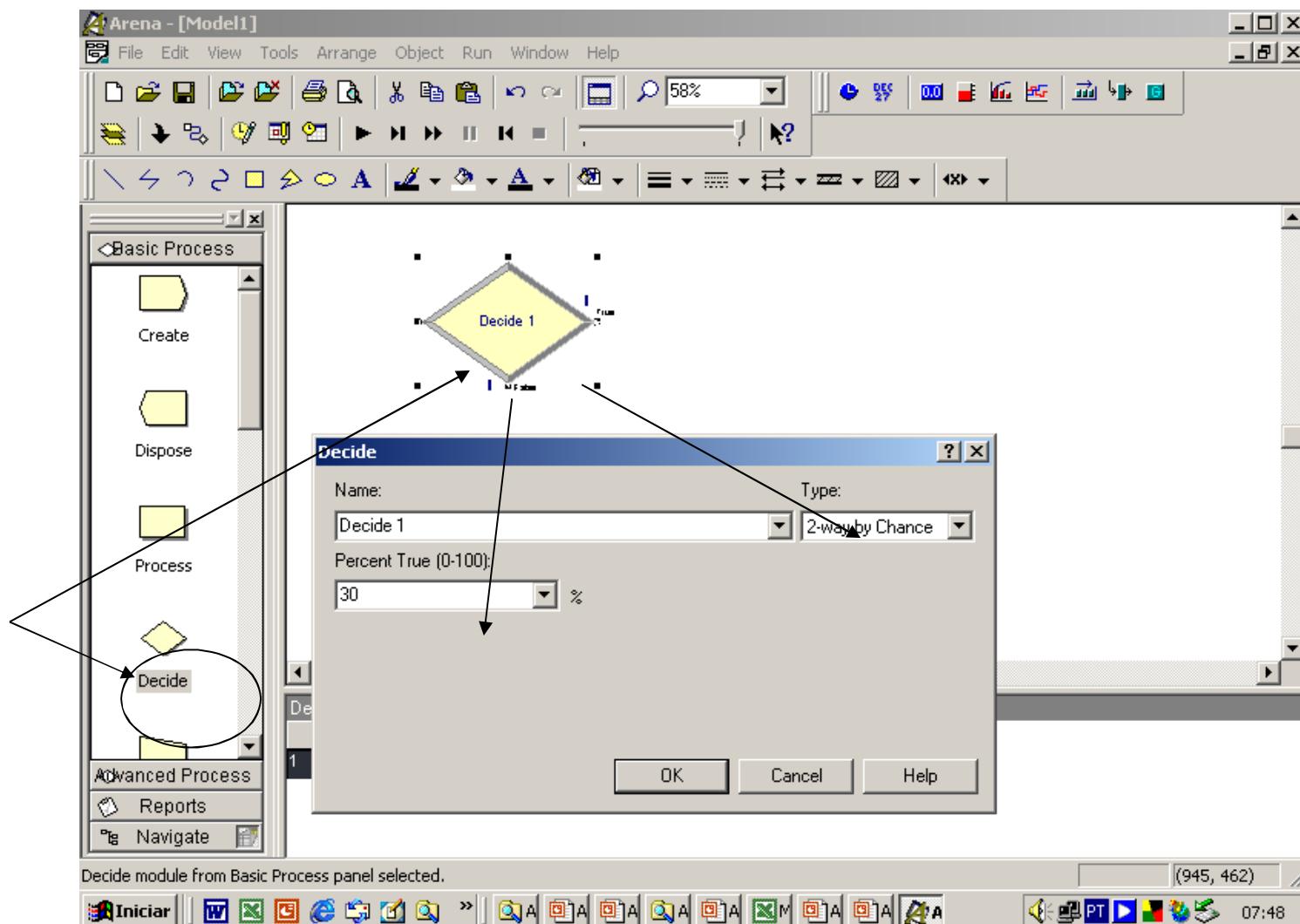
## ■ MÓDULO DECIDE

Este módulo é usado quando em um ponto de fluxo temos diversas opções de continuação. Na estação inspeção temos duas opções de fluxo:

- 20% das peças apresentam defeito e devem ser reenviadas para a correção na estação costura.
- 80% passam no teste de qualidade.

Neste módulo temos:

- Type = 2 Way Chance (2 opções de probabilidade)
- Percent true (0 – 100)= 80: ou seja 80% dos testes devem ser verdadeiros.



# 1º Exercício

1) Uma fábrica de roupas deseja analisar seu processo de produção. Os dados são os seguintes:

Produção diária desejada: 40 unidades

Tempos de produção:

Corte: Tria (8, 10, 12) minutos.

Costura: Tria (18, 22, 28) minutos.

Inspeção igual a 2 minutos.

Índice de rejeição na inspeção de qualidade: 20%

Tempos de deslocamento:

Entre Corte e Costura: 2 minutos.

Entre Inspeção e Costura: 2 minutos.

Entre Costura e Inspeção: 2 minutos.

Entre Inspeção e Estoque: 2 minutos.

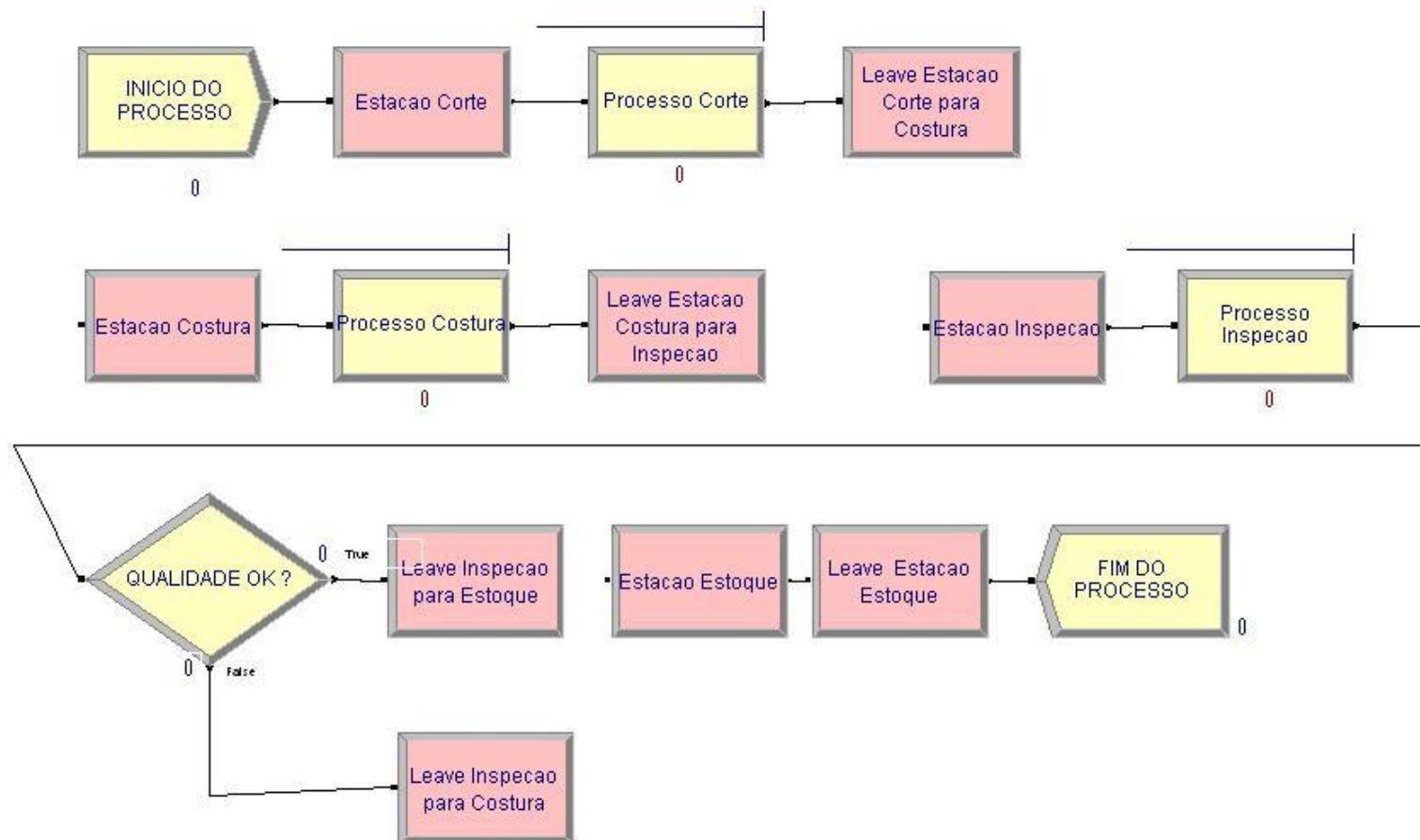
Desejamos saber:

**Qual a capacidade de atendimento em cada estação de trabalho?**

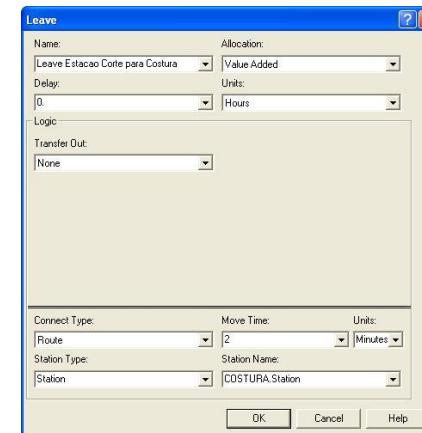
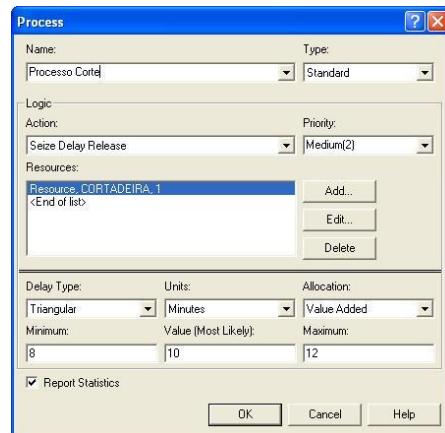
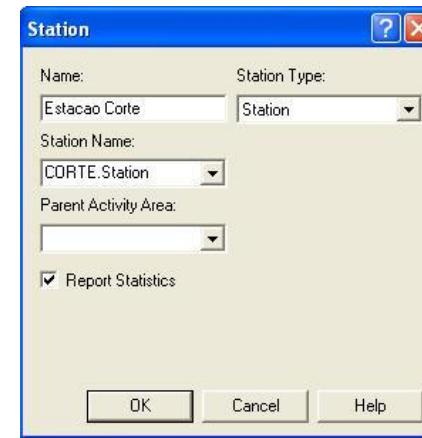
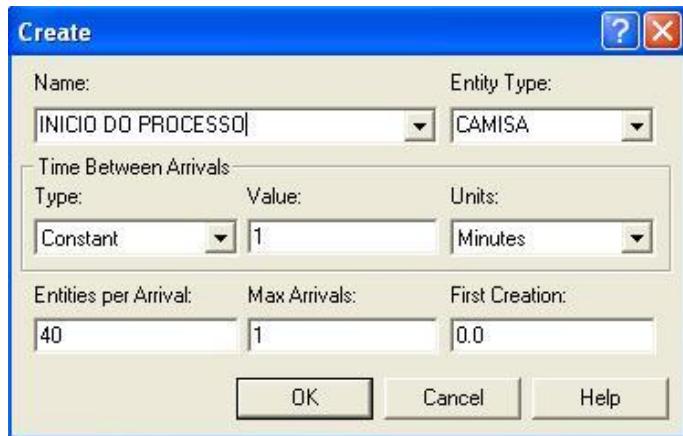
**O tempo médio de confecção de uma peça de roupa.**

**A produção em 600 minutos (10 horas).**

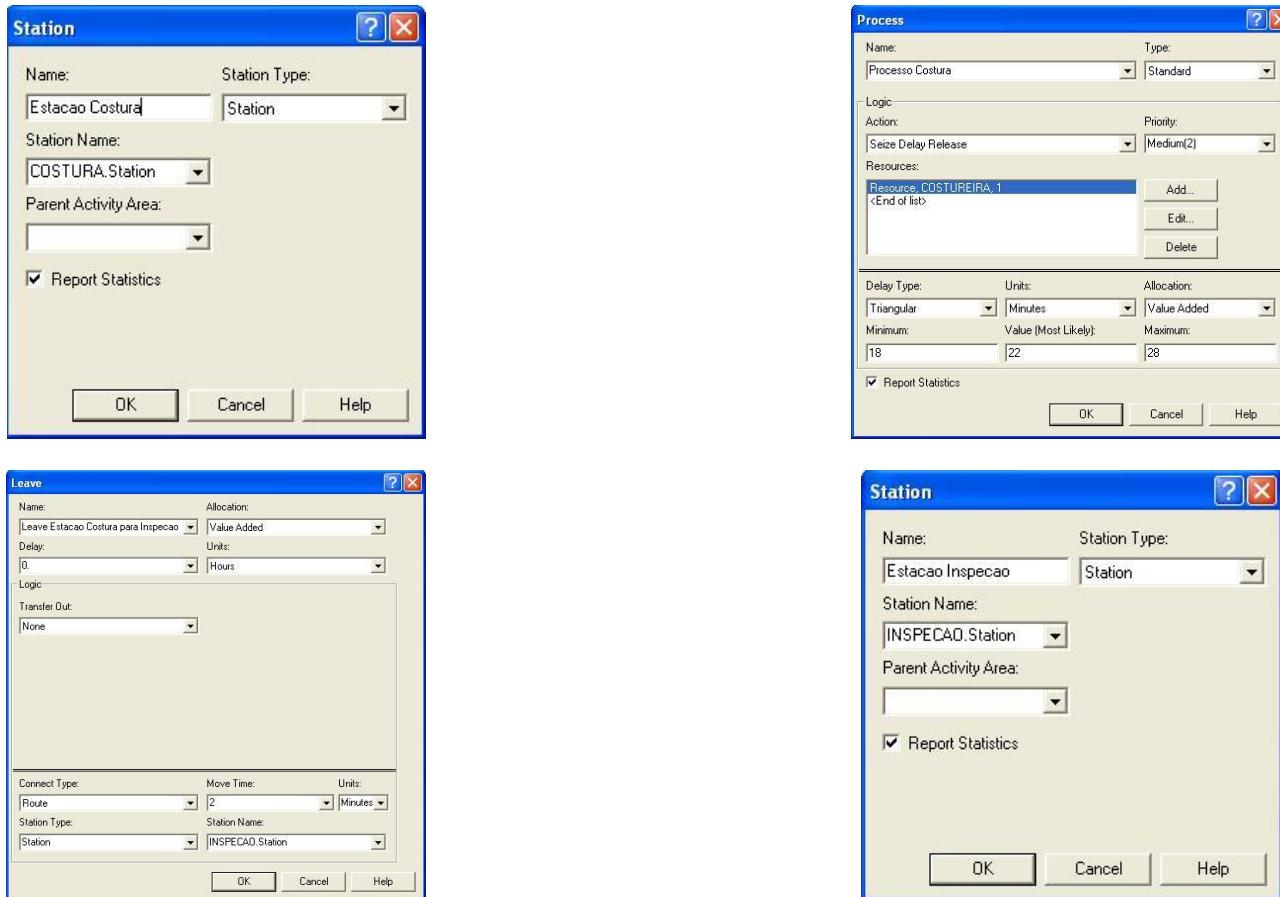
# 1º Exercício – Fluxograma



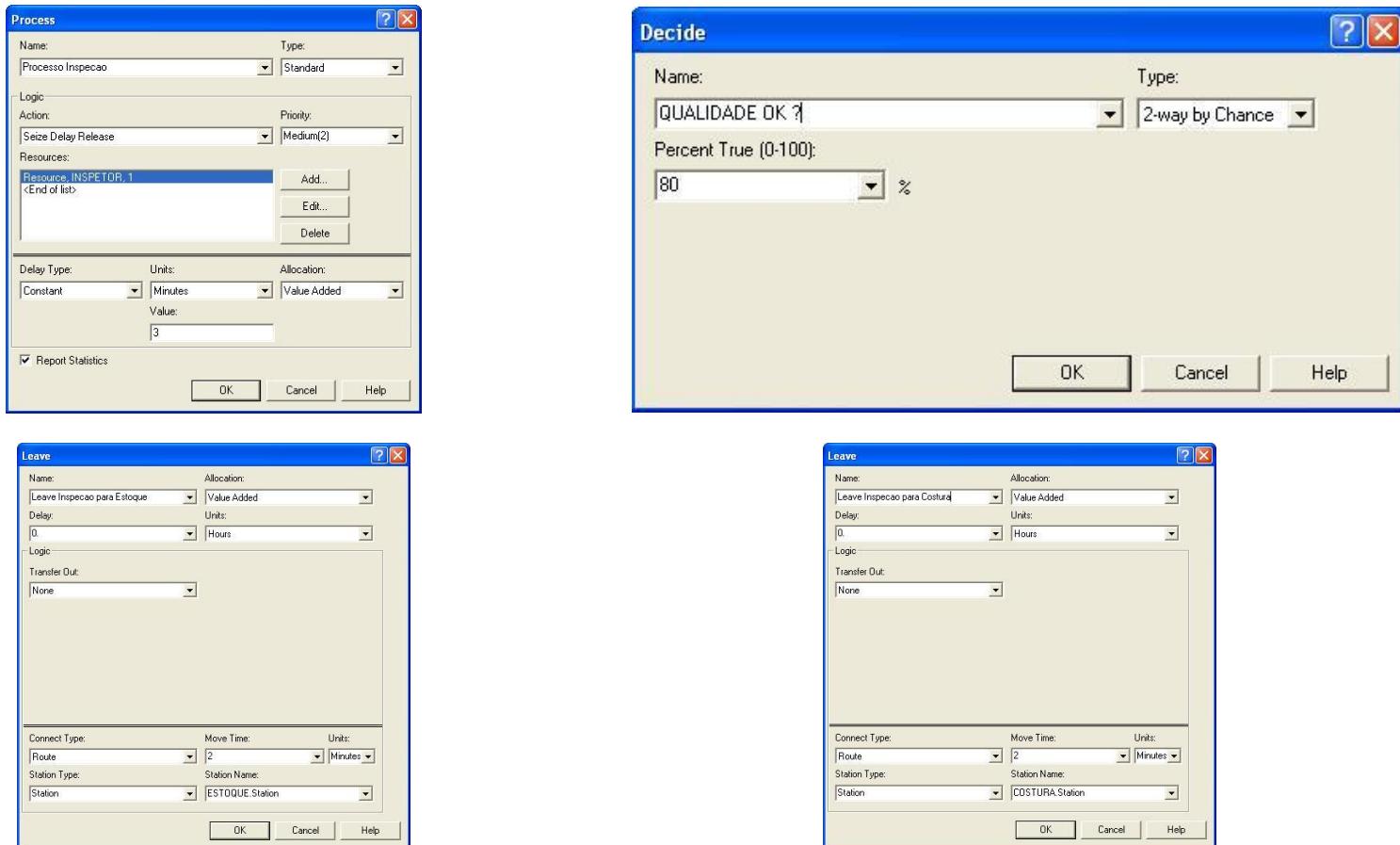
# 1º Exercício – Módulos



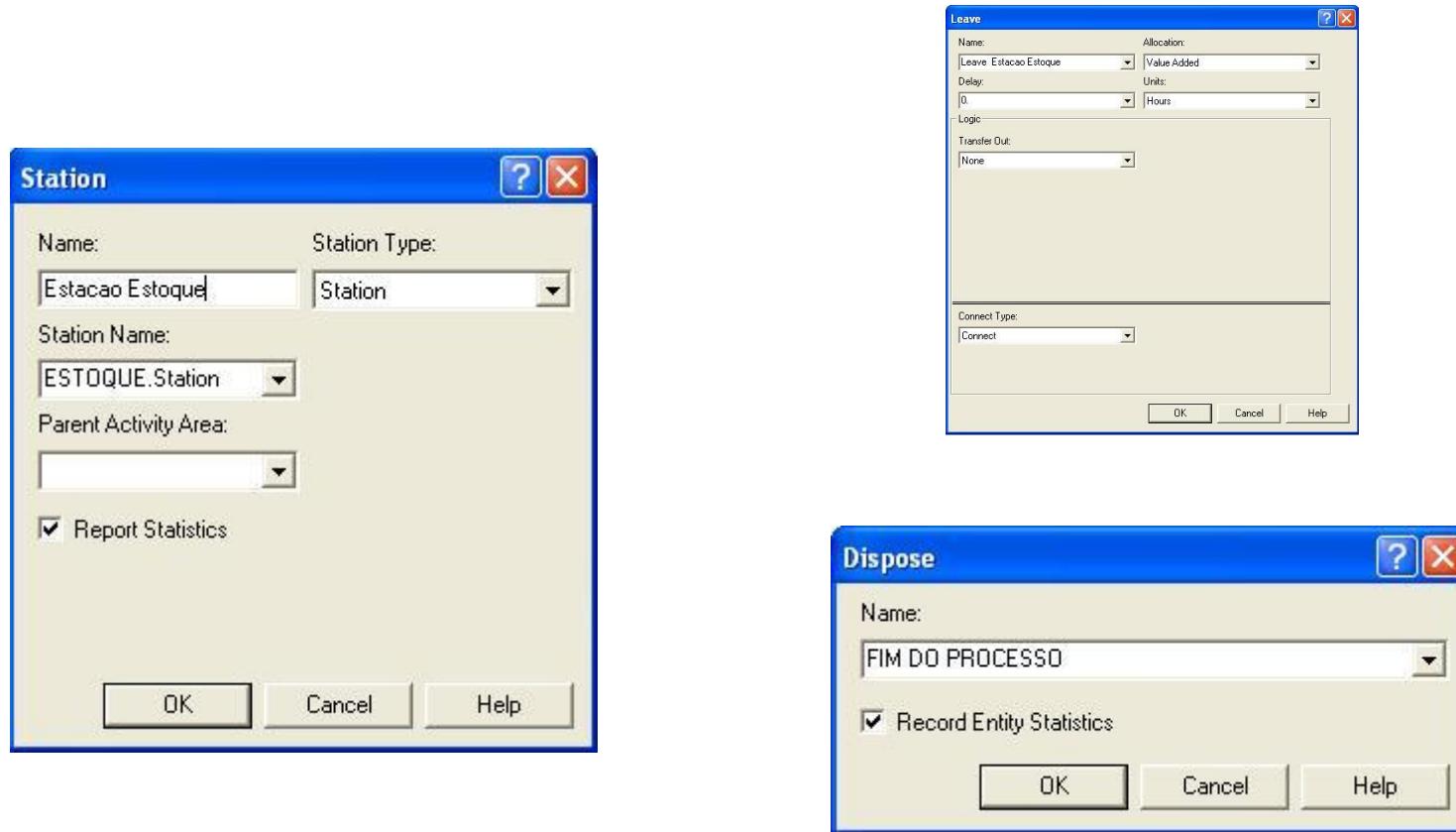
# 1º Exercício – Módulos



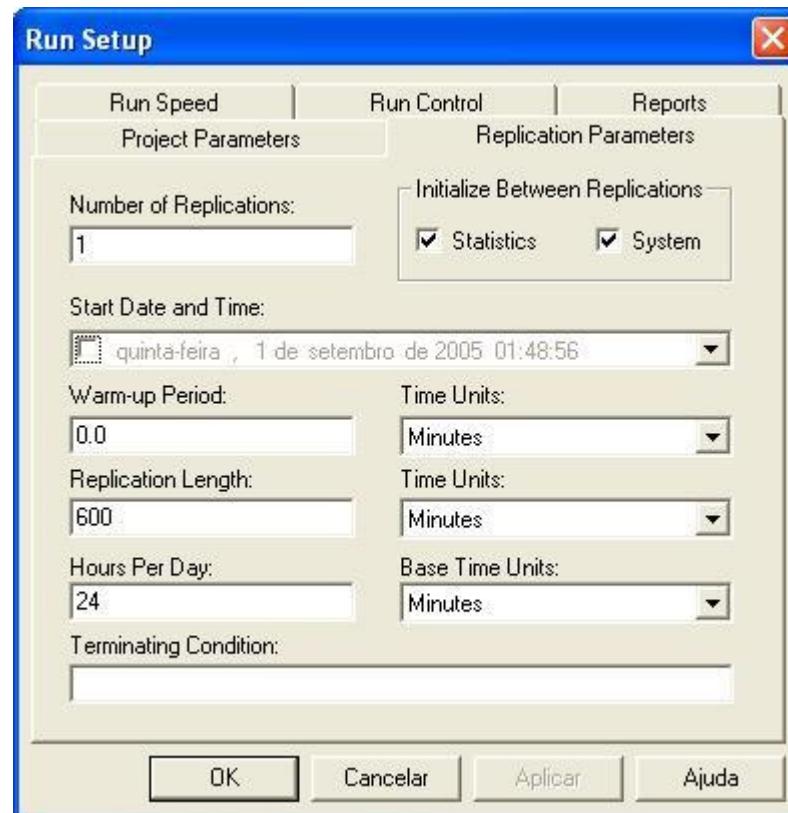
# 1º Exercício – Módulos



# 1º Exercício – Módulos



# 1º Exercício – Setup



# 1º Exercício – Conclusão

- ” Reports: Category Overview, campo System Number Out
- ” Total de peças produzidas: 19
- ” Tempo Médio de produção de uma peça: 296,10 (ver Reports Entities, campo Total Time).
- ” **NÃO ATENDEU AO OBJETIVO.**
- ” A fila da estação Costura é muito alta (Report Queue, Segunda Página, campos Average Number Waiting e Maximum Number Waiting):
  - . Fila Média: 15,06
  - . Fila Máxima: 25
- ” Atenção: existe fila na estação Corte devido ao início se dar no instante Zero, as peças são colocadas em fila.
- ” **Realizando simulações, com 2, 3 e 4 costureiras (a alteração deve ser feita na Área de Planilhas), é verificado que a melhor opção para se alcançar a produção de 40 peças é ter 3 costureiras.**

Servidor	NF	TF
Corte	13,10	196,47
Costura	0,39	4,52
Inspeção	0,00	0,04

- ” Motivo das diferenças: tamanho da amostra. Para se ter maior segurança nos cálculos deveria ser utilizada a replicação de diversos dias (comparando-se com 2 e com 3 costureiras).

# 2º Exercício

2) Em uma fábrica chegam pedidos a cada EXPO(23) minutos e vão para a produção (MAQ.A), gastando UNIF(15,25) minutos. A seguir passam por um processo de inspeção, em que:

- . Duração: TRIA(15, 17, 20) minutos
- . Percentual de falha: 30%

As peças defeituosas vão para uma estação de reparo.

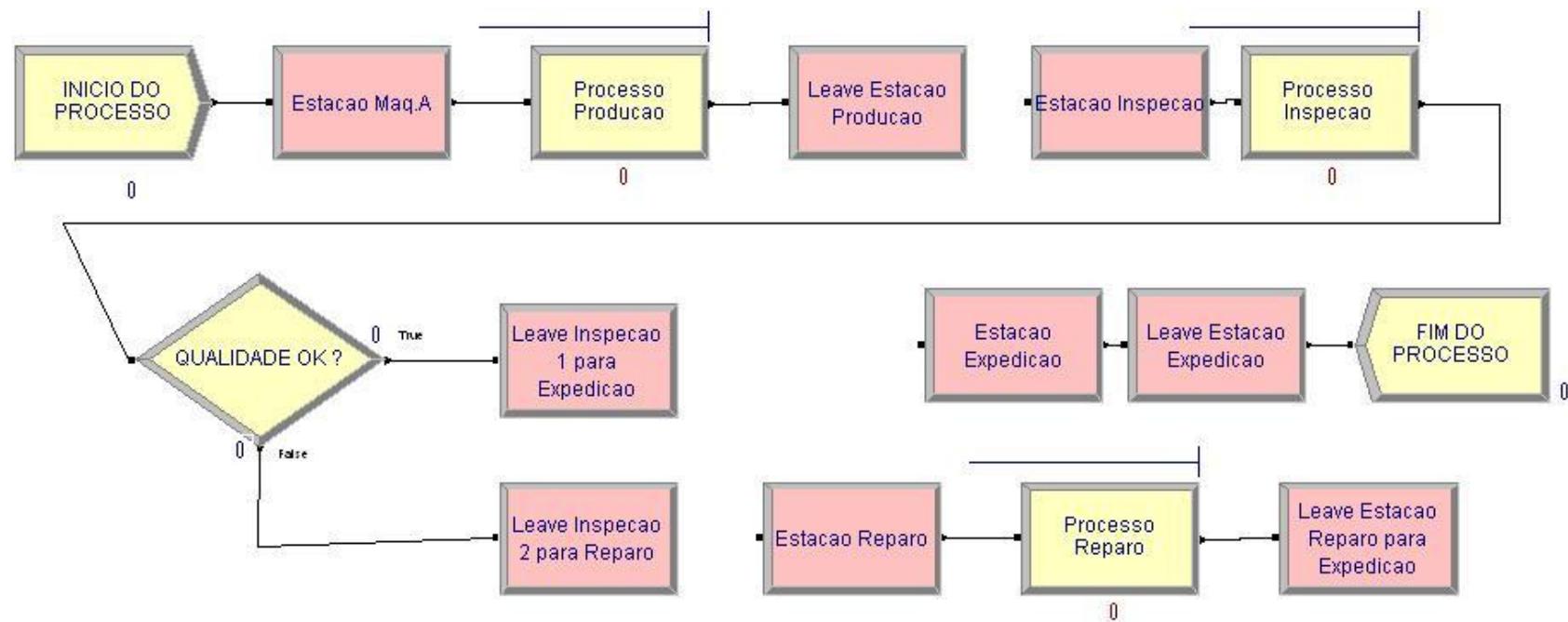
As peças sem defeito vão para a expedição.

Na estação de reparo gasta-se UNIF(10, 15) e, a seguir, as peças vão para a expedição. Todos os tempos de deslocamento são de 2 minutos. Simule 1.000 minutos.

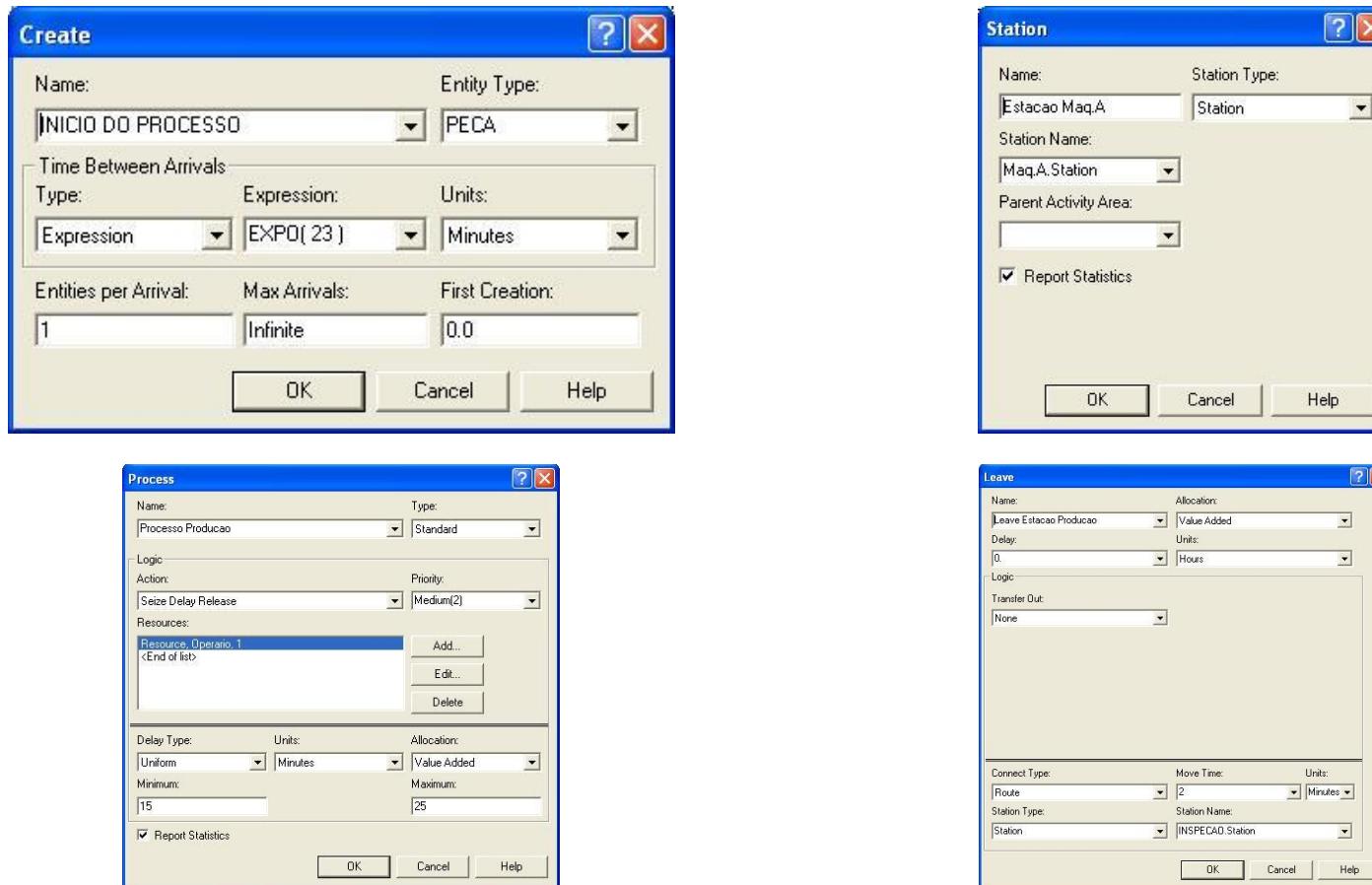
Pede-se:

- . Qual a taxa de ocupação de cada servidor?
- . Qual o tamanho médio de cada fila?
- . Qual o tempo médio em cada fila?
- . Qual o tempo médio para se produzir uma peça?

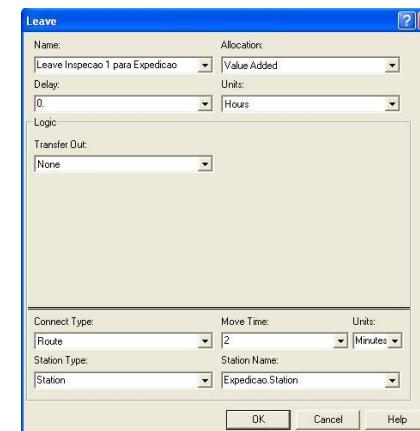
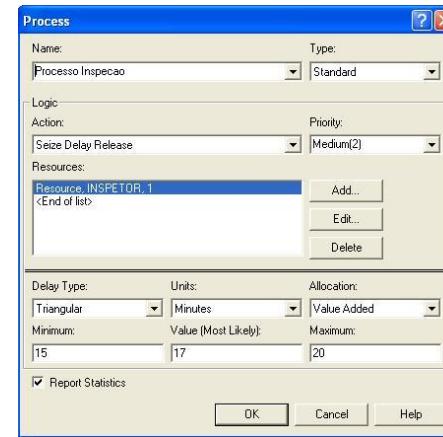
# 2º Exercício – Fluxograma



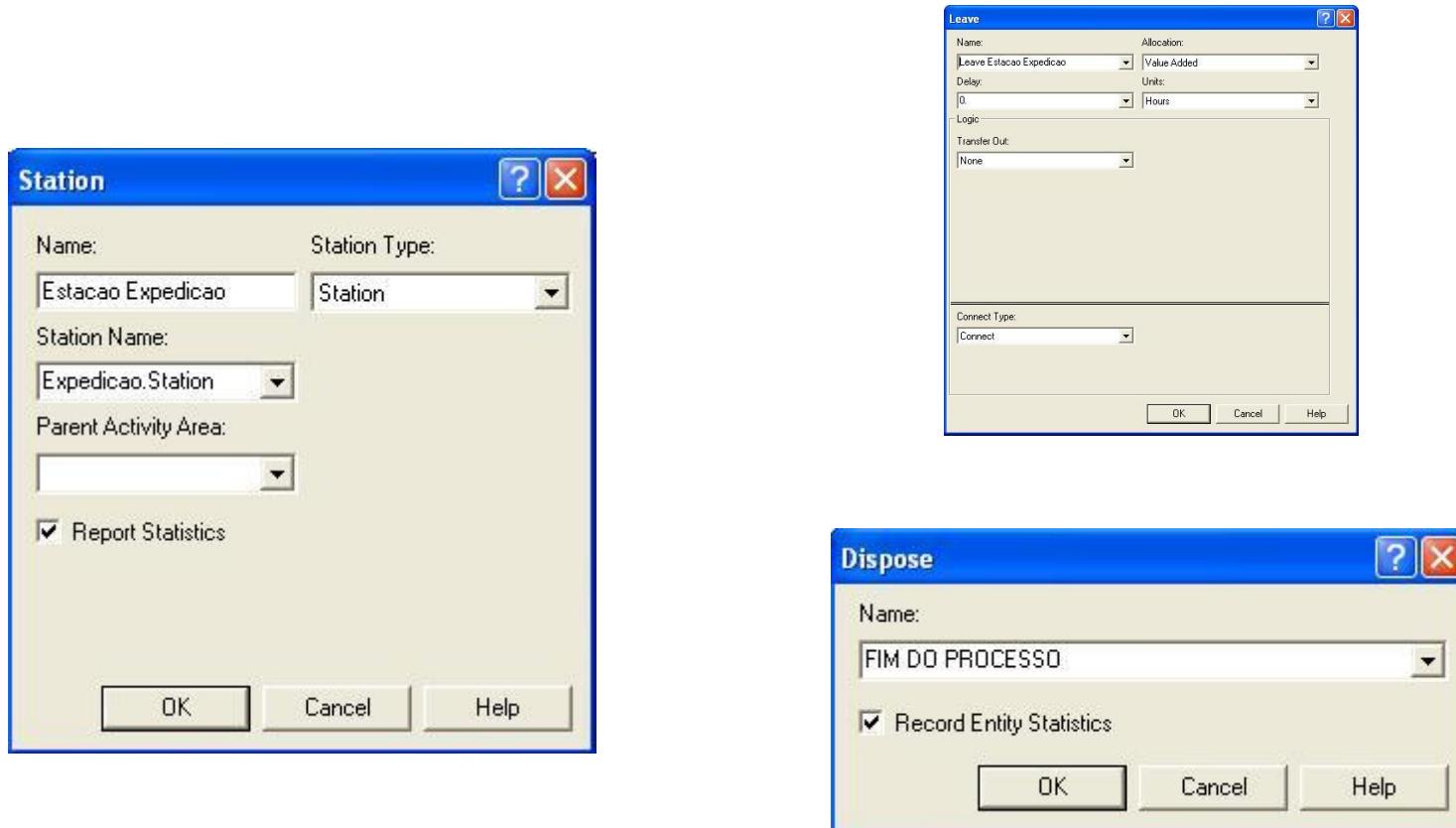
# 2º Exercício – Módulos



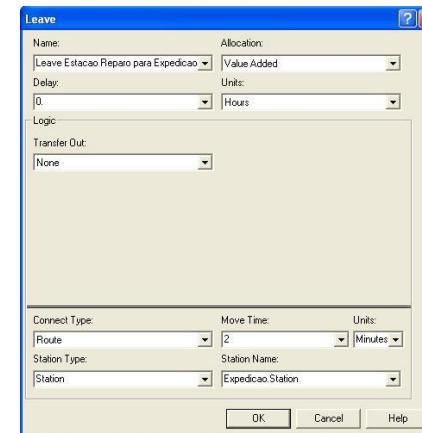
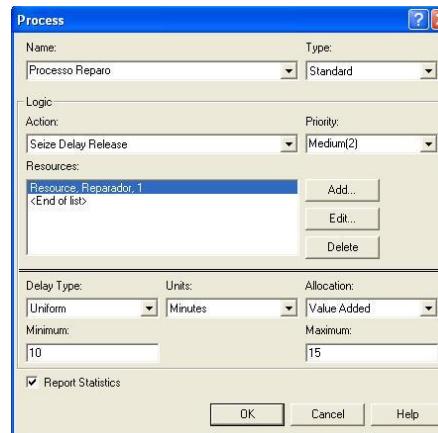
# 2º Exercício – Módulos



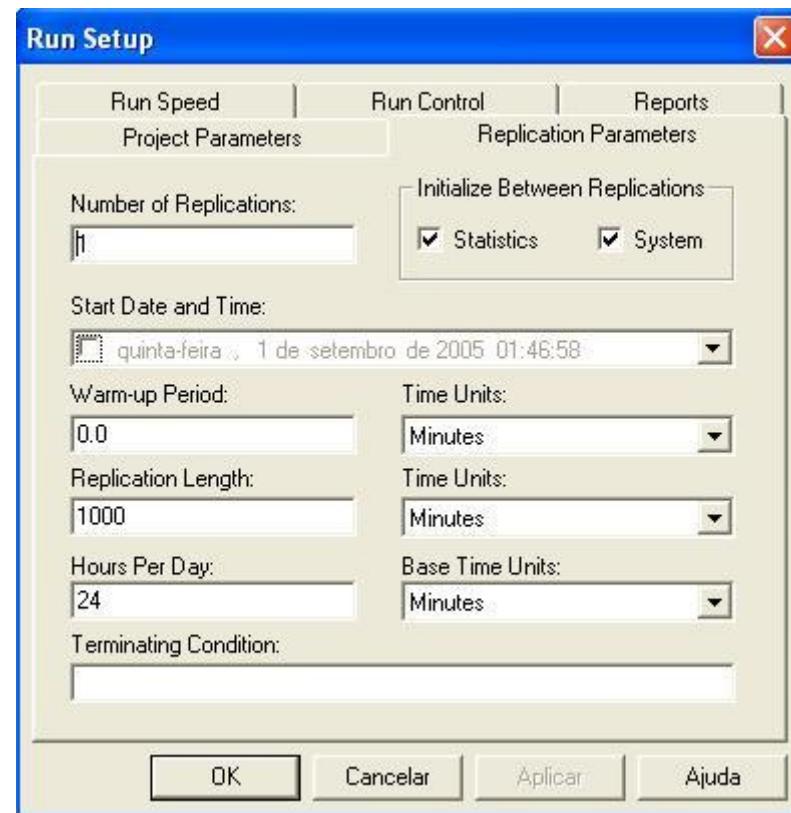
# 2º Exercício – Módulos



# 2º Exercício – Módulos



# 2º Exercício – Setup



# 2º Exercício – Conclusão

- Reports: Category Overview

Servidor	NF (Queues)	Taxa de Ocupação (Relatório Resources)	TF (Queues)
Produção	2,45	0,91	49,47
Inspeção	0,03	0,77	0,67
Reparo	0,00	0,26	0,00

- O tempo para produzir uma peça é de 98,29 minutos (Relatório Entities, coluna Total Time).

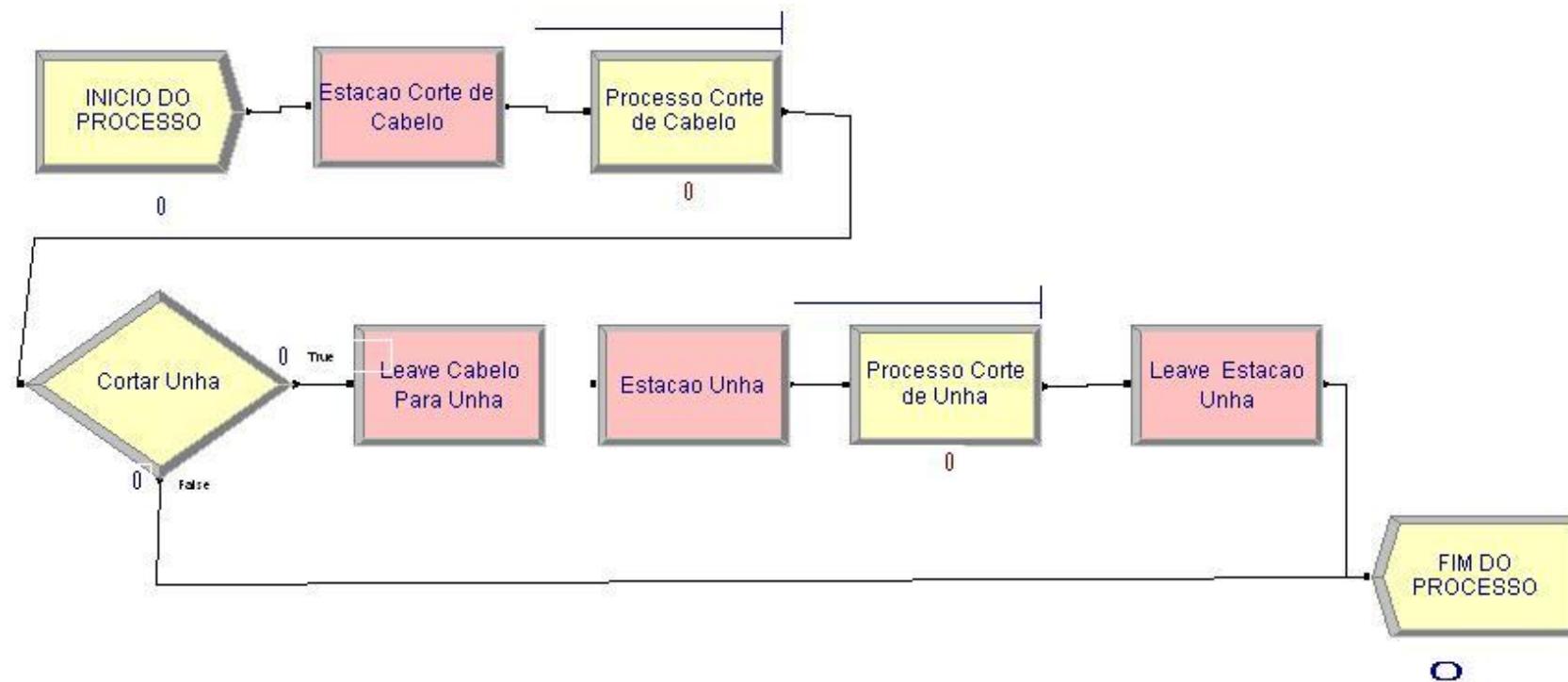
# AULA 3

” Exercícios de Aplicação

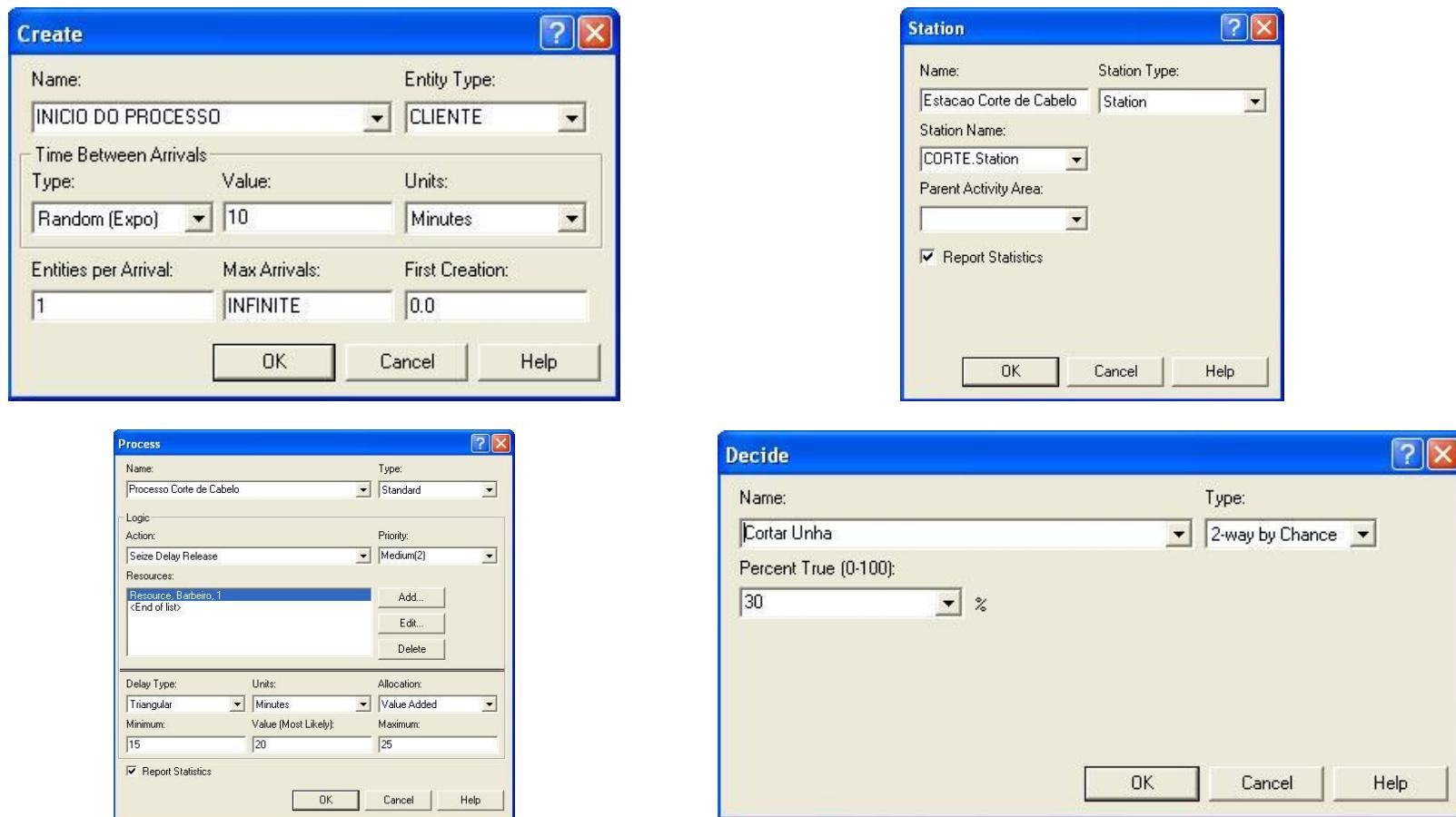
# 3º Exercício

- 3) Em uma barbearia, clientes chegam a cada EXPO(10) minutos. Existem 3 barbeiros e o tempo de corte é de TRIA (15, 20, 25) minutos. Após cortado o cabelo, 30% dos clientes também fazem a unha com uma outra profissional, gastando TRIA (10, 15, 20) minutos. Verifique o tamanho das filas, o tempo nas filas e o tempo que um cliente gasta dentro da barbearia. O tempo de deslocamento entre uma seção de corte de cabelo e a seção de corte de unha é de 1 minuto (o cliente gasta algum tempo conversando, etc.). Conte o total de clientes que foram atendidos no período de 8 horas.

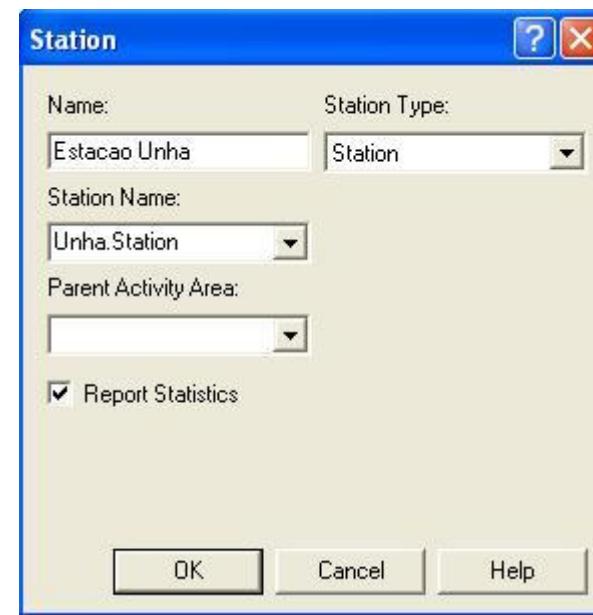
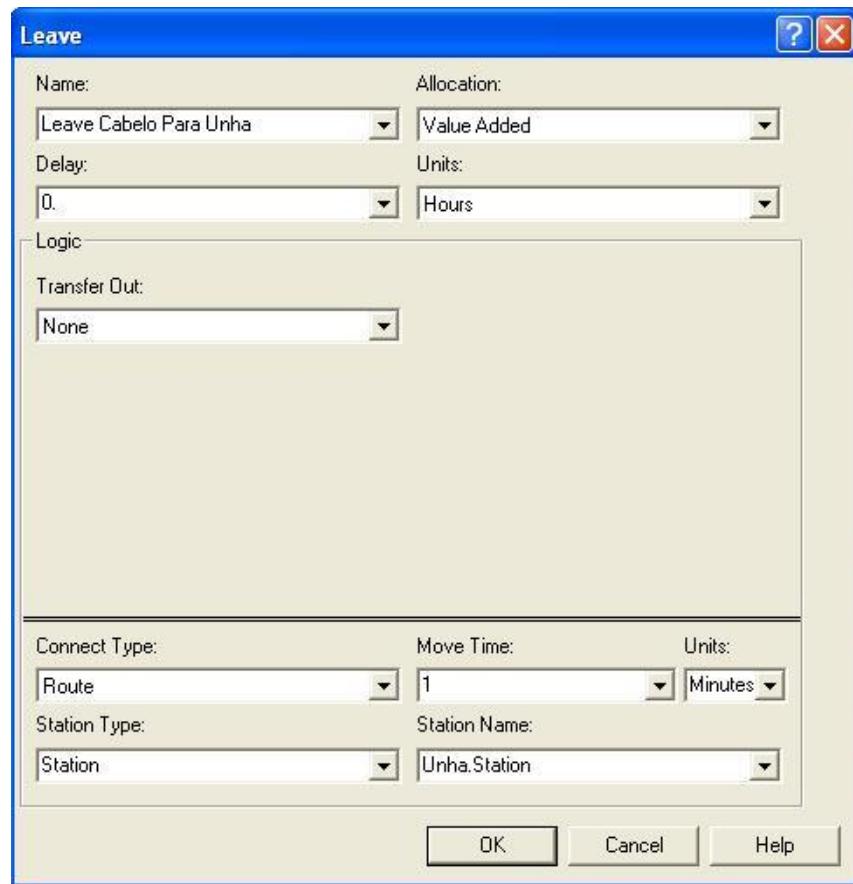
# 3º Exercício – Fluxograma



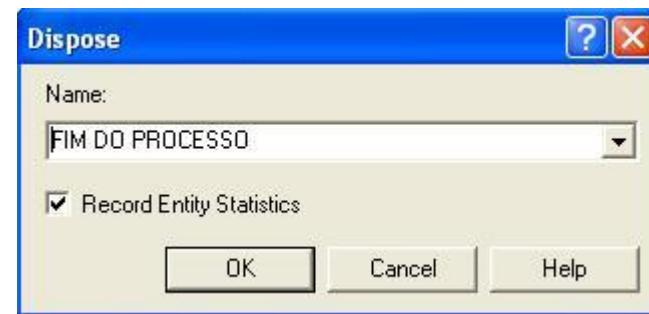
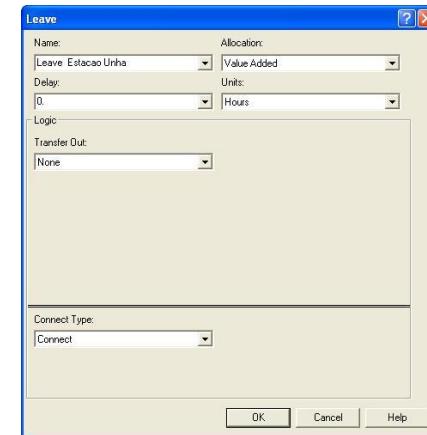
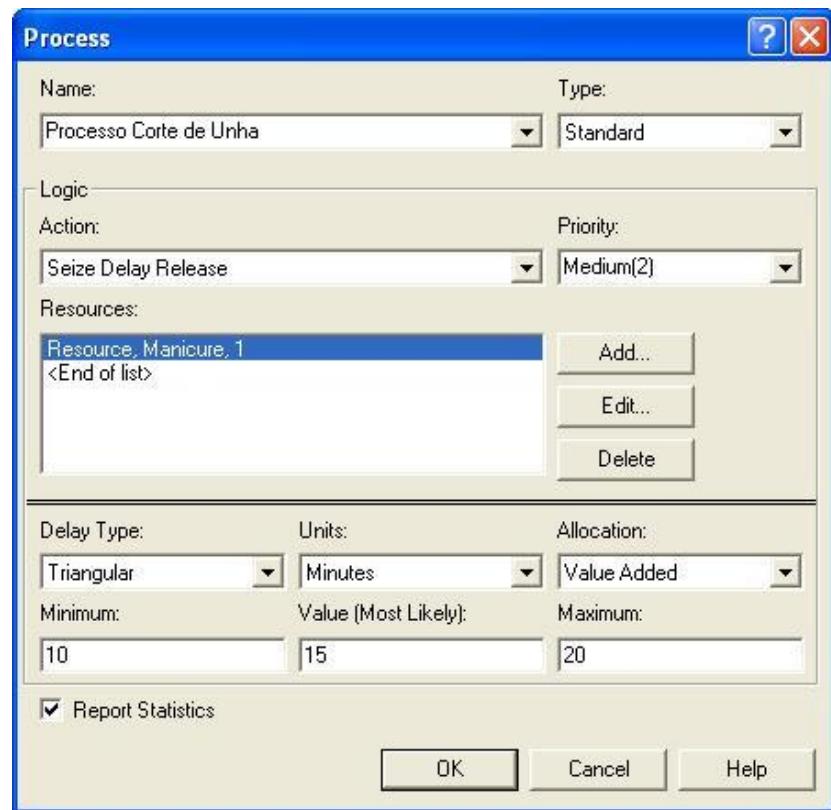
# 3º Exercício – Módulos



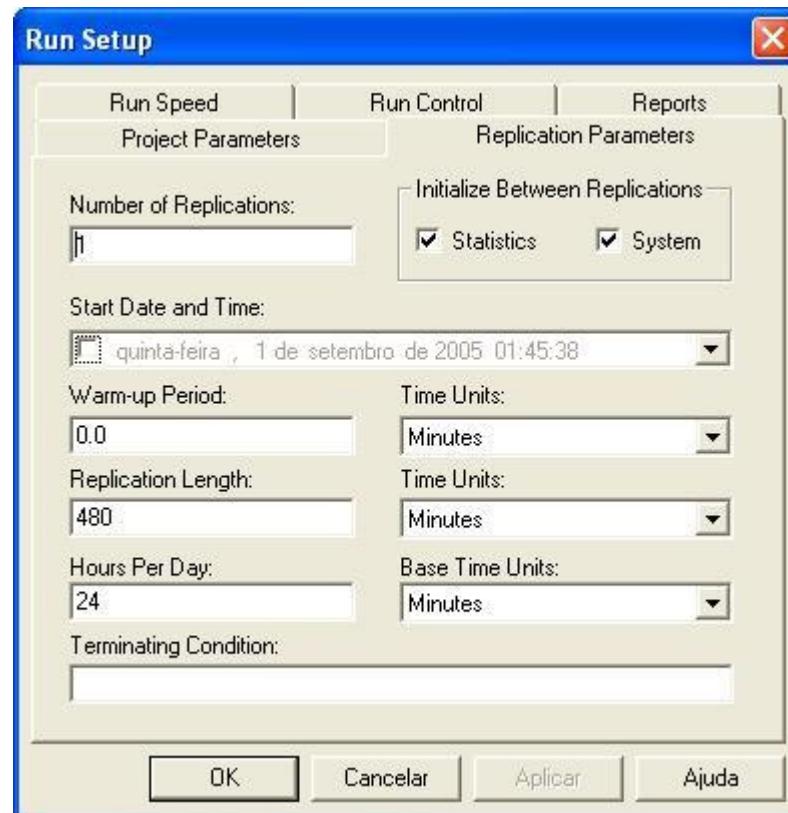
# 3º Exercício – Módulos



# 3º Exercício – Módulos



# 3º Exercício – Setup



# 3º Exercício – Conclusão

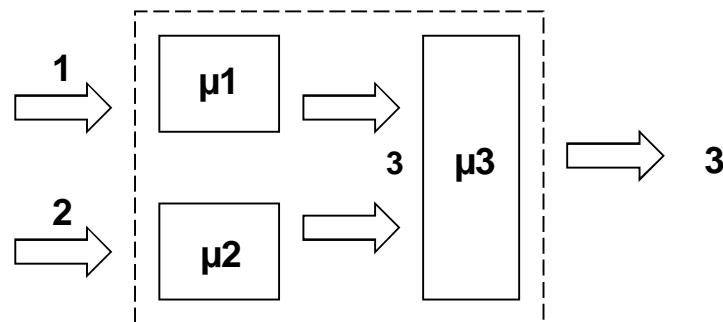
Servidor	NF	TF
Cabelo	0,16	1,46
Unha	0,01	0,44

- “ O tempo no sistema é 24, 49 minutos (relatório Entities).
- “ Total de Clientes: 45 (em 8 horas) (relatório Entities).

# 4º Exercício

4) Em um sistema de filas seqüenciais no qual peças fluem pela linha de produção, temos os seguintes valores para ritmos médios (todas as distribuições são exponenciais negativas):

- .  $\lambda_1 = 10$ ,  $\lambda_2 = 5$  (unidade: chegadas por hora – todas as distribuições são exponenciais negativas).



- .  $\mu_1 = 15$ ,  $\mu_2 = 30$  e  $\mu_3 = 20$  (unidade = atendimentos por hora – todas distribuições Erlang).

Pede-se:

- . Calcular Tempo Médio na Fila, Tamanho Médio da Fila e Taxa de Ocupação de cada servidor;

# 4º Exercício

4) Calcular Tempo Médio no Sistema e Quantidade Média de Pessoas no Sistema;  
Simule 1.000 minutos.

**Observação:** considere as distribuições de atendimento como sendo Erlang-5. A distribuição Erlang-5 deve ser fornecida ao Arena na forma:

**ERLA(ExpoMean, 5)**

sendo que ExpoMean deve ser calculado como:

**ExpoMean = TA/5,**

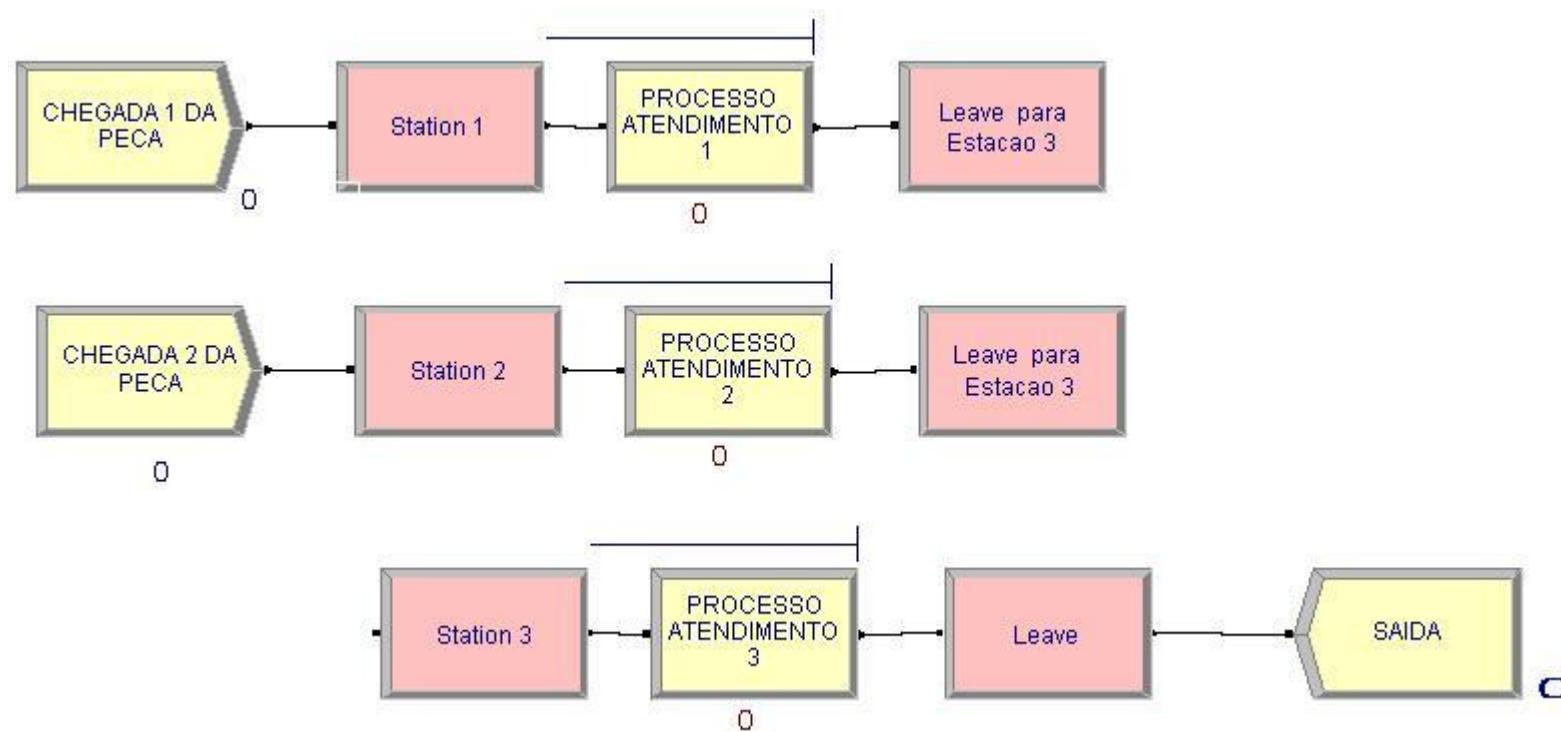
em que TA é o Tempo Médio de Atendimento.

Para trabalharmos com minutos, os valores são:

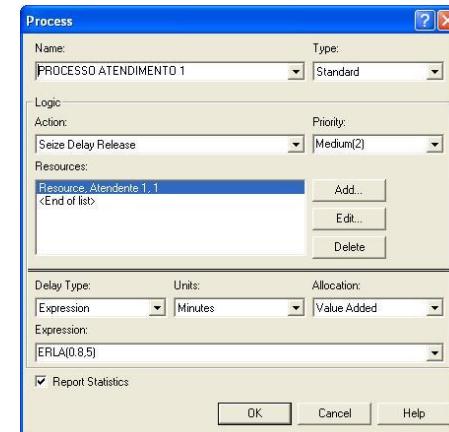
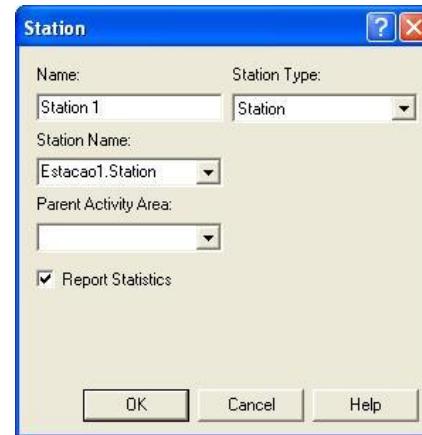
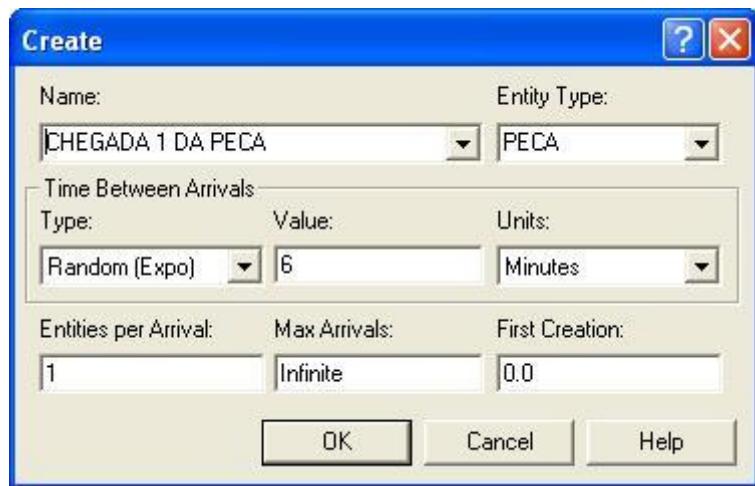
Atendente	$\mu$	TA (hora)	TA (min)	TA/5	<i>Process Time</i>
1	15	0,06	4	0,8	Erla(0, 8, 5)
2	30	0,03	2	0,4	Erla(0, 4, 5)
3	20	0,05	3	0,6	Erla(0, 6, 5)

Atenção: não se esqueça de converter ritmo de chegada em intervalo entre chegadas para fornecer os dados ao módulo Create.

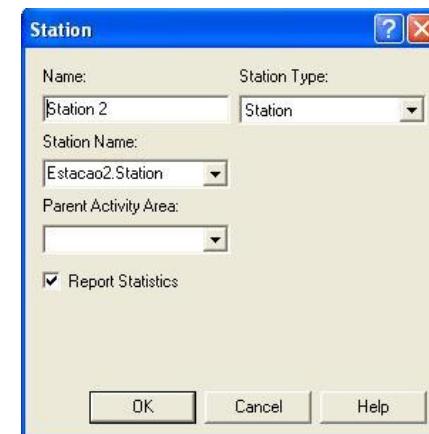
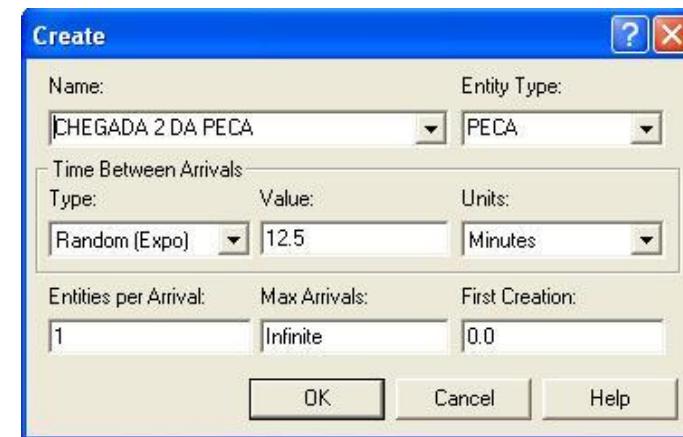
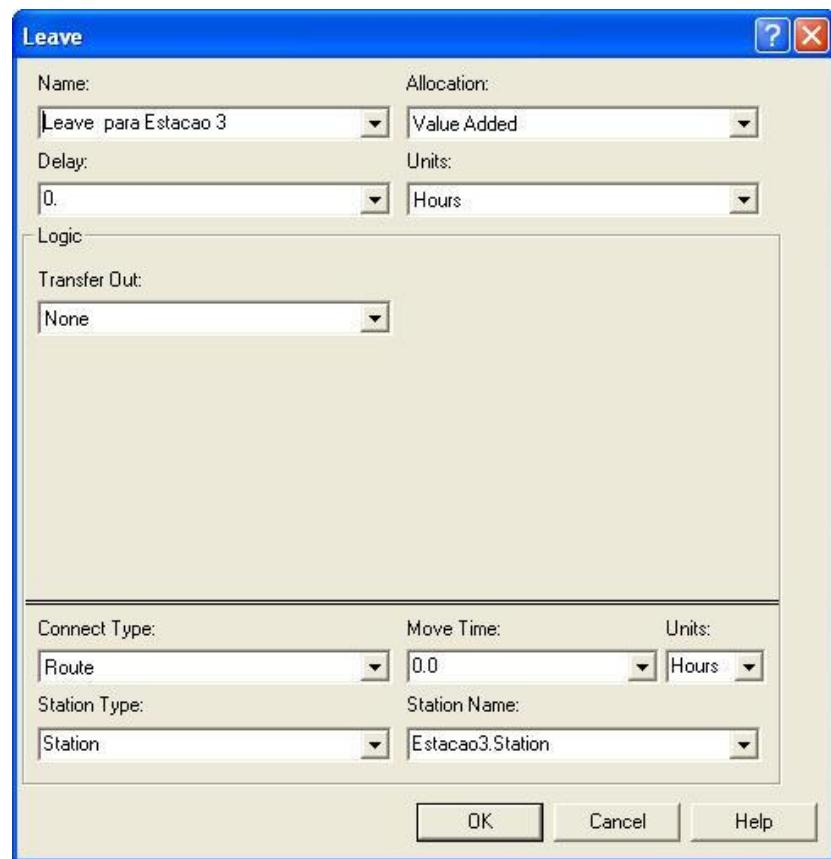
# 4º Exercício – Fluxograma



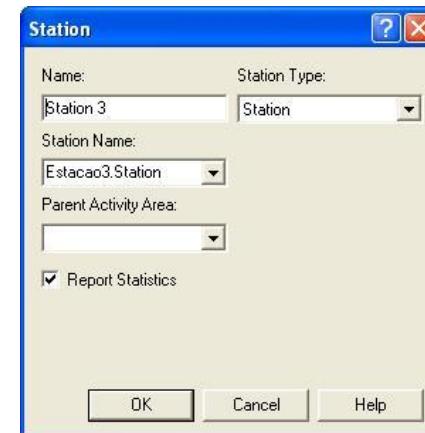
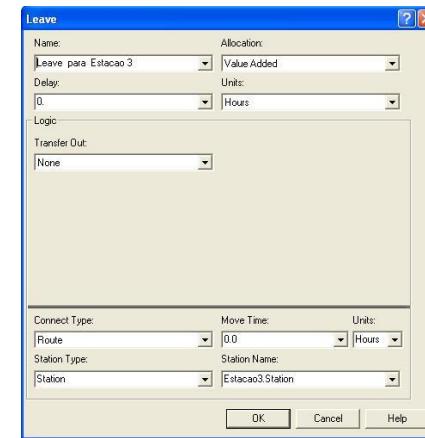
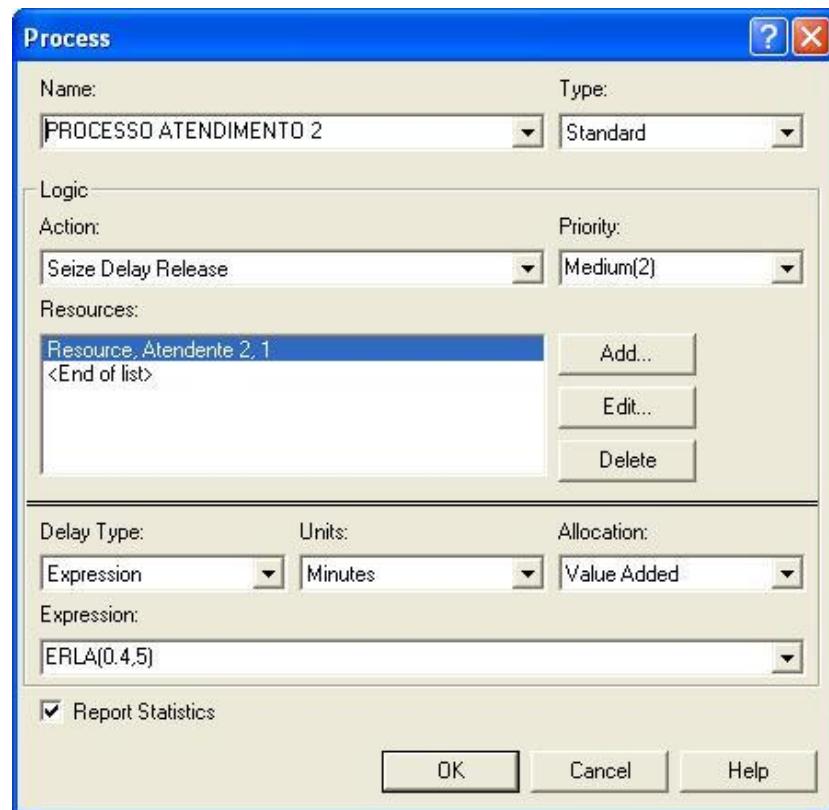
# 4º Exercício – Módulos



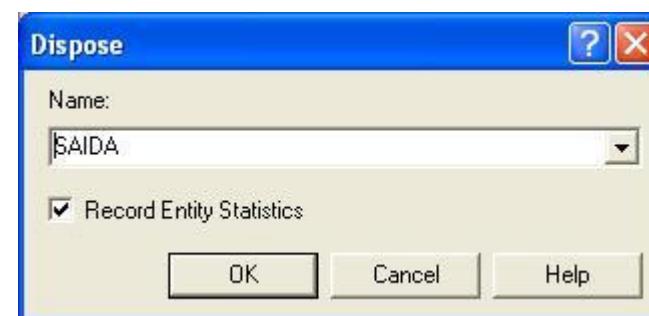
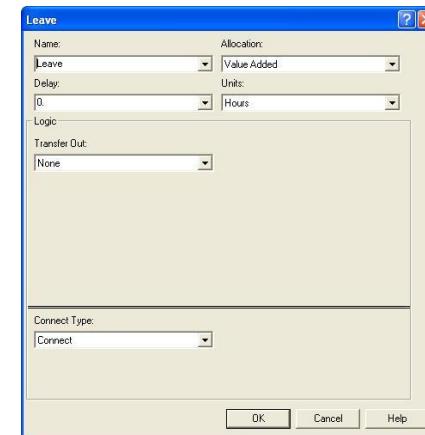
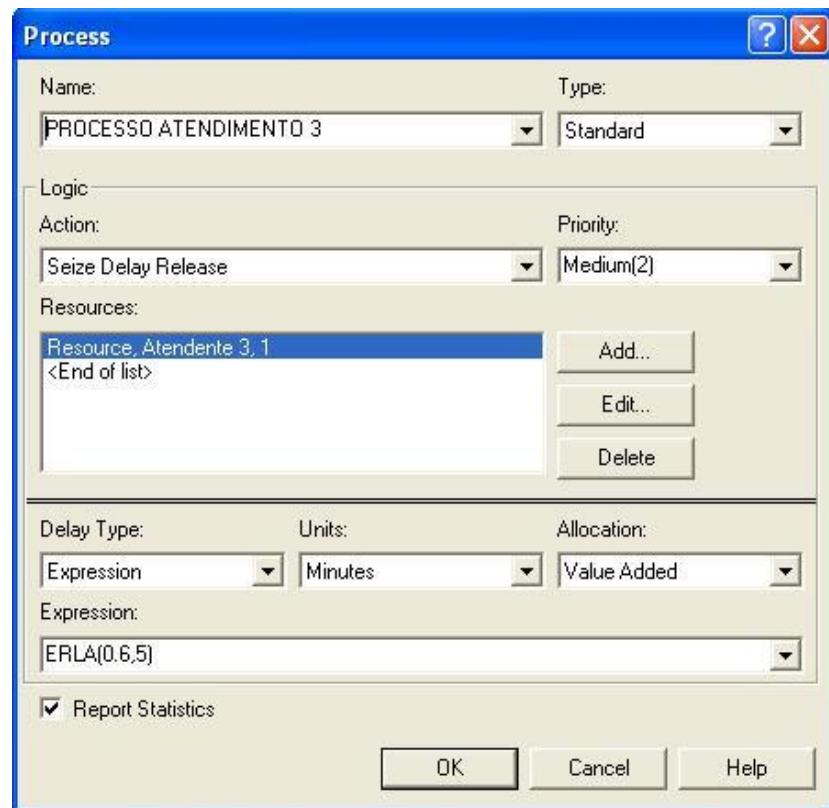
# 4º Exercício – Módulos



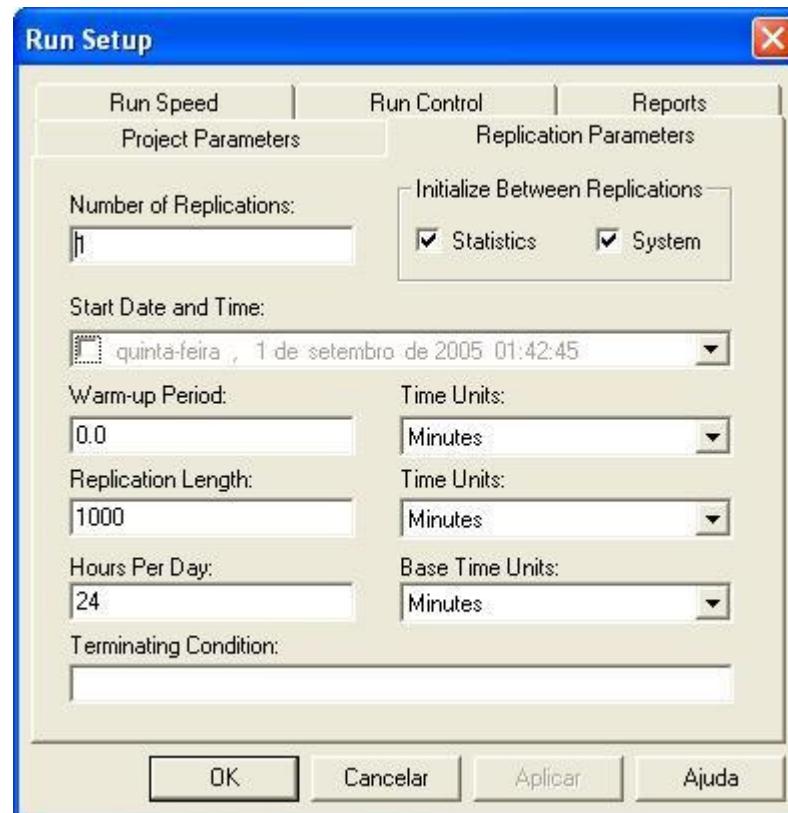
# 4º Exercício – Módulos



# 4º Exercício – Módulos



# 4º Exercício – Setup



# 4º Exercício – Conclusão

“ A.

Servidor	TF	NF	Ocupação
1	4,66	0,76	0,65
2	0,21	0,02	0,21
3	5,54	1,45	0,81

“ B. Para o sistema como um todo, temos:

NS = 3,89 (relatório Entities, página 2, campo WIP).

TS = 14,90 (Coluna Total Time do relatório Entities).

## EXERCÍCIOS PROPOSTOS

Os Exercícios a seguir devem ser implementados e resolvidos usando o Arena. São fornecidos os fluxogramas para serem implementados no ARENA.

5) Em uma fábrica chegam pedidos a cada EXPO(23) minutos e vão para a produção (MAQ.A), gastando UNIF(15,25) minutos. A seguir passam por um processo de inspeção, em que:

Duração: TRIA (15,17,20)

Percentual de falha: 30%

As peças defeituosas vão para uma estação de reparo.

As peças sem defeito vão para a expedição.

Na estação de reparo gasta-se UNIF(10,15) e a seguir, as peças vão para a expedição.

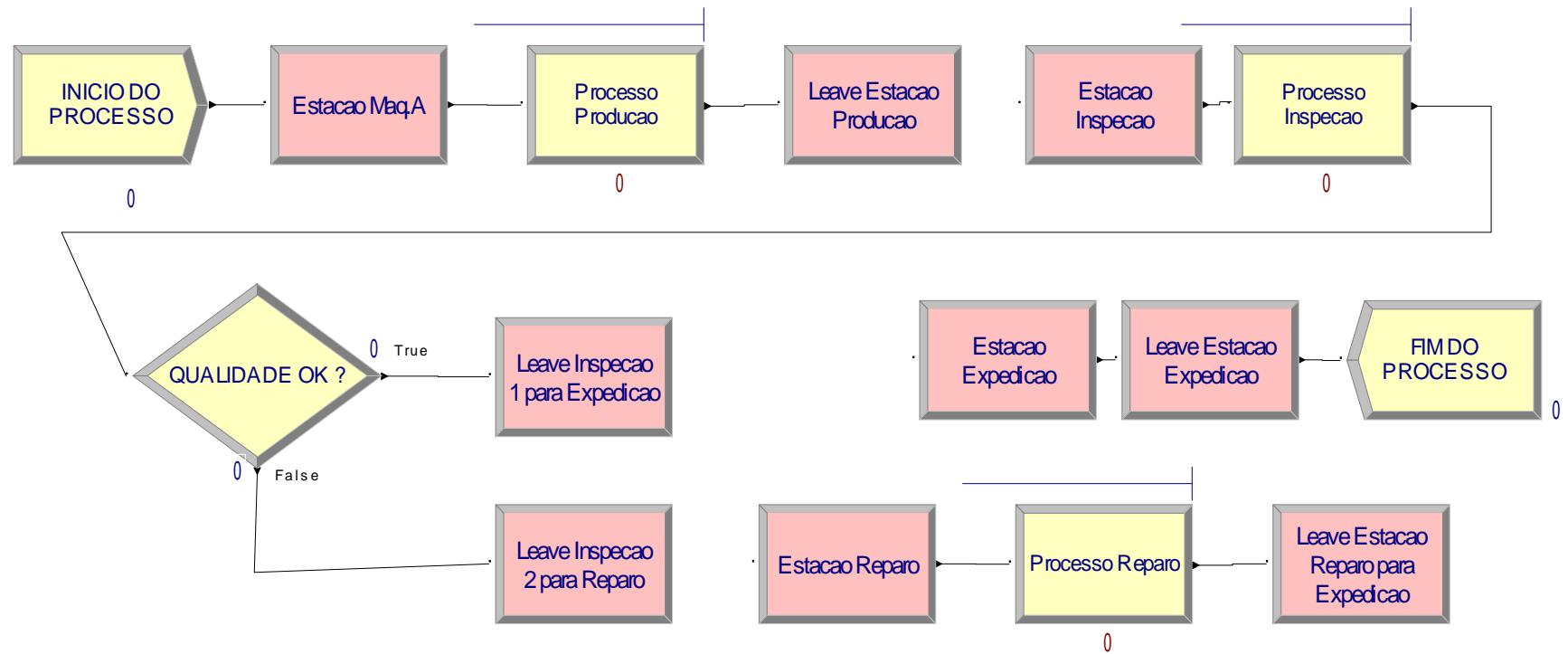
Todos os tempos de deslocamento são de 2 minutos.

Simule 1.000 minutos. Pede-se:

a) Qual a taxa de ocupação de cada servidor?

b) Qual o tempo médio em cada fila?

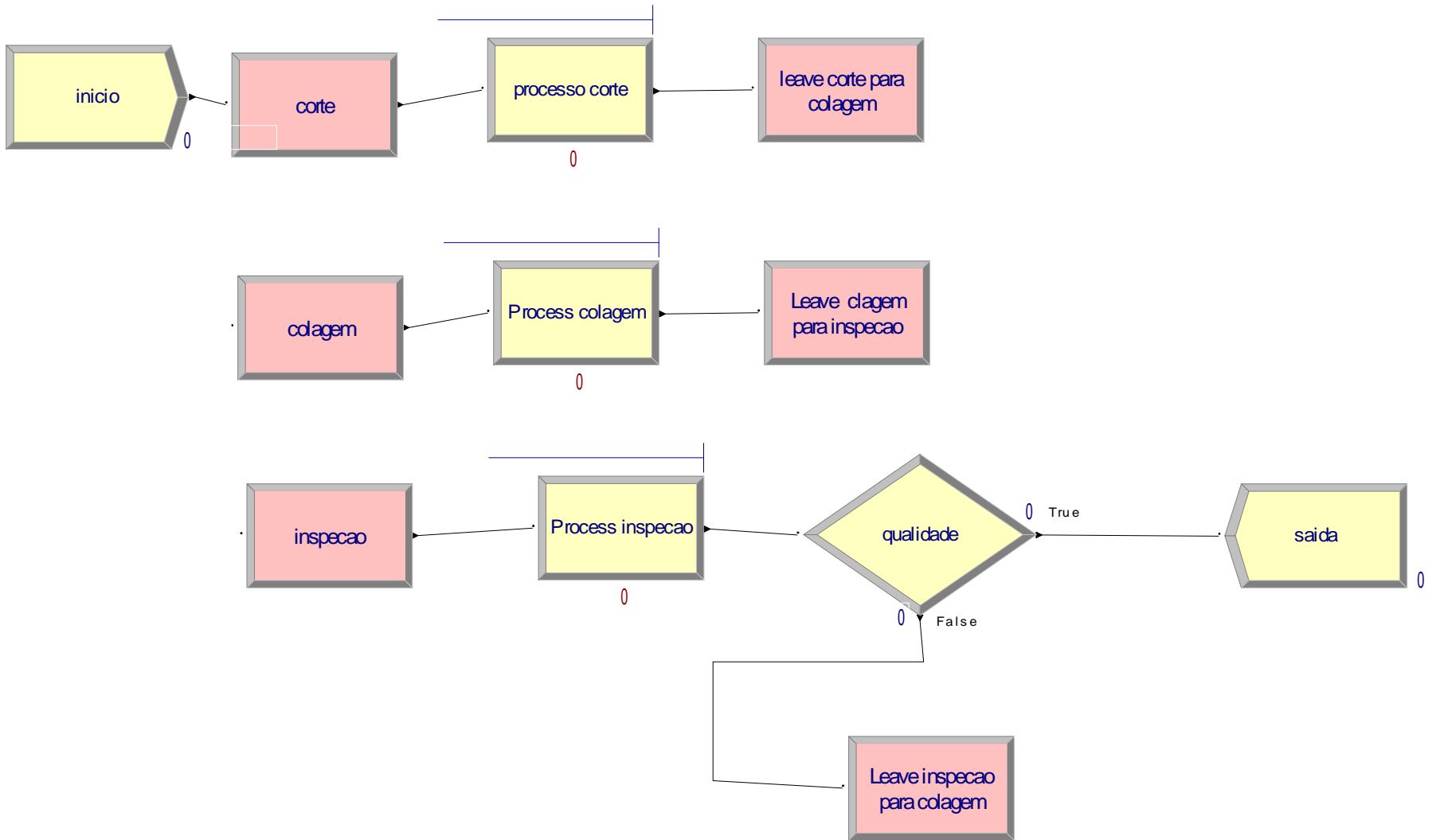
c) Qual o tempo médio para se produzir uma peça?



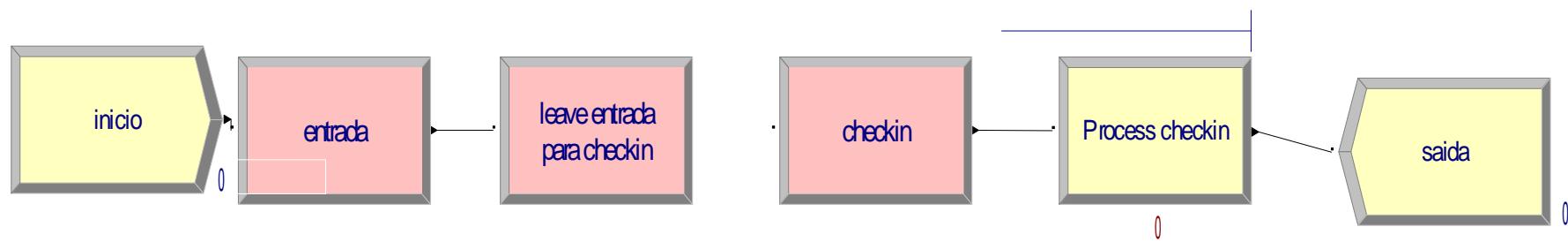
# AULA 4

” Exercícios de Aplicação sem as telas de configuração

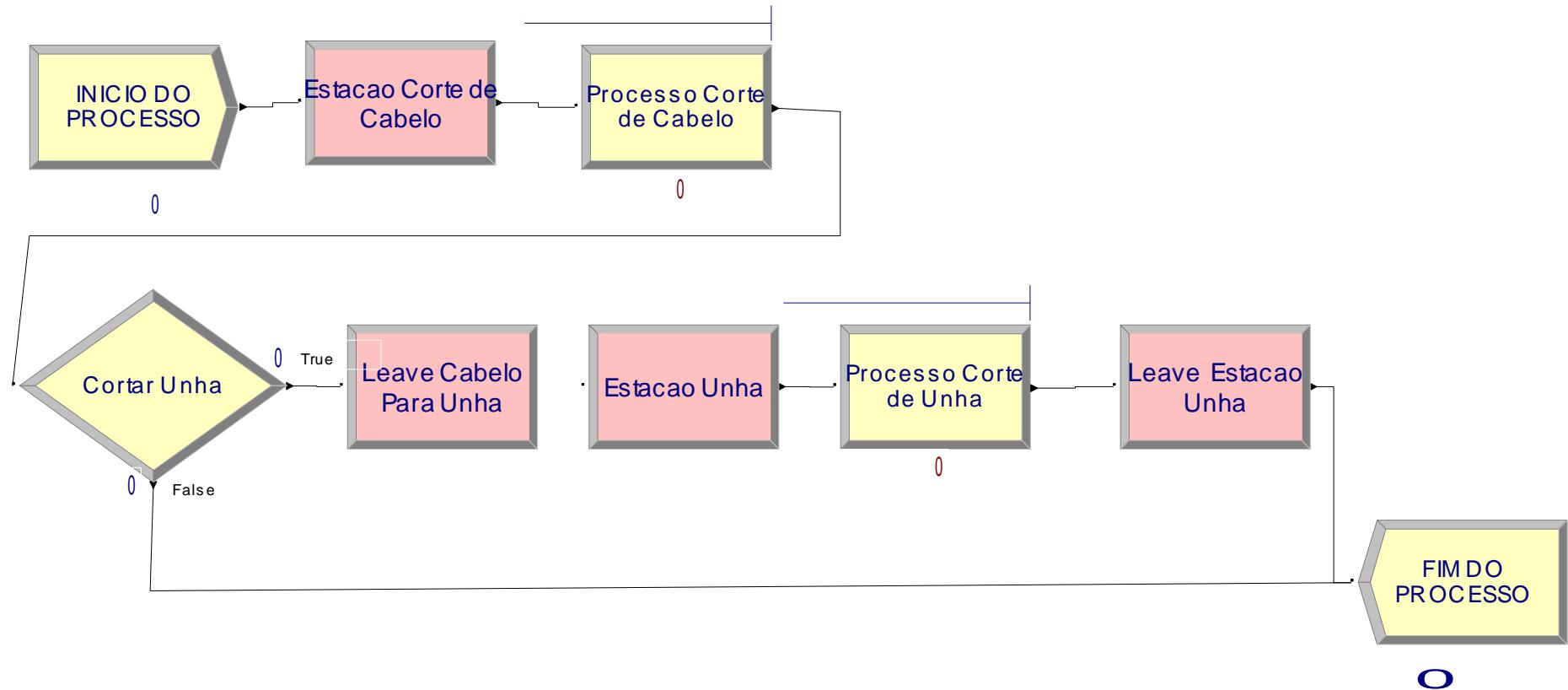
- 6) Um processo de montagem de um componente segue os seguintes passos: quando uma ordem é recebido, o primeiro trabalhador executa uma operação de corte que tem um tempo que obedece a uma distribuição normal  $NORM(12; 1,64)$  min. A colagem segue uma distribuição triangular com valores 10; 11,5 e 14 [ $TRIA(10; 11,5; 14)$ ]. Existe um processo de inspeção com uma taxa de rejeição de 15%. O tempo de inspeção é constante e seu valor é 5min. O tempo entre a estação corte e colagem é 2 min. Enquanto o tempo entre a colagem e inspeção é de 1 min. O tempo entre a rejeição do componente e a sua nova colagem é de 2,5 minutos. Os pedidos chegaram seguindo uma distribuição exponencial com média 11,3 minutos [ $EXPO(11,30)$ ]. Simular para 960 minutos e determinar:
- A produção.
  - As filas médias e tempo médio nas filas.
  - Voce contrataria mais alguém para aumentar a produção?



7) Os passageiros de um aeroporto chegam a porta principal de acordo com uma distribuição exponencial entre chegadas com média 1,6 minutos. O tempo de deslocamento da entrada para a área de Check-in obedece uma distribuição uniforme entre 2 e 3 minutos. Na entrada do Check-in os passageiros formam uma única fila até serem atendidos por um dos cinco atendentes. O processo de Check-in segue uma distribuição normal com média de 7 minutos e desvio padrão de 2 minutos. Compleando o Check-in os passageiros estão livres para irem até os respectivos portões de embarque. Execute a simulação para 16 horas e determine o número médio no sistema, o número de passageiros que completam o Check-in e o tempo médio gasto na fila.

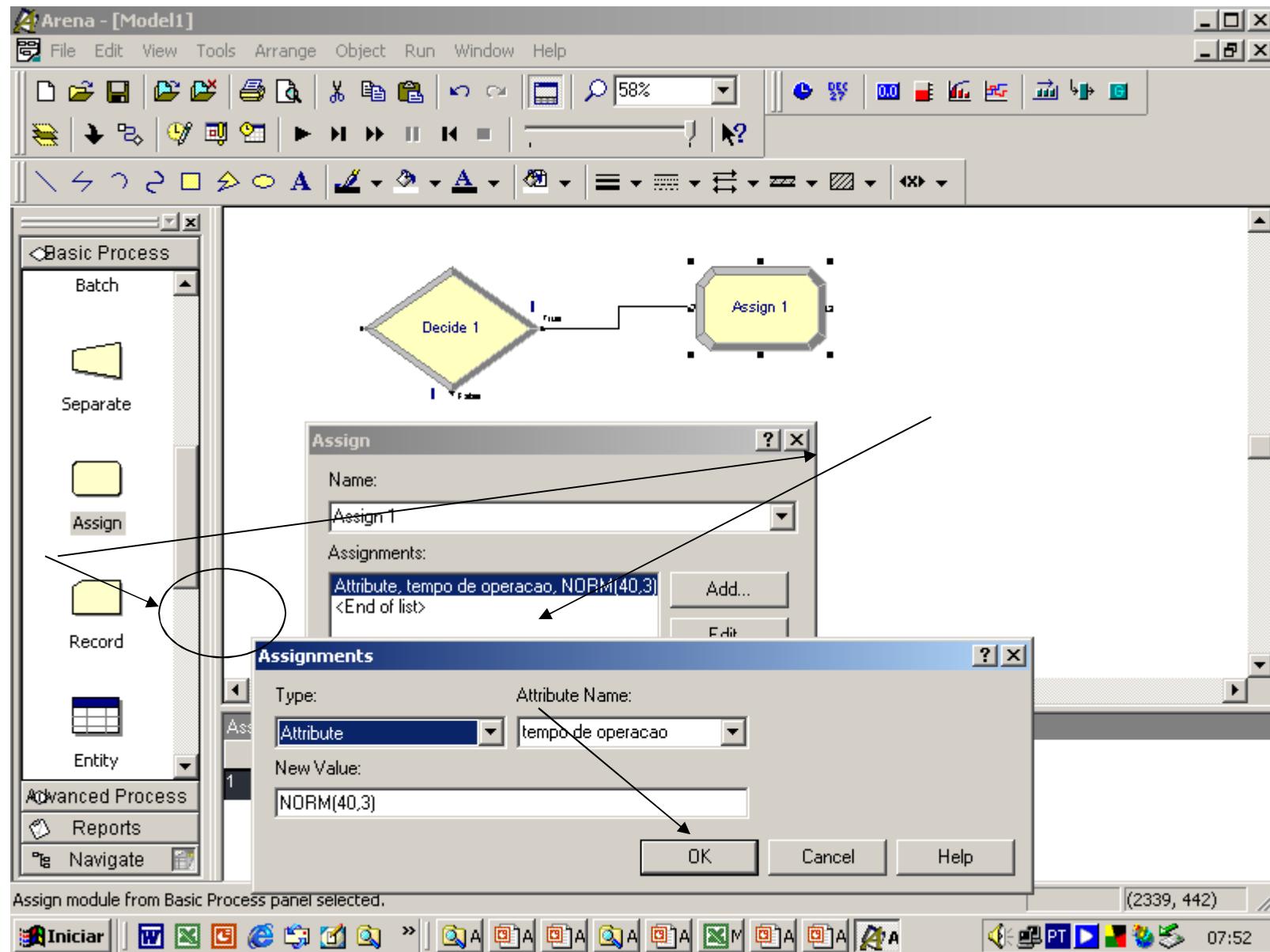


8) Em uma barbearia, clientes chegam a cada EXPO(10) minutos. Existem 3 barbeiros e o tempo de corte é de TRIA(15, 20, 25) minutos. Após cortado o cabelo, 30% dos clientes também fazem a unha com uma outra profissional, gastando TRIA(10, 15, 20) minutos. Verifique o tamanho das filas, o tempo nas filas e o tempo que um cliente gasta dentro da barbearia. O tempo de deslocamento entre uma a seção de corte de cabelo e a seção de corte de unha é de 1 minuto (o cliente gasta algum tempo conversando, etc.). Conte o total de clientes que foram atendidos no período de 8 horas.



# MÓDULOS ASSIGN E PICKSTATION

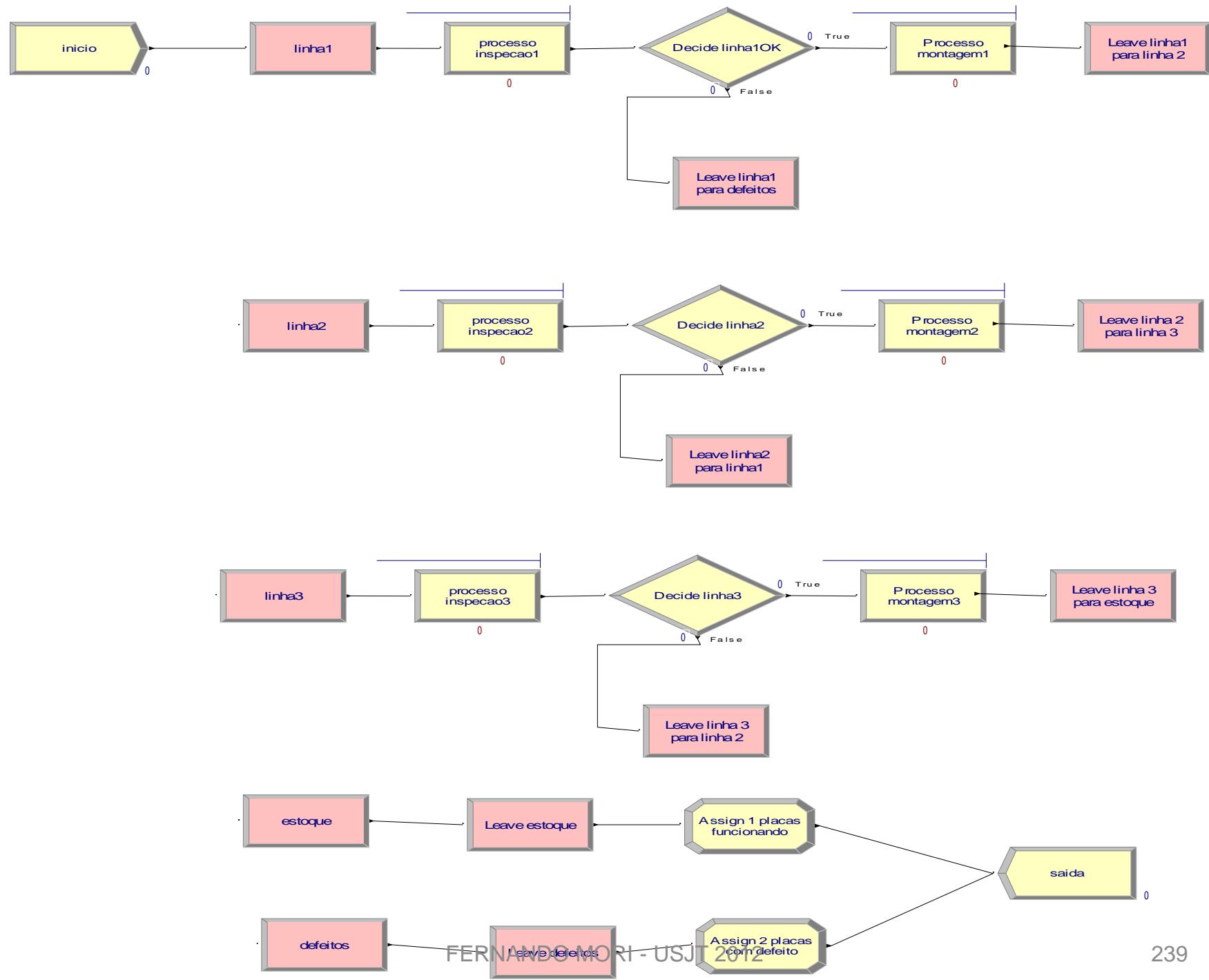
- ” O módulo assign serve para efetuar contagem das entidades que fluem por um determinado ponto do sistema.
- ” O módulo pickstation simula um processo de escolha executado pelas entidades em relação as estações de trabalho, tal como veículos que chegam a praça de um pedágio.



9) Partes chegam a uma única estação de trabalho de acordo com uma distribuição exponencial entre chegadas com média de 20 segundos. Após serem transferidas para a estação elas são processadas. O tempo de processamento é TRIA (16, 19,22) segundos. Existem varias características identificáveis que determinam se uma parte apresenta problemas em potencial. Essas partes, cerca de 10%, são transferidas para uma estação em que sofrem uma inspeção extensiva. O restante são consideradas boas e vão para fora do sistema. A distribuição do tempo de inspeção é NORM(120, 12) segundos. Cerca de 14% dessas partes não passam na inspeção e são rejeitadas. As partes que passam na inspeção são classificadas como boas e deixam o sistema. Supor que todos os tempos de transferência são de 2 minutos. Execute a simulação para 10.000 segundos para determinar o número de partes boas que deixam o sistema, o número de rejeitos e o número inspecionado.

10) Uma linha de montagem de placas de computadores consiste de 3 estações que adicionam componentes a um circuito impresso. Cada estação é composta por dois postos de serviço: inspeção e montagem. Quando uma placa entra numa estação, ela é primeiro enspecionada e se aprovada, a ela são agregados mais componentes. Se uma placa falha na inspeção, é mandada de volta ao posto de montagem anterior para retrabalho. Se uma placa falhar na primeira estação ela deverá ser retirada do sistema. Uma nova placa entra no sistema a cada 2 minutos. Os tempos para inspeção são uniformemente distribuídos entre um mínimo de 1,3 e 2,1 minutos. Os tempos de montagem seguem uma distribuição NORMAL (1,8; 0,4) minutos. As probabilidades de falhas nas diversas seções são diferentes: 0,04; 0,06 e 0,01 para as estações 1 a 3. Os tempos de transferência das placas entre os postos de uma mesma estação são desprezíveis. O tempo de transferência das placas entre posto de montagem e inspeção seguinte é 1 minuto. Entre inspeção e posto de montagem anterior (falhas) é 2,5 minutos. Modele e simule o sistema para um dia de trabalho (8 hrs ou 480 min) determinando:

- a) O número de placas produzidas sem defeito.
- b) O número de placas defeituosas.
- c) O número de placas no sistema.



## MÓDULO ASSIGN EXEMPLO

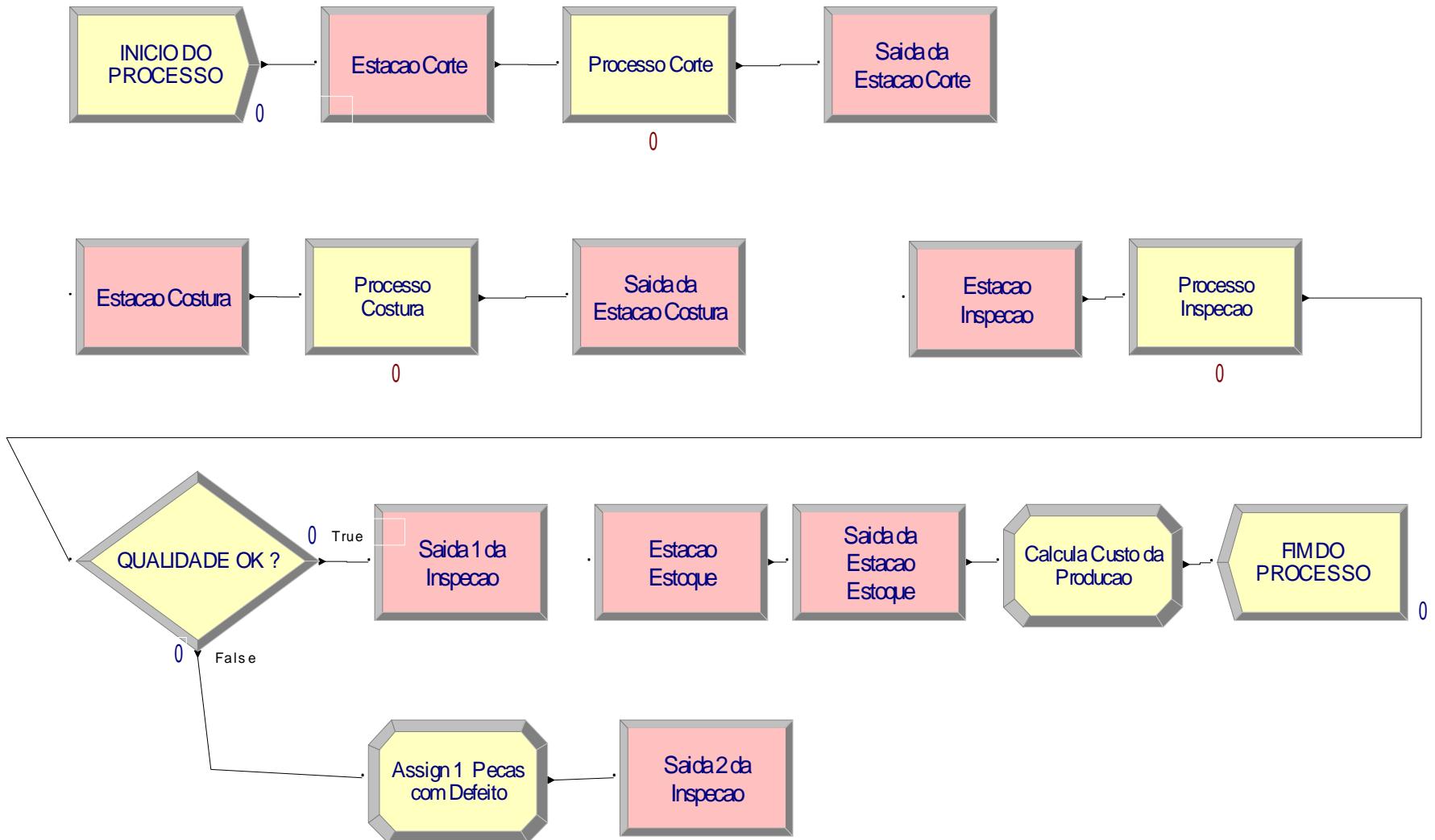
O **Módulo Assign** pertence ao template Basic Process e pode ser usado com a função de contar em uma determinada etapa do processo.

Exemplo:

- 1) Consideremos uma fábrica de roupa com três processos.

- corte: TRIA (8, 10, 12) min.
- costura: TRIA (18, 22, 28) min.
- inspeção: constant (3) minutos.

Os tempos de deslocamento entre as estações é de 2 minutos e a porcentagem de peças recusadas é de 20%. Desejamos contar as peças com defeito e calcular o custo das peças produzidas sabendo que o custo para cada peça é \$2,50.



## MÓDULO PICKSTATION

Tendo em vista que situações de desvio e escolha são muito comuns em modelagem, o Arena possui o módulo **Pickstation** que permite escolher o melhor desvio e encaminhar a entidade para o local.

Exemplo:

Construir o modelo de um pedágio ao qual os veículos chegam a cada 10 segundos em média obedecendo uma distribuição exponencial, escolhem o caixa adequado, pagam o pedágio em UNIF(20, 30) segundos e retiram-se. O veículo que chega escolhe o posto que tem a menor quantidade total de veículos nas seguintes posições:

- Fila
- Veículos dirigindo-se para o posto.
- Quantidade de recursos ocupados.

Os tempos de deslocamento entre as estações são iguais a 10 segundos.

Os nomes empregados nas estações de trabalho são:

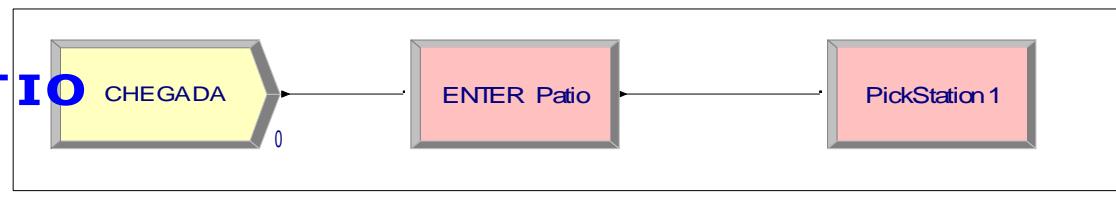
Estação	Nome da estação	Nome do processo	Nome da fila
Patio	Patio.Station		
$P_1$	$P_1$ .Station	Processo $P_1$	Processo $P_1$ .Queue
$P_2$	$P_2$ .Station	Processo $P_2$	Processo $P_2$ .Queue
$P_3$	$P_3$ .Station	Processo $P_3$	Processo $P_3$ .Queue
Saída	Saída.Station		

Simular para 8 horas.

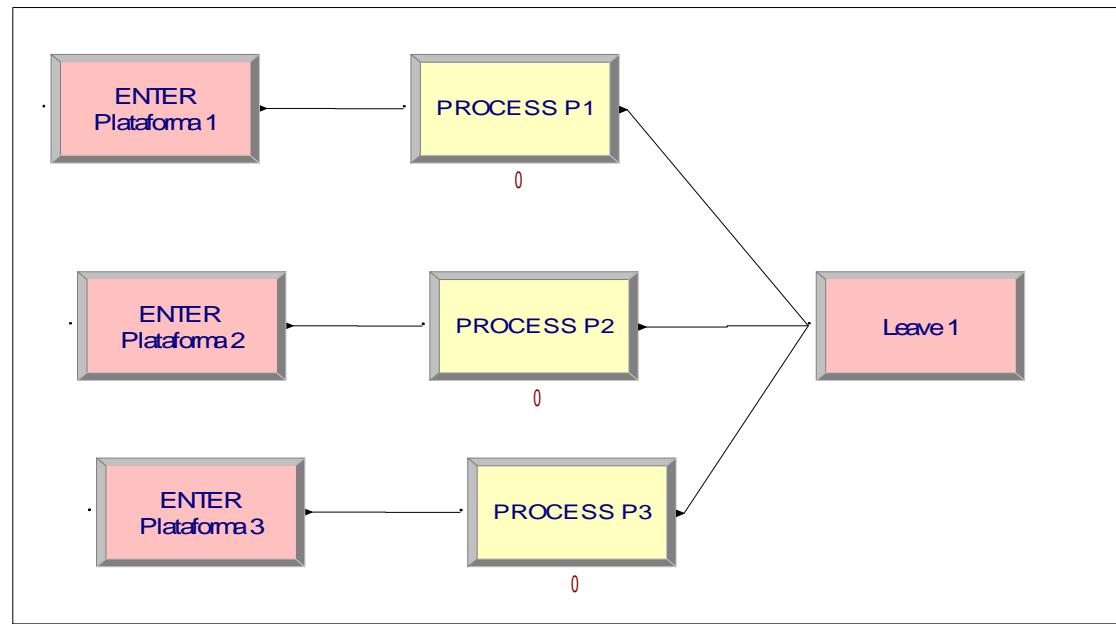
Vamos codificar o módulo Pickstation para que a rota escolhida seja aquela que fornece o menor valor para a soma da quantidade em fila + quantidade em rota + quantidade de atendentes ocupados.

O módulo Pickstation encaminha a entidade para a estação selecionada, tendo uma função leave embutido.

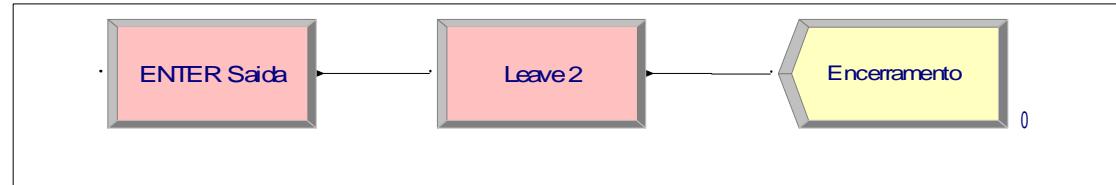
## CHEGA DA A O PATIO



## ATENDIMENTO



## SAIDA



# AULA 5

” Exercícios de Aplicação sem o Modelo Arena

# EXERCÍCIOS GERAIS

1) Desenvolva um modelo de um sistema serial simples composto de três processos. As entidades chegam ao sistema com um tempo médio entre chegadas de 10 minutos. Elas são imediatamente enviadas para o Processo 1 que tem um fila ilimitada e um único recurso com um tempo médio de serviço de 9 minutos. Completando este processo elas são enviadas para o Processo 2 que é idêntico ao Processo 1, e depois ao processo 3 que também é idêntico ao processo 1. Simular para 20000 minutos e obter o número médio na fila de cada processo e a produção ao final de um ciclo para os casos:

Execução 1 : tempo entre chegadas exponencial e tempo de serviços exponencial.

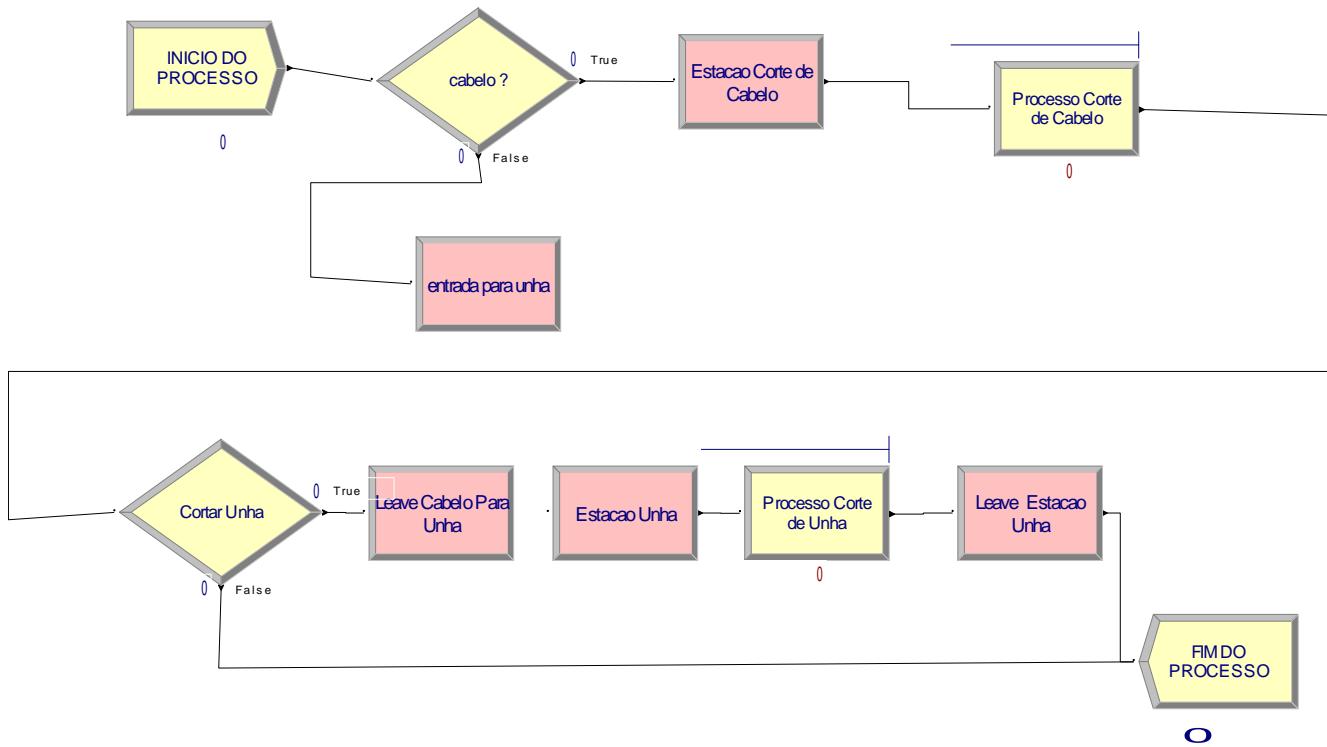
Execução 2 : tempo entre chegadas constante e tempos de serviço exponencial.

Execução 3 : tempo entre chegadas constante e tempos de serviços constantes.

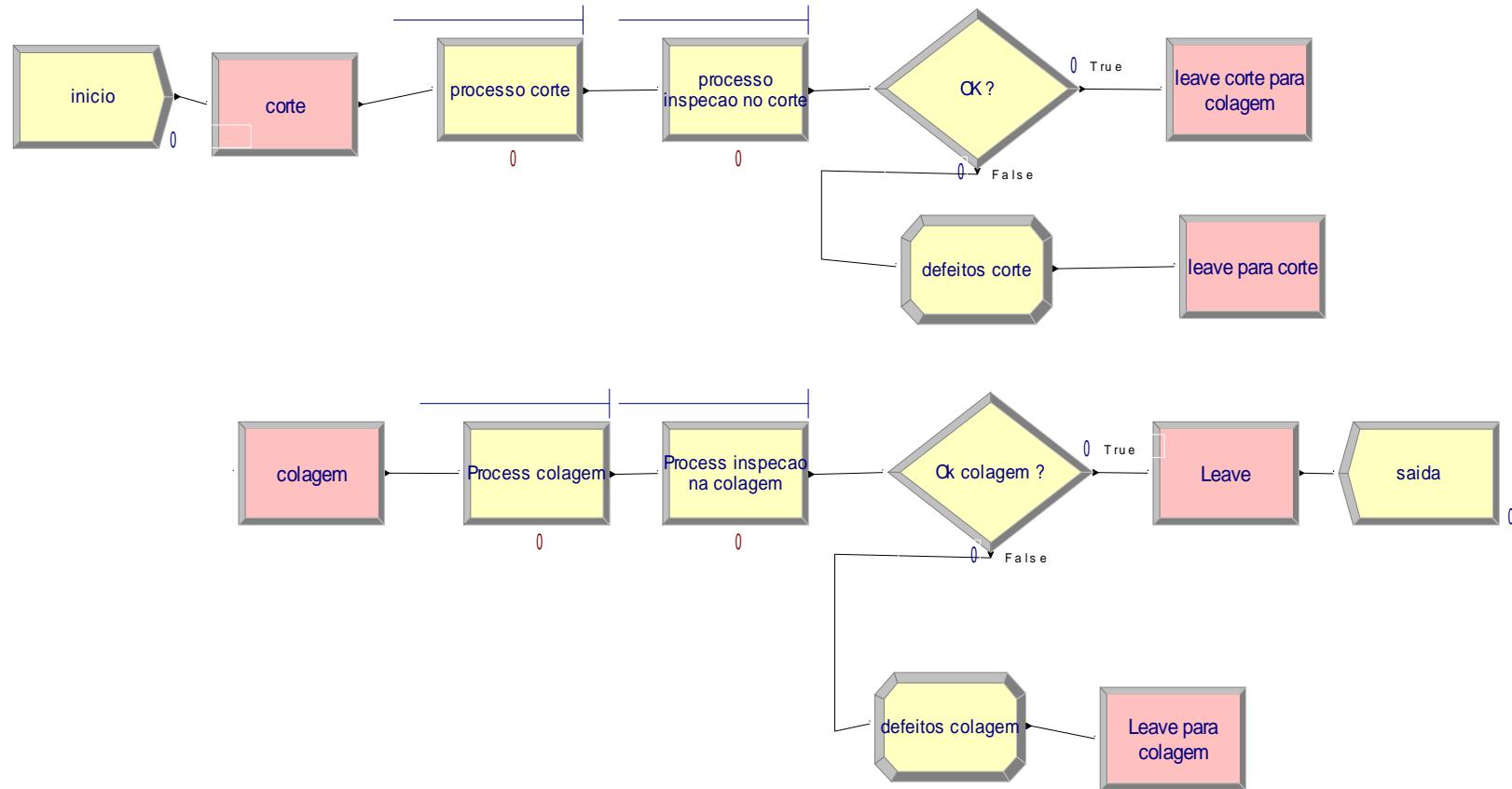
Execução 4: tempo entre chegadas exponencial e tempos de serviços constantes.

2) Construir o modelo de um pedágio ao qual os veículos chegam a cada 10 segundos em média obedecendo uma distribuição exponencial, escolhem o caixa adequado, pagam o pedágio em  $UNIF(15, 25)$  segundos e retiram-se. O veículo que chega escolhe o posto que tem a menor quantidade total de veículos na fila. Existem 6 postos de atendimento, cada um com apenas um atendente. Os tempos de deslocamento entre as estações são iguais a 15 segundos. Use o módulo pickstation e simule para um período de 8 horas. Use o módulo assign para contar quantos veículos passam em cada um dos seis postos de pedágio.

3) Em uma barbearia, clientes chegam a cada EXPO(5) minutos. Na entrada, 25% dos clientes decidem fazer apenas unha, e o tempo que o cliente gasta para ir até a manicure é 3 minutos. Existem 2 barbeiros e o tempo de corte é de TRIA(15, 20, 25) minutos. Após cortado o cabelo, 30% dos clientes também fazem a unha com uma outra profissional, gastando TRIA(10, 15, 20) minutos. Verifique o tamanho das filas, o tempo nas filas e o tempo que um cliente gasta dentro da barbearia. O tempo de deslocamento entre a seção de corte de cabelo e a seção de corte de unha segue uma distribuição uniforme com mínimo de 2 e máximo de 3 minutos. (o cliente gasta algum tempo conversando, etc.). Conte o total de clientes que foram atendidos no período de 8 horas e as filas médias. Devemos contratar mais manicures ou barbeiros? Por-quê? Use o fluxograma dado a seguir.



4) Em uma confecção os pedidos chegam de acordo com uma distribuição EXPO(3) minutos. Em seguida passam por um processo de corte no qual o cortador gasta um tempo que obedece a uma distribuição normal com média 1,5 minutos e desvio padrão 0,5 minutos. Em seguida passa por um processo de inspeção que recusa 43% dos produtos fazendo com que eles passem novamente pelo processo de corte. Em seguida vem o processo de colagem em que o colador gasta um tempo que obedece a uma distribuição triangular com 2;4 e 6 minutos. Em seguida vem o processo de inspeção que recusa 44% dos produtos e faz com que eles passem novamente pelo processo de colagem. Ambos os processos de inspeção são realizados por inspetores diferentes que gastam um tempo que obedece a uma distribuição uniforme com mínimo de 1 minuto e máximo de 2 minutos. Todos os tempos de deslocamento entre as estações são iguais a 1 minuto. Determine a produção para um período de 8 horas, o tamanho médio das filas. Conte o número de peças com defeito e que foram retrabalhadas usando o módulo assign. Justifique claramente.



5) Partes em uma linha de montagem chegam a uma única estação de trabalho de acordo com um tempo entre chegadas que obedece a uma distribuição exponencial com média de 20 segundos. Após serem transferidas para a estação as partes são processadas. O tempo de processamento obedece a uma distribuição  $TRIA(16,19,22)$  segundos. Existem várias características visuais identificáveis que determinam se uma parte tem um problema potencial de qualidade. Estas partes, cerca de 10% são transferidas a uma estação onde sofrem uma inspeção extensiva. As outras partes são consideradas boas e transferidas para outro sistema. O tempo de inspeção é  $NORM(120,12)$  segundos. Cerca de 14% dessas partes não passam na inspeção e são transferidas para a sucata. As partes que passam pela inspeção são classificadas como boas e levada para fora do sistema. Supor que todos os tempos de transferência são de 2 minutos. Execute a simulação para 10000 segundos e determine o número de partes que foram inspecionadas, o número de partes que aceitas como boas e o número das que viraram sucata.

6) Uma empresa recebe no inicio do dia um lote de 100 peças que deverão ser trabalhadas. Essas peças passam por um exame inicial, feito por um examinador que leva um tempo sempre igual a 2 minutos. Nesse exame 25% das peças são recusadas. Em seguida temos um processo de usinagem executado por um torneiro mecânico que gasta um tempo TRIA ( 8,10,12) com cada peça. Na seqüência temos um processo de polimento feito por um polidor que gasta com cada peça TRIA (12,15,17) minutos e um processo de montagem executado por um montador gasta TRIA(20,25,30) minutos. Em seguida temos o processo de inspeção em que o inspetor sempre gasta 3 minutos e recusa 30% das peças enviando-as de volta par o setor de polimento. As peças aceitas vão para o estoque. Planejar a quantidade de funcionários necessária para que no final de um dia de 8 horas de trabalho as 100 peças tenham sido processadas com o menor custo possível.

7) Um posto do INSS recebe pacientes a uma taxa EXPO (120) segundos. Esses pacientes passam por uma triagem inicial em que são verificados seus documentos e escolhida uma especialidade médica. Esse processo é executado por um atendente que leva um tempo TRIA (30,90,120) segundos. Em seguida um fiscal do INSS examina o cadastro do paciente gastando um tempo TRIA (60,120,180). Esse fiscal recusa 30% dos atendimentos. Os que são aceitos para o atendimento médico, passam para o estágio seguinte feito por uma enfermeira que os entrevista gastando um tempo TRIA (200,300,400) segundos e manda voltar novamente a triagem 50% dos pacientes para nova análise da especialidade médica. Os 50% aceitos passam pelo atendimento médico, onde o médico gasta um tempo TRIA (600,900,1200) segundos com cada paciente. O posto possui 2 atendentes, 1 fiscal, 2 enfermeiras e 1 médico. Temos necessidade de mais funcionários? Em um dia quantos pacientes são atendidos e quantos são recusados? Qual o tamanho médio das filas?

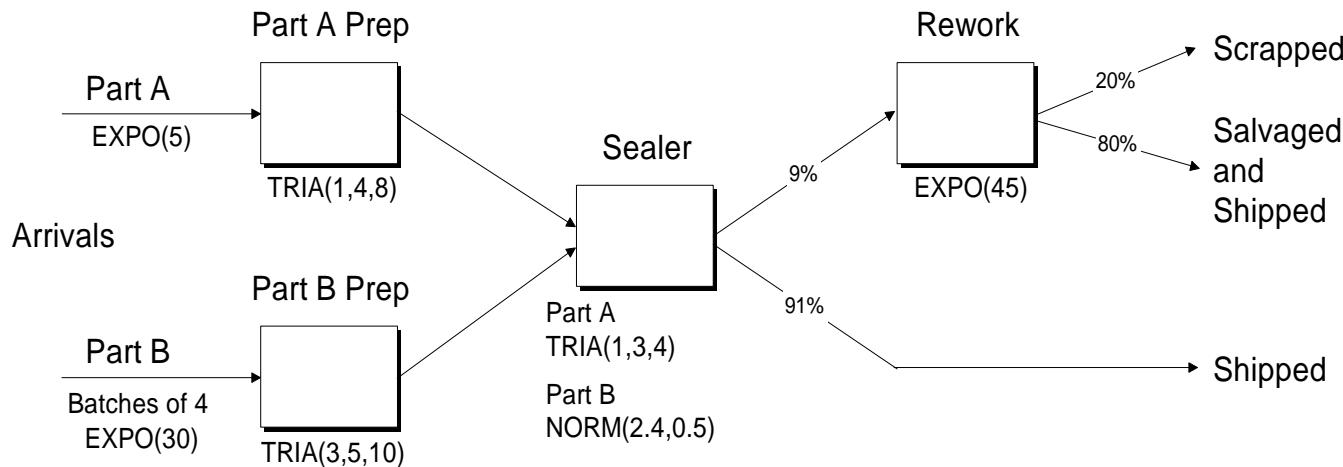
# AULA 6

” Um modelo realista que deve ser elaborado seguindo as etapas fornecidas a seguir. Neste modelo iremos incluir uma escala para um dos recursos e animações.

# O quê será feito

- ” Modelo 1: Sistema de montagem/testes de equipamentos eletrônicos
  - . abordagens de modelagem;
  - . estações e animação;
- ” Modelo 2: Melhorando a modelagem
  - . recursos, estados dos recursos, programação;
  - . salvando dados estatísticos;
  - . analisador de saída (somente dados);
- ” Modelo 3: Melhorando a animação
  - . filas, representação de entidades e recursos;
  - . adicionando gráficos e variáveis.

# Sistema de Montagem/Teste (Modelo 1)



- ✓ Produz dois tipos de unidades eletrônicas seladas (A,B);
- ✓ chegadas: caixas metálicas processadas para receber as partes eletrônicas;
- ✓ parte A, parte B — áreas de preparação separadas;
- ✓ ambas vão ao equipamento de selagem para montagem e teste — então são enviadas se perfeitas ou caso contrário reprocessadas;
- ✓ reprocessamento — recuperação ou descarte.

# Parte A

- ” chegadas:  $\exp(5)$  min.;
- ” tempo de trânsito entre estações: 2 min.:
  - . não há espera por pessoa ou veículo para transporte — partes têm locomoção própria (será relaxado mais adiante);
- ” vai à área de preparação da parte A:
  - . Processo = (máquina + aparar + limpar) ~ tria (1,4,8)
- ” vai ao lacrador:
  - . Processo = (montar + testar) ~ tria (1,3,4) min.
  - . 91% passam e são despachadas;
  - . caso contrário, vão para o reprocessamento;
- ” reprocessamento (+ teste) ~  $\exp(45)$ 
  - . 80% passam e são despachadas; 20%, descartadas.

# Parte B

- “ chegadas: *lotes* de 4,  $\exp(30)$  min.;
- “ tempo de trânsito entre estações: 2 min.;
- “ vai à área de preparação da parte B:
  - . Processo = (máquina + aparar + limpar)  $\sim$  tria (3,5,10)
- “ vai ao lacrador
  - . Processo = (montar + testar)  $\sim$  norm (2.4, 0.5) min. ,  
*diferente* da parte A, embora na mesma estação;
  - . 91% passam e são despachadas; 9% retornam para reprocessamento;
- “ reprocessamento (+ teste) =  $\exp(45)$  min.:
  - . 80% passam e são despachados; 20%, rejeitados.

# Condições de Execução, Saída e Animação

- ” inicia vazio, desocupado e roda por 2.000 min.;
- ” saída:
  - . utilização de todos os recursos;
  - . número em cada fila;
  - . tempo em cada fila;
  - . tempo no sistema, separados por partes despachadas, recuperadas/despachadas e descartadas;
- ” animação:
  - . filas, recurso ocupado/livre, como anteriormente;
  - . movimentação das entidades entre as estações (atraso de 2 min.).

# Transferência entre Estações

- ” envio de uma entidade de uma estação para outra;
- ” opções de modelagem:
  - . *conectado* (transferência em tempo zero)
  - . *rotear* (possivelmente um tempo positivo, sem restrições);
  - . *restrito ao recurso*;
  - . *transportadores*;
  - . *correias transportadoras*;
- ” possibilidade de animação para cada opção de transferência;
  - . para roteamento (route): rotear objeto pelo barra de animação.

# Melhorando o Modelo (Modelo 2)

“ uma história:

- . modelo 1 é apresentado ao gerente de produção;
- . que chama à atenção que isto é somente o primeiro turno de um dia com dois turnos — no segundo turno, há dois operadores na estação de reprocessamento (a estação “gargalo”);
- . Podemos pensar em um turno de 8 horas com um reparador e o turno seguinte com dois reparadores.

# Programação (cont.)

” área “server data”:

- . para o tipo da capacidade (capacity type), escolher “schedule” em lugar de “capacity” (pelo menu);
- . na caixa “capacity”, mudar para:
  - ” “schedule” — dar um nome para o horário;
  - ” escolher entre “preempt/ignore/wait” — o quê fazer quando o recurso estiver ocupado e programado para ficar indisponível?
- . clicar botão “schedule...”:
  - ” caixa de diálogo “schedule”;
  - ” adicionar pares capacidade/duração (capacity, duration);
  - ” se todas as durações forem especificadas, a programação se repetirá indefinidamente;
  - ” caso contrário, a duração padrão (infinito) será assumido.

# Melhorando a Animação (Modelo 3)

- “ animação padrão “espartana”:
  - . usualmente, suficiente para verificação e validação;
- “ freqüentemente, há necessidade de personalizar e melhorar um pouco:
  - . maior realismo e impacto;
- “ é possível posicionar a animação longe do modelo lógico:
  - . útil para grandes modelos e animações complexas.

# Modificando Animação das Filas

- ” aumentar tamanho (clicar, arrastar, possivelmente segurando a tecla *shift*);
- ” re-orientar por rotação, para melhorar realismo;
- ” mudar a “forma” da fila do padrão linha (*Line*), para ponto (*Point*) — localização fixa das entidades:
  - . duplo clique na fila;
  - . selecionar tipo de ponto;
  - . clicar no botão “Points ...”;
  - . sucessivamente clicar no botão “Add” e “OK”;
  - . arrastar na tela;
  - . escolher botão “Rotate”, para mostrar movimentação das entidades.

# Modificando a Representação das Entidades

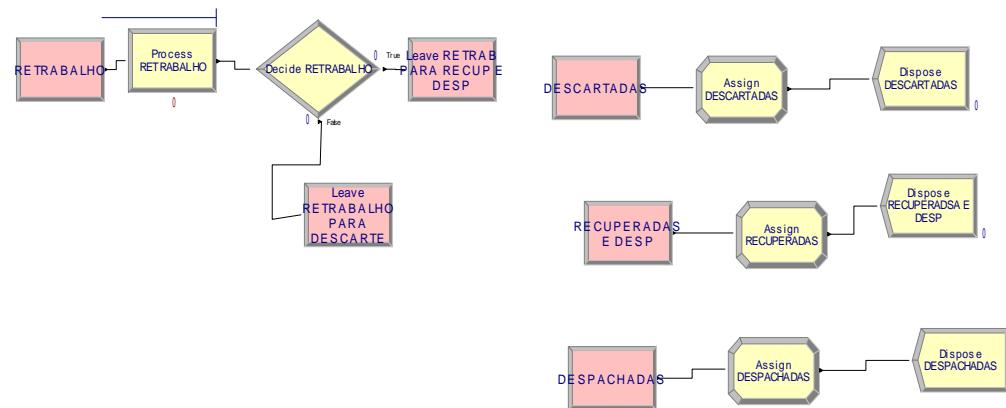
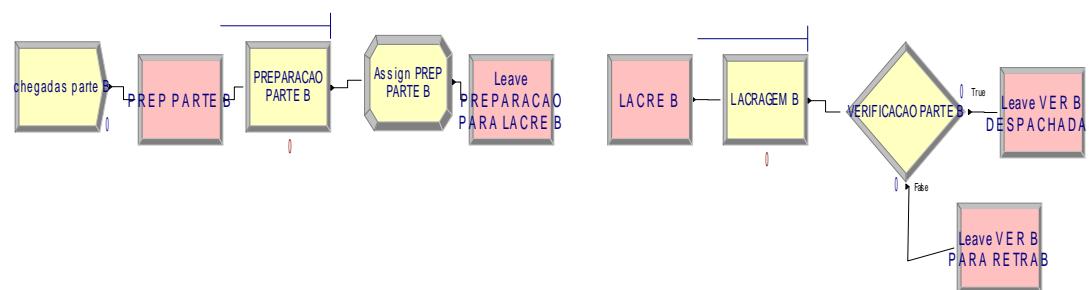
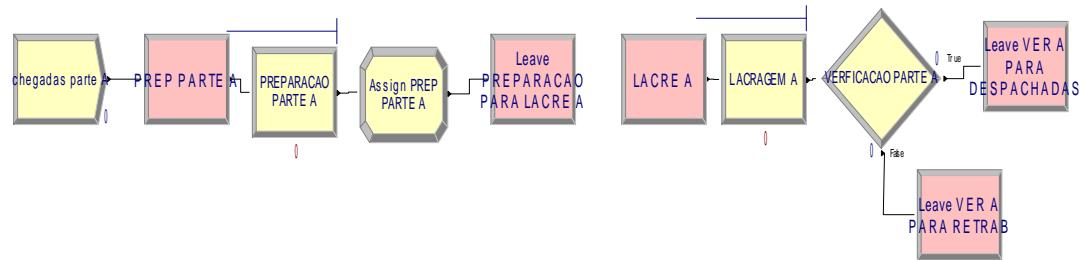
- ” diferenciar entidades diferentes, mudanças durante o processo, representar realisticamente os lotes (batches);
- ” representação padrão: acima do módulo simular (simulate);
- ” definir diferentes representações: **botão Animate...** e diálogos:
  - . chegada (arrive);
  - . servidor (server);
  - . inspeção (inspect);
- ” dar um nome à representação.

# Alterando Representação de Recursos

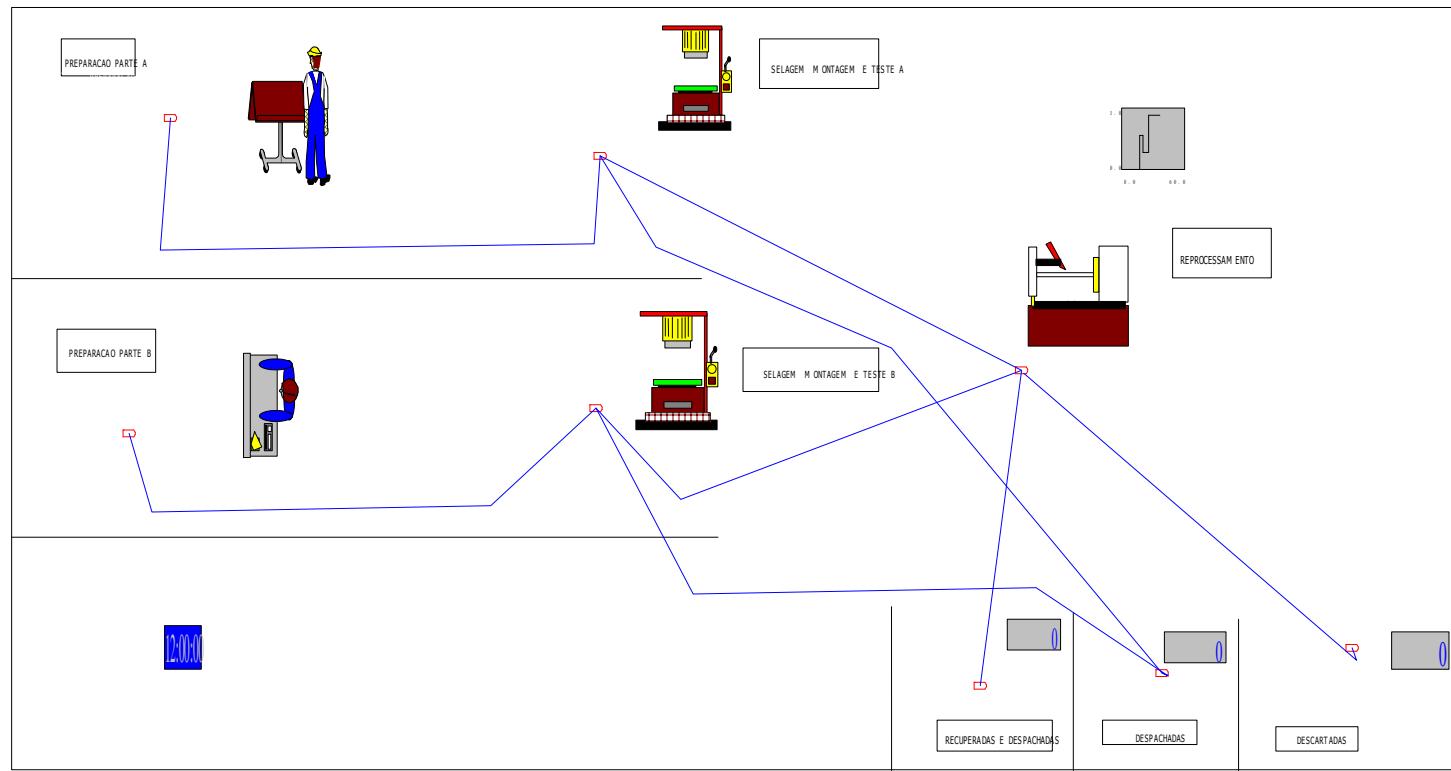
- “ realismo e indicação do estado (ocupado, livre, etc.);
- “ duplo clique e edição semelhante às entidades;
  - . trabalho de arte;
  - . bibliotecas de figuraas (arquivos .**p1b**);
  - . exemplo: melhorar o recurso “sealer” no modelo 2;
- “ ajustar tamanho — fator de escala;
- “ e os recursos com capacidade múltipla?

# SOLUÇÃO PROPOSTA

“ Colocamos a seguir a solução para este modelo proposto. Observe que incluimos também a animação de acordo com a estrutura lógica construída.



# FABRICA EQUIPAMENTOS ELETRONICOS



# AULA 7

” Utilização da ferramenta Input Analyser do ARENA

# Entradas Determinísticas *versus* Aleatórias

- ” *determinística*: não-aleatória, valores fixos:
  - . número de unidades de um recurso;
  - . tempo de trânsito entre entidades (?);
  - . tempos entre chegadas e de atendimento (?);
- ” *aleatório (estocástico)*: modelado como uma distribuição:
  - . tempos de trânsito, entre chegadas, atendimento;
  - . qual distribuição? Quais parâmetros?
  - . tornam a saída também aleatória;
- ” não assuma aleatoriedade automaticamente — validade.

# Coletando Dados

- ” geralmente difícil, caro, frustrante e aborrecido:
  - . o sistema pode não existir;
  - . dados disponíveis das coisas erradas — pode ser necessário alterar o modelo de acordo com o que há disponível;
  - . dados incompletos e “contaminados”;
  - . muito dados disponíveis (!)
- ” sensibilidade das saídas às entradas;
- ” correspondência do modelo à qualidade dos dados;
- ” custo da coleta — não deixar de incluir no projeto;
- ” captura de variabilidade nos dados — validade do modelo;
- ” lembre-se: lixo na entrada, lixo na saída!

# Utilizando Dados Coletados: Alternativas e Considerações

- “ uso direto dos dados:
  - . utilizar nas entradas dados reais (tempos entre chegadas, de serviço, tipos de entidades, etc.);
  - . todos os valores serão “legais” e realísticos;
  - . simulação fica restrita ao que foi observado;
  - . pode não haver dados disponíveis para simulações longas e replicações;
  - . computacionalmente lento (leitura de arquivos em disco);
- “ ou ajuste de uma distribuição aos dados:
  - . “extrair” ou “gerar” observações sintéticas das distribuições, para alimentar as entradas do modelo;
  - . é o que tem sido feito até este ponto;
  - . é possível ir além dos dados observados (bom e ruim, ao mesmo tempo);
  - . pode não ser possível obter um bom ajuste — validade?

# Ajustando Distribuições via Analisador de Entradas (input analyzer) do Arena

„ suposições:

- . disponibilidade de dados amostrais: conjunto de dados do sistema real independentes e identicamente distribuídos;
- . desejado selecionar ou ajustar uma distribuição de probabilidades para uso na geração de entradas do modelo de simulação;

„ analisador de entradas (input analyzer):

- . aplicativo separado, também acessível via menu “Tools” no Arena;
- . ajusta distribuições, fornece expressões matemáticas válidas para geração de amostras, copiáveis diretamente no modelo de simulação.

# Ajustando Distribuições via Analisador de Entradas (input analyzer) do Arena

- ” ajuste = decisão sobre a forma da distribuição (exponencial, gama, empírica, etc.) e estimativa dos seus parâmetros:
  - . vários métodos disponíveis (máxima verossimilhança, momentos, mínimos quadrados, ...);
  - . teste de aderência (goodness-of-fit ) via teste de hipóteses:
    - ”  $H_0$ : a distribuição representa adequadamente os dados;
    - ” obter valor  $p$  para o teste (baixo = ajuste ruim);
- ” ajuste de distribuição “teórica” *versus* empírica;
- ” distribuição contínua *versus* discreta;
- ” “melhor” ajuste entre várias distribuições.

# Arquivo de Dados para o Analisador de Entradas (input analyzer)

- ” criar arquivo de dados (editor, processador, planilha, ...):
  - . necessariamente em ASCII (salvar como texto ou exportar);
  - . valores separados por espaços brancos (brancos, tabulação, ou nova linha);
  - . caso contrário, formato livre;
- ” abrir dados no analisador de entradas:
  - . menu *File/New* ou 
  - . menu *File/Data File/Use Existing ...* ou 
  - . traças histograma, análise exploratória de dados;
  - . para visualizar o arquivo de dados: menu *Window/Input Data*;
- ” gerar dados fictícios para “brincar”:
  - . menu *File/Data File/Generate New ...*

# Menu Ajuste (fit)

- “ ajusta distribuição e executa testes de aderência;
- “ ajusta uma distribuição específica:
  - grafa densidade sobre o histograma, para “teste” visual;
  - fornece a expressão exata para “copy” e “paste” no modelo de simulação;
  - pode incluir um “offset” dependendo da distribuição;
  - fornece resultados de testes de aderência (goodness-of-fit):
    - “ testes chi-quadrado e Kolmogorov-Smirnov;
    - “ parte mais importante: **valor p**:
      - probabilidade de obter dados mais inconsistentes com a distribuição ajustada do que os atuais, se a distribuição ajustada é realmente a verdadeira;
      - “baixo”  $p$  ( $< 0.05$ ): ajuste ruim (tente novamente, ou desista!).

# Menu Ajuste

- ” ajustar todas as distribuições teóricas de uma só vez:
  - . menu *Fit/Fit All* ou 
  - . retorna a distribuição com menor erro quadrático;
    - ” lead to different “best” distribution
  - . ainda assim, pode resultar em um ajuste ruim (checkar valor p); 
  - . para visualizar “rank” de todas as distribuições, menu *Window/Fit All Summary* ou
- ” “ajusta” distribuições empíricas (contínuas e discretas):  
*Fit/Empirical*
  - . distribuições empíricas podem ser empregadas, quando as “teóricas” não se ajustam bem.

# Independencia da Amostra

Uma hipótese importante feita por muitas técnicas estatísticas é que as observações  $X_1, X_2, \dots, X_n$  sejam uma amostra independente de uma distribuição subjacente. Por exemplo, o teste de Chi-quadrado supõe independencia dos dados.

Duas técnicas gráficas podem ser usadas de maneira informal para verificar se os dados  $X_1, X_2, \dots, X_n$  (listados em ordem temporal) são independentes. São eles o gráfico de correlação e o diagrama de espalhamento.

O gráfico de correlação é o gráfico da correlação amostral entre observações que são afastadas de j observações no tempo. São definidas por,

$$\rho_j = \frac{C_j}{S^2(n)}$$

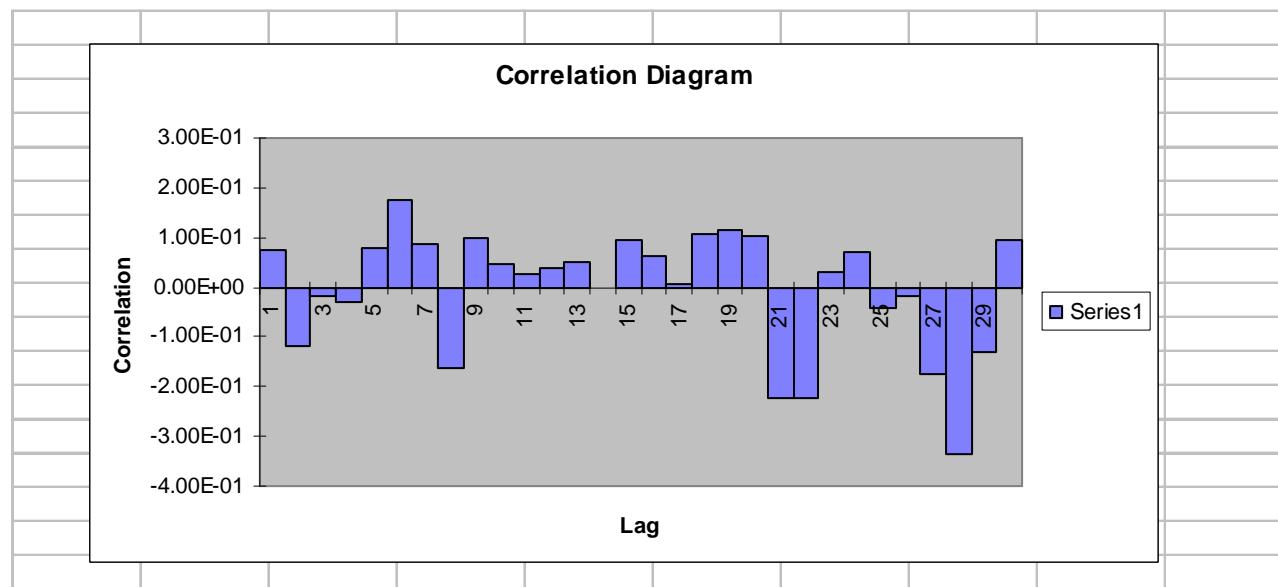
$$C_j = \frac{\sum_{i=1}^{n-j} [X_i - \bar{X}(n)][X_{i+j} - \bar{X}(n)]}{n-j}$$

$$\bar{X}(n) = \frac{\sum_{i=1}^n X_i}{n}$$

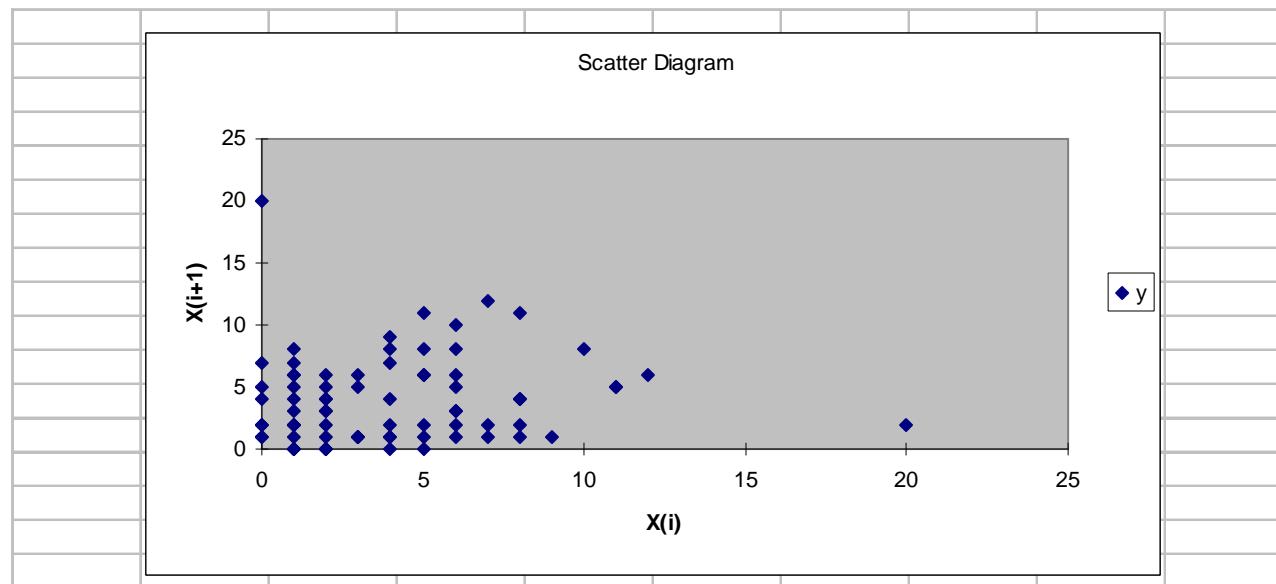
$$S^2(n) = \frac{\sum_{i=1}^n [X_i - \bar{X}(n)]^2}{n-1}$$

## Dados de volume de demanda de um produto

2	7	1	3	6	1	3
2	0	1	5	11	5	3
2	8	1	7	4	8	4
4	0	2	20	0	2	5
1	6	12	7	0	5	11
8	6	2	0	4	2	4
8	10	6	6	5	2	6
3	6	5	0	1	3	1
0	2	1	8	5	6	1
0	1	9	4	1	4	2
2	1	1	2	1	4	



O diagrama de espalhamento é um gráfico de pares de pontos  $(X_i, X_{i+1})$  para  $i=1,2,\dots,n-1$ . Por simplicidade supor que os  $X_i$ s são não negativos. Se os  $X_i$ s forem independentes, os pontos  $(X_i, X_{i+1})$  devem estar espalhados aleatoriamente no primeiro quadrante. Se as observações tiverem correlação positiva, os pontos irão se alinhar na direção de uma reta com coeficiente angular positivo. Se os pontos tiverem correlação negativa, eles irão se alinhar na direção de uma reta com coeficiente angular negativo.



# ARENA INPUT ANALYZER

O Input Analyzer é uma ferramenta que vem com o ARENA. É feita para ajustar distribuições aos dados observados, fornecer estimativas de seus parâmetros, e medir a qualidade deste ajuste.

Podemos resumir em quatro os passos para a utilização do Input Analyzer:

1. Criação de um arquivo texto contendo os valores dos dados.
2. Realize o ajuste de uma ou mais distribuições ao conjunto de dados.
3. Escolha uma função de distribuição.
4. Copiar a expressão da distribuição para o seu modelo ARENA.

Criar o arquivo de dados ( editor, word, excellõ ..)

Deve ser arquivo plain ASCII text ( salve como texto ou exporte)

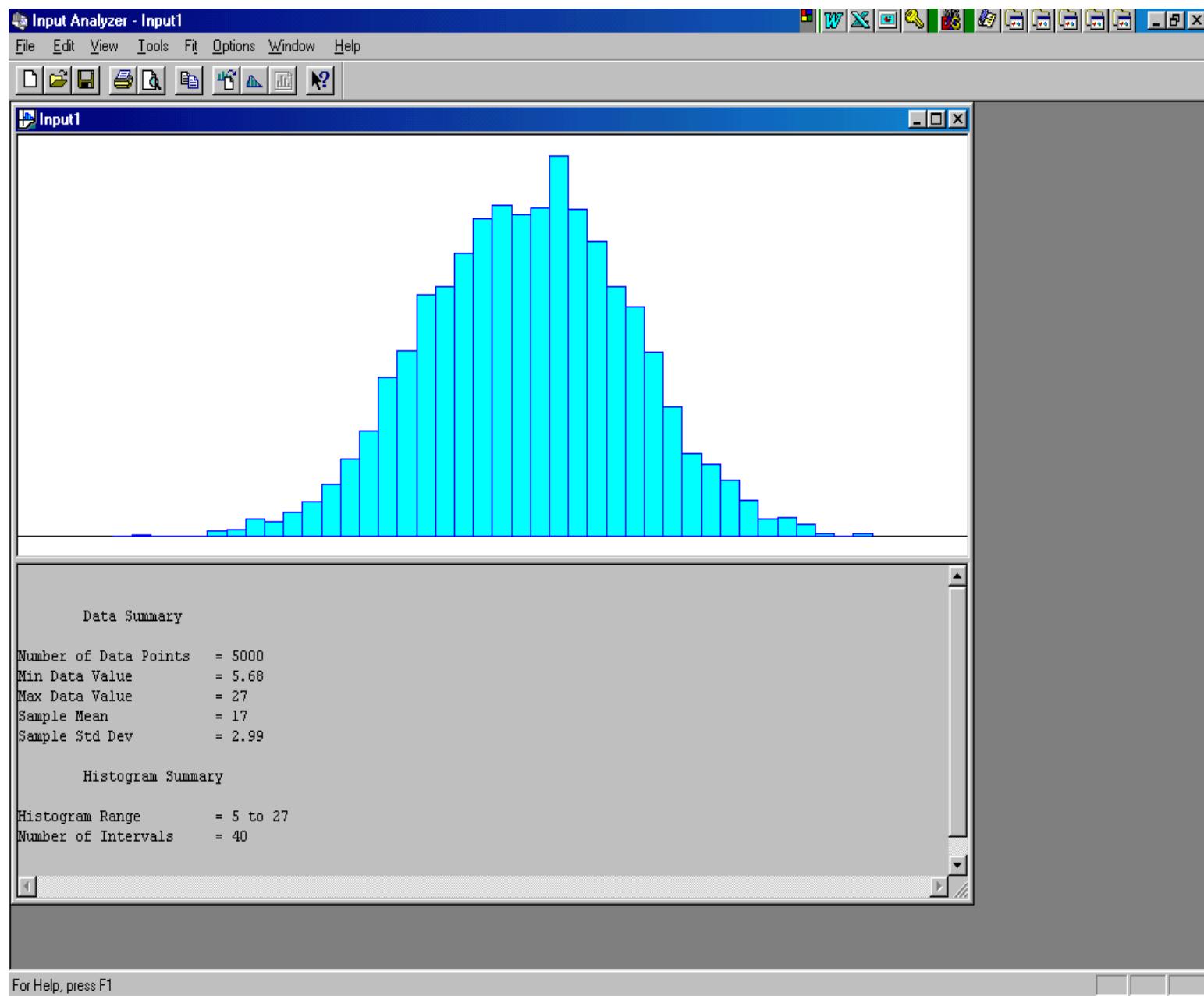
Valores de dados devem ser separados por espaços em brancoData (blanks, tabs, linefeeds)

A extensão padrão para arquivos de dados a serem usados pelo Input Analyzer é **.dft**.

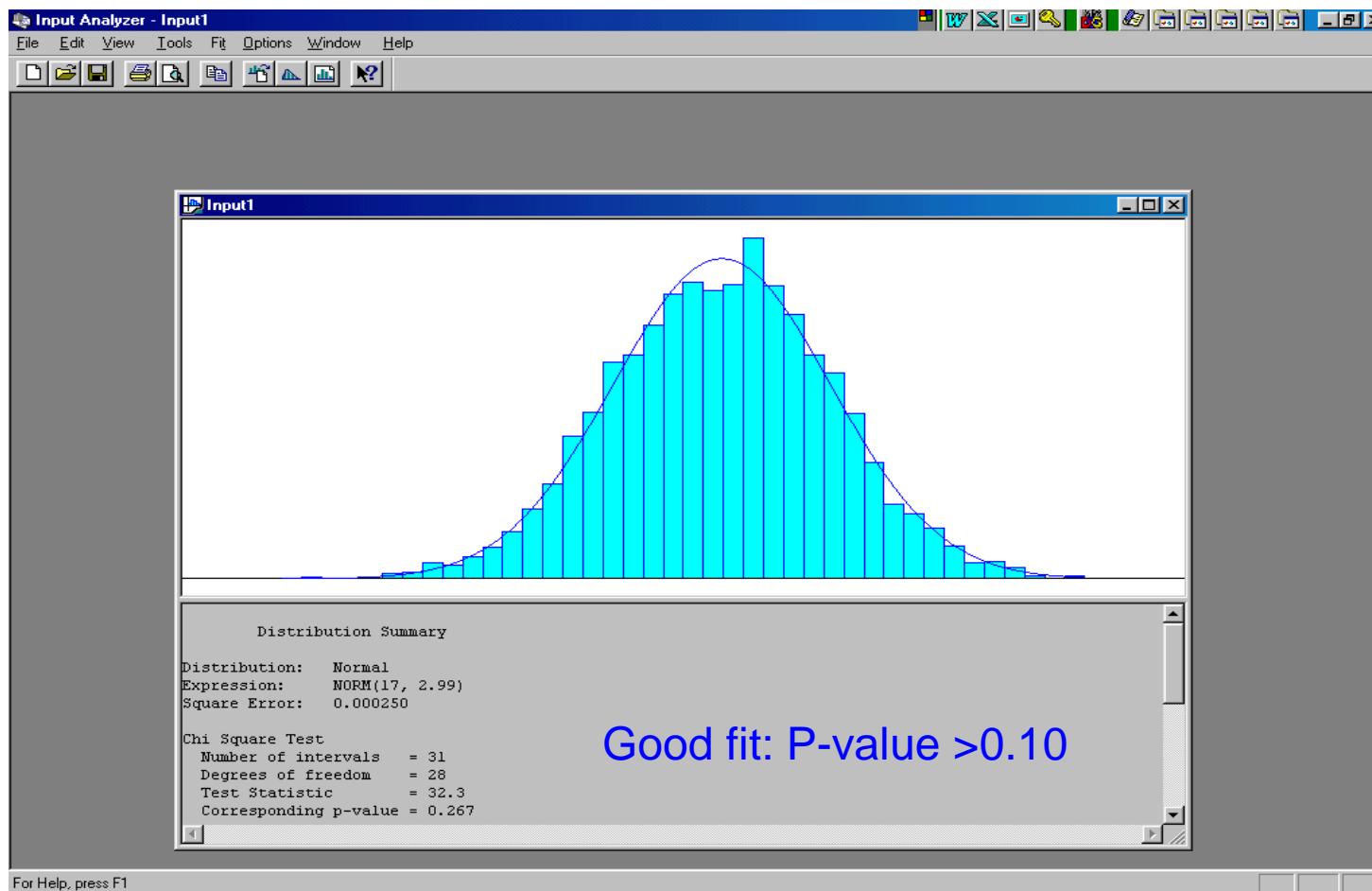
Para carregar o arquivo de dados no Input Analyzer, primeiro abra-o usando **%File/New+**

E então attach o arquivo de dados usando **%File/Data File/Using Existing+**

Pode-se gerar dados falsos para praticar usando  
**File/Data File/Generate New** menu



Para ajustar uma distribuição aos dados, use %Fit/Fit All+.



Para copiar a expressão da função de distribuição para um modelo ARENA, click em %Edit/Copy Expression+e então para o campo de dados do modelo, click %botão esquerdo do mouse+e escolha %Paste+.

# Exercício usando o Input Analyser

Considere um sistema em que navios chegam a um porto para carregar algum produto. Abaixo estão anotados os valores de intervalos entre chegadas (em horas) para 20 navios:

Navio 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20

interv. 10 02 13 07 02 08 08 08 10 09 01 14 14 01 10 09 09 09 08 14

As durações da carga (em horas) de cada navio são as seguintes:

Navio 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20

Duraç. 05 05 03 03 06 07 06 08 02 05 08 08 08 03 04 03 03 04 05 05

Utilizar o input analyser para conhecer as distribuições de entrada e atendimento.

Verificar os seguintes valores para esta distribuição:

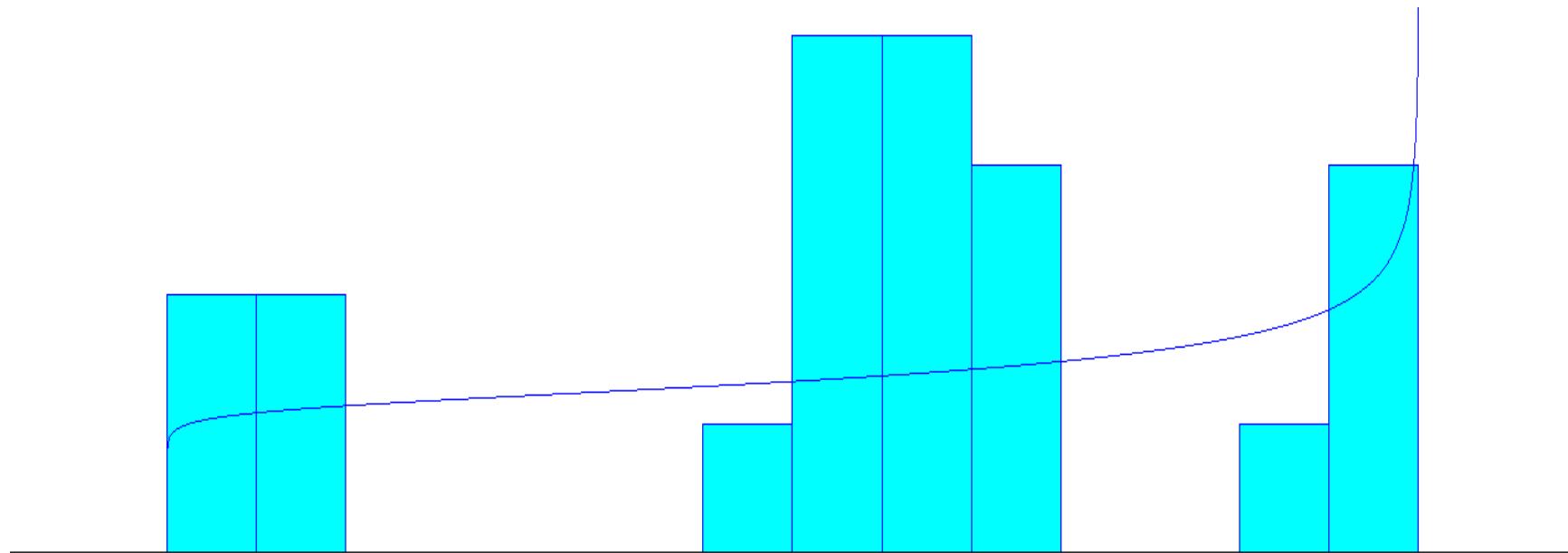
- . O intervalo médio entre chegadas
- . Duração média da carga
- . calcule o tamanho médio da fila
- . calcule o tempo médio de espera na fila

Simular dois cenários:

- . comprimento da corrida 166 horas
- . Comprimento da corrida 10.000 horas

Realizar 10 corridas e verificar a variação das estimativas.

# Ajuste dos dados de chegada



## Distribution Summary

Distribution: Beta  
Expression:  $0.5 + 14 * \text{BETA}(1.05, 0.836)$   
Square Error: 0.074985

## Chi Square Test

Number of intervals = 3  
Degrees of freedom = 0  
Test Statistic = 4.24  
Corresponding p-value < 0.005

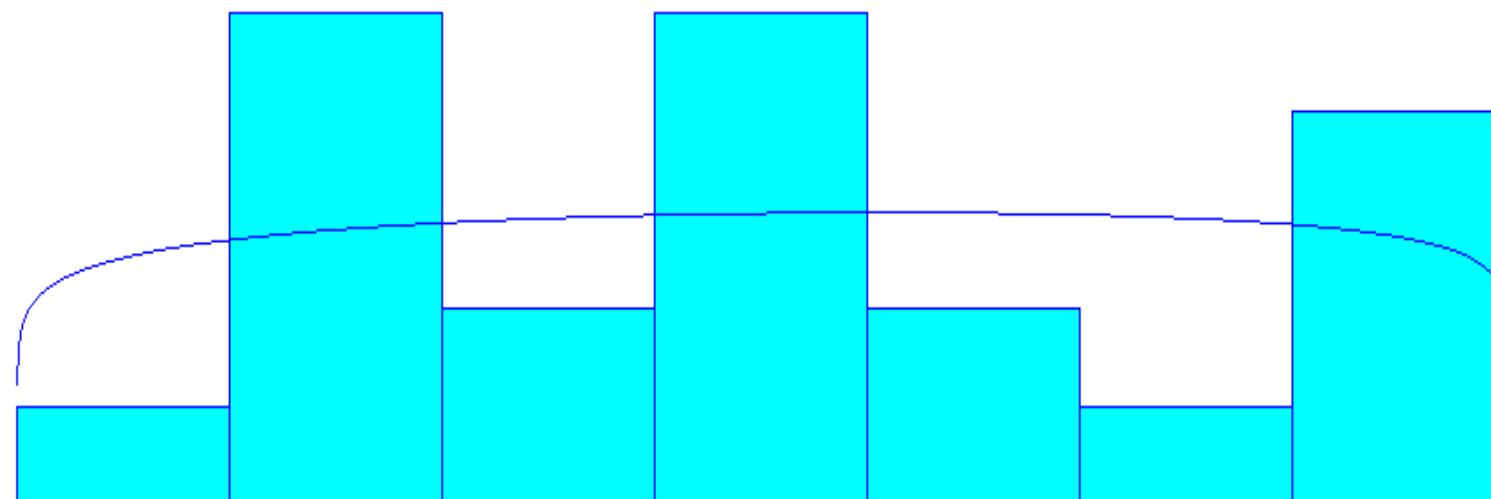
## Data Summary

Number of Data Points = 20  
Min Data Value = 1  
Max Data Value = 14  
Sample Mean = 8.3  
Sample Std Dev = 4.09

## Histogram Summary

Histogram Range = 0.5 to 14.5  
Number of Intervals = 14

# Ajuste do Atendimento



05 05 03 03 06 07 06 08 02 05 08 08 08 03 04 03 03 04 05 05

## Distribution Summary

Distribution: Beta  
Expression:  $1.5 + 7 * \text{BETA}(1.11, 1.08)$   
Square Error: 0.045733

## Chi Square Test

Number of intervals = 3  
Degrees of freedom = 0  
Test Statistic = 2.02  
Corresponding p-value < 0.005

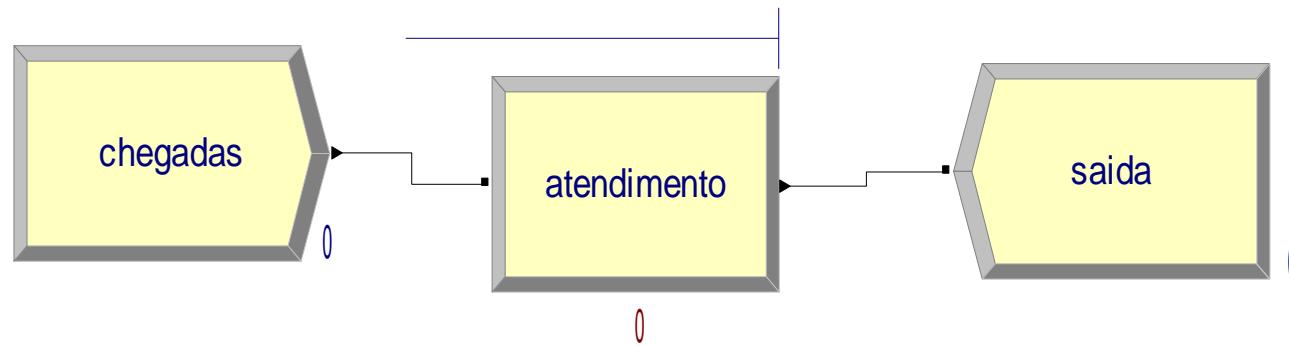
## Data Summary

Number of Data Points = 20  
Min Data Value = 2  
Max Data Value = 8  
Sample Mean = 5.05  
Sample Std Dev = 1.96

## Histogram Summary

Histogram Range = 1.5 to 8.5  
Number of Intervals = 7

# Modelo Porto



# Sem Dados?

- ” ocorre com freqüência maior que a desejada;
- ” não há uma boa solução; algumas opções (ruins):
  - . consulte “experts”
    - ” min, max: uniforme;
    - ” média, erro % error ou erro absoluto: uniforme;
    - ” min, moda, max: triangular;
      - . moda pode ser diferente de média — permite assimetria;
  - . tempos entre chegadas — independentes e estacionário:
    - ” exponencial — ainda assim há necessidade de um valor médio;
  - . número de eventos aleatórios por intervalo: Poisson
  - . soma de partes independentes: normal
  - . produto de partes independentes: lognormal

# Processos de Chegada Não-Estacionários

- ” eventos externos (chegas) cuja taxa varia ao longo do tempo:
  - . horário de almoço em restaurantes;
  - . tráfego na hora do “rush”;
  - . centros de atendimento a clientes;
  - . demanda sazonal de produtos;
- ” pode ser crítico para a validade do modelo a modelagem de processos não-estacionários:
  - . ignorar picos e “vales” pode mascarar comportamentos importantes;
  - . horas de “rush” podem ser perdidas;
- ” bom modelo: *processo de Poisson não-estacionário*.

# Dados de Entrada Multivariados e Correlacionados

- ” usualmente, assume-se que as observações ao longo da simulação são independentes (embora possivelmente de diferentes distribuições);
- ” algumas vezes, isto não é verdade:
  - . uma parte “difícil” requer um tempo de processamento longo na área de preparação e selagem;
  - . isto é correlação positiva;
- ” ignorar tais relações invalidaria o modelo;
- ” foge ao escopo tratar tais problemas.

# AULA 8

” Construção de Modelos Aplicados a Industria.

# EXERCÍCIOS

1) MODELO DE UMA ETAPA DO PROCESSO DE PRODUÇÃO DE LENTES EM UM LABORATÓRIO ÓPTICO.

São fornecidas todas as informações relevantes para a construção do modelo.



## PROCESSO DE PRODUÇÃO



No Laboratório Rigor, o bloco passa por um rigoroso processo de avaliação antes de sua industrialização. Avaliamos sua estrutura física e especificação técnica fornecida pelo fabricante (base e adição) em busca de alguma imperfeição. Chamamos este processo de controle de qualidade inicial.



A próxima etapa é o setor de cálculo. Este setor é o cérebro da produção, pois o profissional que nele trabalha determina como a lente deve ficar depois de surfaçada. Neste setor são feitos os cálculos matemáticos de acordo com o tamanho, tipo e formato da armação para determinação da espessura e curvatura interna da lente, dados necessários para formação da dioptria. Dispomos de uma ferramenta tecnológica poderosa: um software que auxilia neste processo de cálculo de espessura e também na transmissão de dados para a blocadora e para o gerador de curvas . ambos computadorizados . pois este software é interligado com estes equipamentos.



A etapa seguinte chama-se blocagem. Utilizamos uma máquina chamada Blocadora, equipamento computadorizado automático, onde acontece a fixação do bloco e posicionamento do eixo, em um suporte de alumínio necessário para o processo de industrialização.



Feita esta fixação, o bloco é encaminhado para o Gerador de Curvas, equipamento computadorizado que tem a finalidade de gerar a curvatura interna, reduzir o diâmetro e espessura de acordo com os dados gerados no setor de cálculo. Somente nesta etapa acontece a transformação do bloco em lente propriamente dita, pois até então, o bloco permanece conforme veio do fabricante. Em seguida, a lente vai para o setor de Polimento e sofre um aperfeiçoamento e polimento da superfície interna em máquinas cilíndricas. Neste processo, a lente adquire o seu brilho característico.



Depois de polidas, as lentes são descoladas do suporte de alumínio e então levadas ao Setor de Conferência. Lá é utilizado um aparelho chamado Lensômetro Computadorizado, que tem por objetivo conferir a dioptria e o eixo que foi determinado no pedido. Neste setor também conferimos a espessura central e das bordas das lentes.



Após a conferência técnica, as lentes são encaminhadas para o setor de Controle de Qualidade final onde acontece a inspeção de sua estrutura física em busca de qualquer imperfeição da parte interna e externa.

# Simulação do Processo de Surfaçagem

## Etapa 1: Estoque

A PREENCHER PELO CLIENTE PARA USO DA

NOME DO USUÁRIO:																
CLIENTE: _____ FONE: _____																
ENDERECO: _____ N°: _____ CIDADE: _____ U.F.: _____																
LADO	ESF.	CIL.	EIXO	D.N.P.	ADIÇÃO	DIAM.	ALTURA		ASSINALE A INFORMAÇÃO CORRESPONDENTE							
							PEL.	C.O.	TIPO DE LENTE	V.SIMPLES	BIFOCAL	PROGRESSIVA	ASFÉRICA			
O.D.									MATERIAL	CRISTAL	RESINA	TRANS.	POLY	INCOL	COLOR	FOTO
O.E.									ÍNDICE	1.5	1.523	1.560	1.6	1.7	1.8	1.9

**ARMAÇÃO**

MARCA: _____	MODELO: _____	TAMANHO: _____	CÓD. REF.: _____			
COLORAÇÃO =		ROSA <input type="checkbox"/>	VERDE <input type="checkbox"/>	CINZA <input type="checkbox"/>	MARROM <input type="checkbox"/>	OUTROS: _____

**CENTRO DA PONTE**



DESENHO O MODELO DA LENTE (OLHO DIREITO)

ARO TOTAL  FIO DE NYLON  PARAFUSADA

**TRABALHOS A SEREM EXECUTADOS**

- SURFAÇAGEM
- ENDURECIMENTO (MINERAL)
  - FACE INTERNA
  - AMBOS OS LADOS
- ANTI-REFLEXO
  - COM ANTI-RISCO
  - DESTRATAMENTO

**MATERIAL:**  CLIENTE  OP. S  TERCEIROS

**FABRICANTE / MARCA DA LENTE:** \_\_\_\_\_

- ASSINALE O DIÂMETRO DESEJADO  
- EM LENTES POSITIVAS ACIMA DE 1,00 D. INDIQUE A ESPESSURA DE BORDA DESEJADA

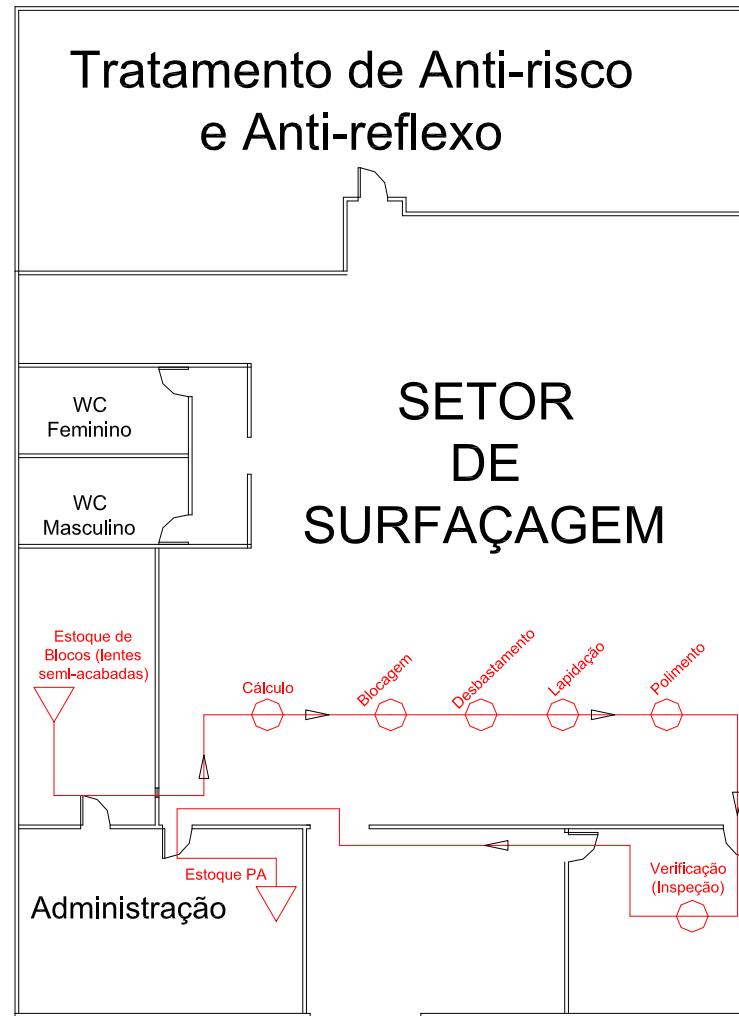
DIÂMETRO	<input type="checkbox"/> Ø 60	mm
	<input type="checkbox"/> Ø 65	mm
	<input type="checkbox"/> Ø 70	mm
	<input type="checkbox"/> Ø 75	mm

**OBSERVAÇÕES:** \_\_\_\_\_

ENTREGUE EM	ASSINATURA DO CLIENTE	PROMETIDO PARA	ASSINATURA DA	RECEBI EM	ASSINATURA DO CLIENTE
_____/_____/_____	_____/_____/_____	_____/_____/_____	_____/_____/_____	_____/_____/_____	_____/_____/_____

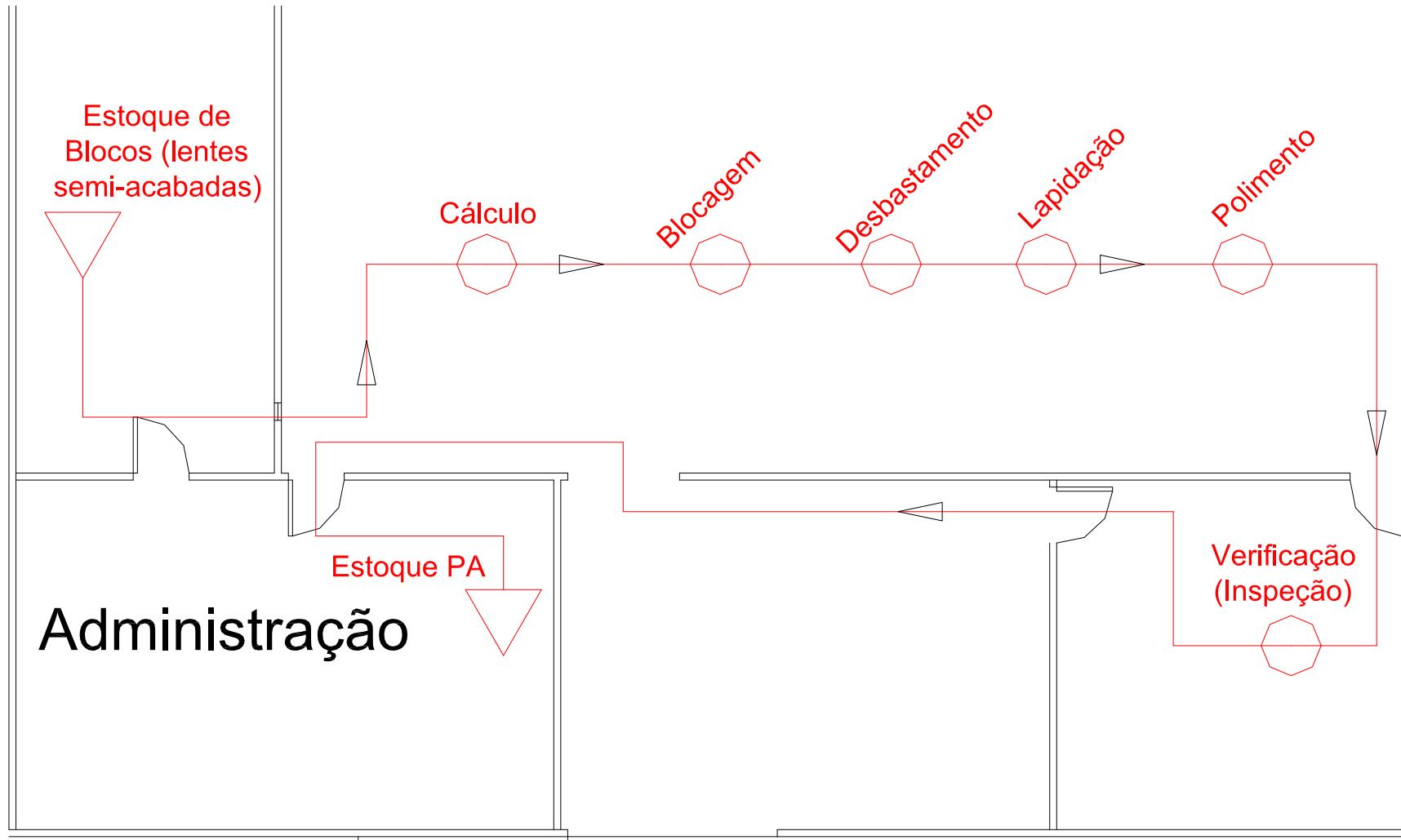
# Simulação do Processo de Surfaçagem

## Mapofluxograma



# Simulação do Processo de Surfaçagem

## Mapofluxograma



## Formulação do Problema

- O intervalo de tempo entre as chegadas de pedidos foi um valor estocástico;
- Cada pedido que chegava ao processo deveria passar por sete etapas;
- O tempo gasto por um pedido em cada etapa do processo foi um valor estocástico.

## Formulação do Problema

### Questões a Serem Respondidas:

- Tempo médio de processamento;
- Tempo médio em fila;
- Tamanho médio de fila.

# Formulação do Problema

## Configuração do Sistema Modelado:

- Entidades;
- Produto Final;
- Capacidade da Fila;
- Canal de Atendimento;
- Número de Servidores;
- Capacidade de Servidores;
- Estratégias de Admissão;
- Chegada dos Clientes;
- Prioridade da Fila;
- Tamanho da População;
- Taxa de Chegada;
- Taxa de Atendimento;
- Tempo entre Chegadas;
- Tempo de Atendimento.

## Formulação do Problema

### Avaliadores de Desempenho:

- Tempo médio de processamento
- Tempo médio em fila
- Tamanho médio de fila
- Números de elementos atendidos

# Formulação do Problema

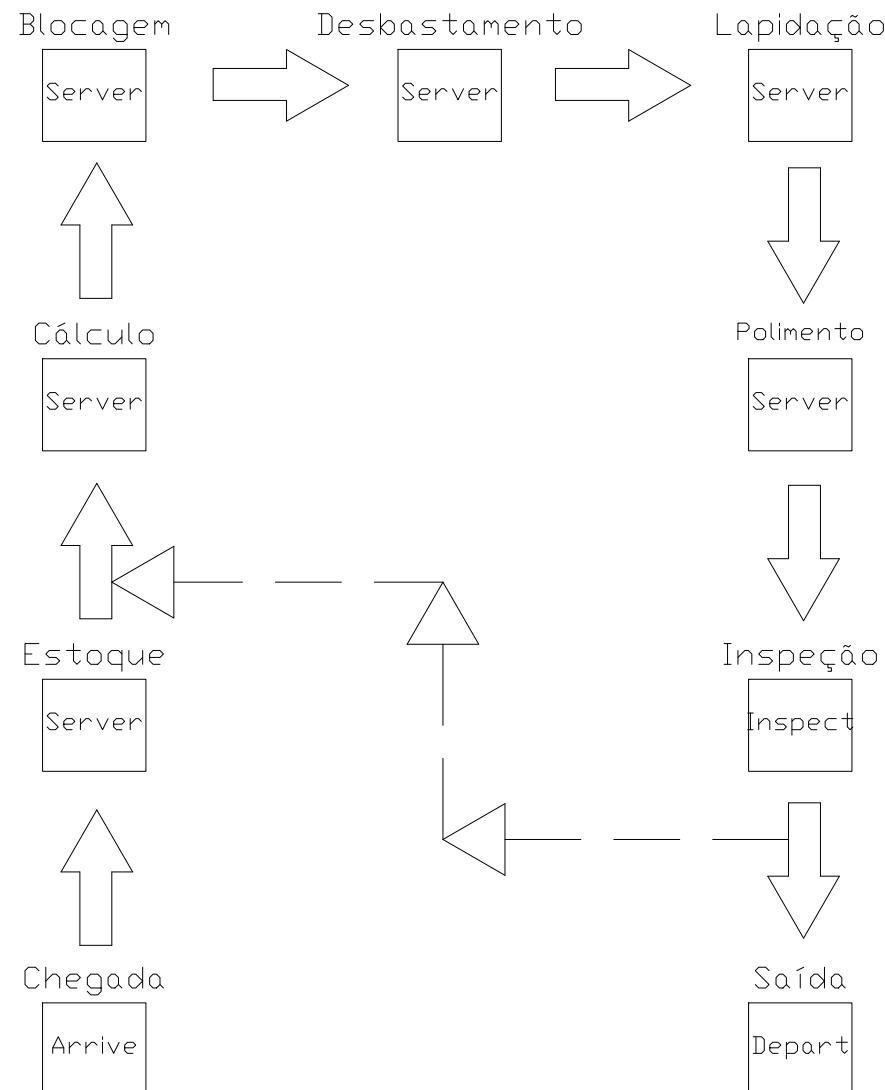
## Etapas do Processo de Simulação:

- Coleta de Dados e Definição do Modelo
- Validação do Modelo Conceitual
- Construção do Modelo de Simulação e Verificação
- Realização de Execuções Piloto
- Validação do Modelo Programado
- Projeto dos Experimentos
- Execuções das Simulações
- Análise de Resultados

## Resultados do *Input Analyser*:

Chegada	$289 + \text{gama}(7.03, 1.18)$
Estoque	$80.5 + 19 \times \text{beta}(2.04, 1.47)$
Cálculo	$13.5 + 3 \times \text{beta}(1.76, 1.97)$
Blocagem	$140 + 6 \times \text{beta}(1.2, 1.1)$
Desbastamento	$171 + 19 \times \text{beta}(1.72, 1.33)$
Lapidação	Triangular(65, 75, 85)
Polimento	Triangular(290, 300, 310)
Inspeção	$61.5 + 38 \times \text{beta}(2.42, 1.49)$

## Diagrama de Blocos:



**2) Em um setor de uma fábrica, o produto que está sendo fabricado chega para receber componentes adicionais através de um instalador. Após instalados os componentes pelo instalador, o produto é inspecionado por um profissional qualificado. Os produtos que passam na inspeção vão para outro setor da fábrica e os que são rejeitados (20%) vão para um setor de reparo, após o que também vão para o outro setor. Os dados do cenário atual são os seguintes:**

- ” A cada EXPO(40) minutos chega um novo produto ao setor (exponencial negativa);
- ” O instalador gasta UNIF(25,30) para instalar os componentes;
- ” O inspetor gasta UNIF(5,10) para inspecionar o trabalho realizado;
- ” O reparador gasta UNIF(5,15) para efetuar os reparos necessários;
- ” Todos os tempos de deslocamento são iguais a 1 minuto.

**É previsto um aumento das vendas e o novo intervalo entre chegadas será 20 minutos.**

**Pede-se:**

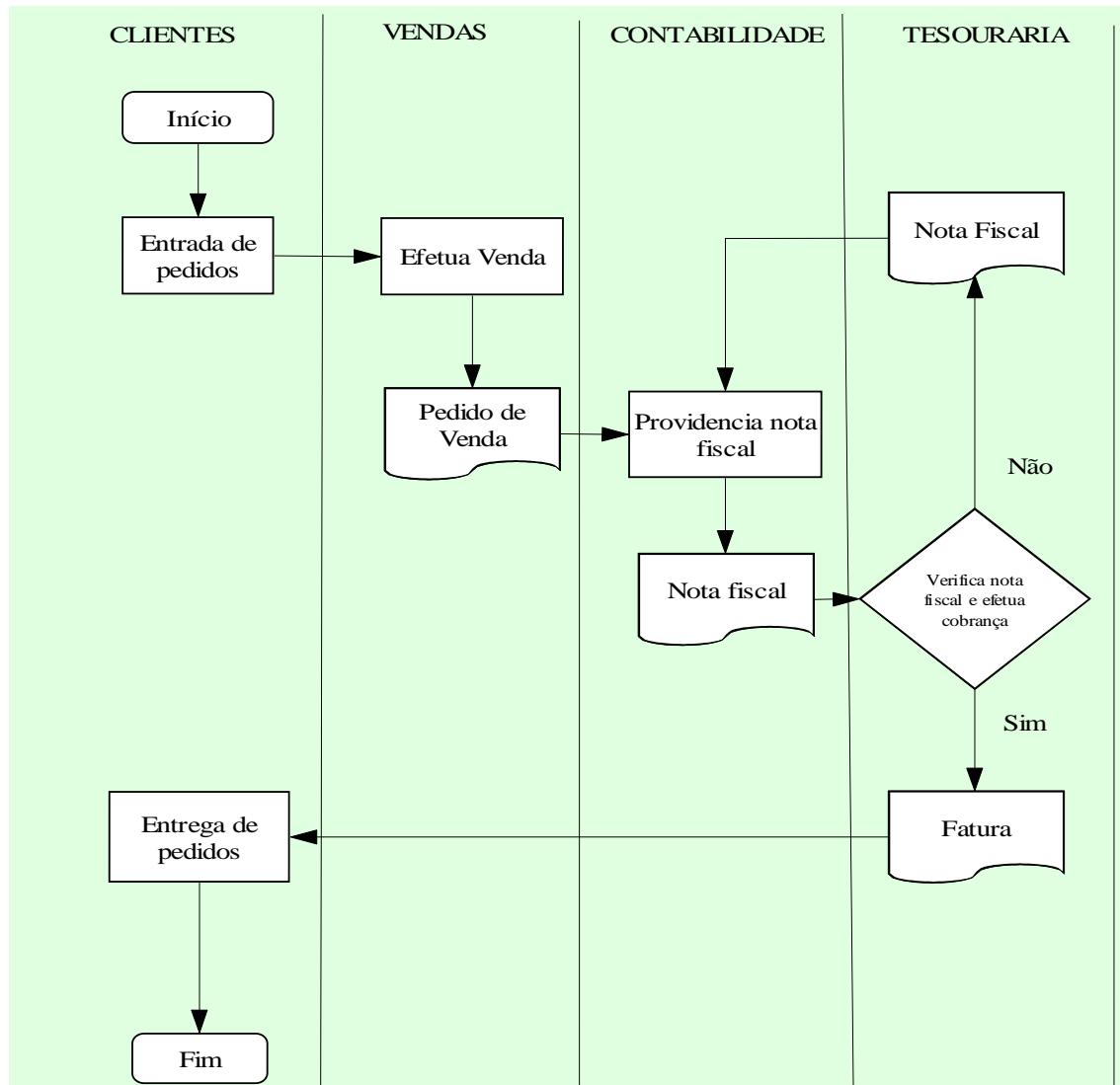
- “ Redimensione, para o cenário futuro, a quantidade de funcionários de cada setor de modo que a fila seja menor que 4;**
- “ Produção obtida em 1000 minutos e o tempo médio de produção de uma unidade do produto.**

# AULA 9

- ” Construção de um Modelo Aplicado a um Call Center.
- ” Noções sobre Análise dos Resultados da Simulação

# Um modelo aplicado

8) Uma loja, líder no mercado de venda de roupas de uma grife, decide contratar os serviços de uma consultora para poder melhorar seus processos de atendimento de pedidos via telefone, tendo em vista as inúmeras reclamações recebidas dos clientes. O processo de atendimento simplificado se encontra ilustrado pelo fluxograma do atendimento e o *lay-out* da loja, indicados nas figuras a seguir:



**Fluxograma do atendimento de pedidos**

Devolve Nota Fiscal = 20%

O processo de atendimento simplificado de pedidos é descrito a seguir:

**Cliente** (Chegada de pedidos: Expo(15)):

Faz pedido do produto (deslocamento Chegada e Vendas = 2 segundos).

**Vendas** (Tempo de venda: Tria(8,10,12)):

Efetua a venda de roupas;

Emite pedido de vendas e envia à Contabilidade (Tempo de deslocamento entre Vendas e Contabilidade = 2 segundos).

**Contabilidade** (Tempo de preenchimento da Nota Fiscal: Tria(18,22,28)):

Providencia a Nota Fiscal – NF;

Envia a NF para a Tesouraria (Tempo de deslocamento entre Contabilidade e Tesouraria = 2 segundos)..

**Tesouraria:**

Verifica a Nota Fiscal – NF e efetua cobrança (Tempo de inspeção = 2 segundos e Tempo de deslocamento entre Tesouraria e Contabilidade = 2 segundos);

Devolve a NF para a Contabilidade (Índice de rejeição = 20%);

Emite fatura e entrega o pedido de compra para o cliente (Tempo de deslocamento entre Tesouraria e a Entrega = 2 segundos)..

**Cliente:**

Recebe e confere o pedido juntamente com a NF.

Fim

Para colaborar com a análise da equipe, pede-se responder as questões a seguir:

Determine os seguintes valores para a “Contabilidade” e a “Tesouraria”:

Taxa de ocupação;

Tamanho médio da fila;

Tempo médio na fila.

Com a mudança de estação, a loja em estudo pretende atender 100 pedidos em um tempo de 1440 segundos (24 minutos do fluxo de pedidos mais crítico). Ajude ao dono da loja a verificar se o estabelecimento tem condições de atender esta demanda, uma vez que, se não atender satisfatoriamente a demanda prevista, perderá clientela e estará sujeita a uma multa.

Analise as propostas de solução definidas a seguir pela equipe consultora e defina qual é a melhor:

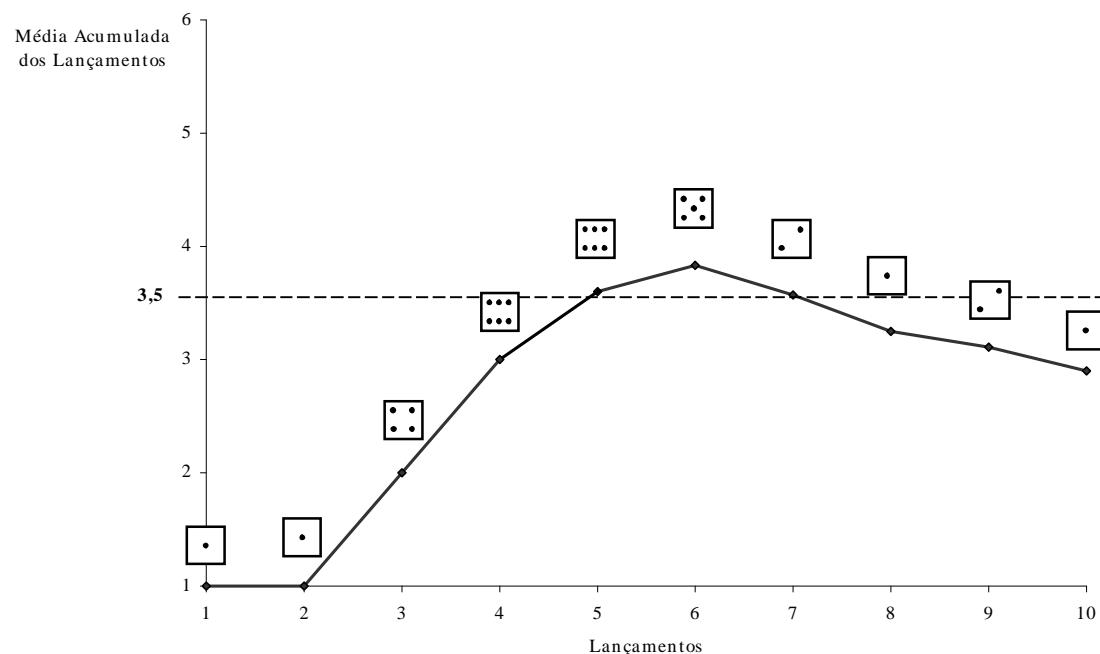
diminuir o tempo do processo de contabilidade. Isto poderia aumentar a taxa de rejeição e seria necessário fazer alguns testes práticos mas, pela simulação, poderíamos testar o impacto da diminuição do tempo de contabilidade, por exemplo, para  $TRIA(8,10.5,12)$  com uma taxa de rejeição aumentada para 30%;

colocar um segundo servidor na estação de contabilidade e enviar as notas fiscais rejeitadas para outra estação de trabalho. Por exemplo, elas seriam enviadas para a estação REFAZER, com um tempo de processamento de  $TRIA(5,7.5,10)$ . A seguir elas voltariam para a estação de tesouraria, com um tempo de deslocamento de 2 segundos.

# Análise dos Resultados da Simulação

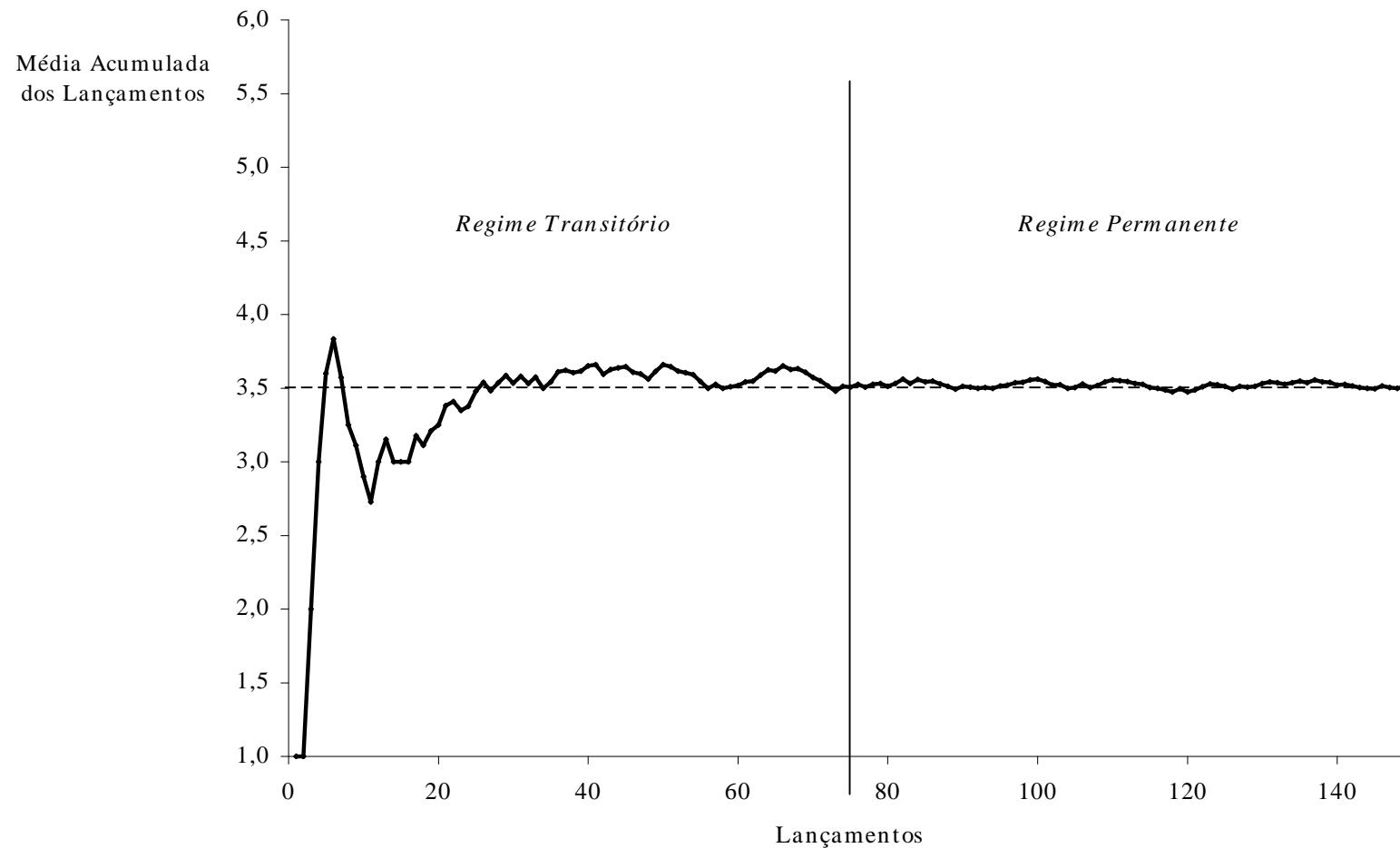
- ” O que é regime transitório e o que é regime permanente;
- ” O que é simulação terminal e o que é simulação em regime;
- ” O que são medidas de desempenho;
- ” O que é replicação e o que é “rodada”;
- ” O que é intervalo de confiança.

# Análise dos Resultados da Simulação



Lançamento	Número Obtido	Média Acumulada
1	1	1/1=1,0
2	1	(1+1)/2=1,0
3	2	(1+1+2)/3=2,0
4	3	(1+1+4)/4=3,0
5	6	3,6
6	5	3,8
7	2	3,6
8	1	3,5
9	2	3,3
10	1	3,1

# Análise dos Resultados da Simulação



# Análise dos Resultados da Simulação

- *Simular por um período muito longo de modo que o número de amostras em regime transitório seja desprezível em relação ao número de amostras em regime (jogar mais vezes o dado);*
- *Eliminar o período transitório através de alguma técnica apropriada;*
- *Iniciar o sistema já em um estado dentro do regime permanente, o que equivale, no exemplo do dado, a considerar a média inicial igual a 3,5.*

# Análise dos Resultados da Simulação

- **NÃO TERMINAL:** a simulação não possui um tempo exato para terminar. Somente há interesse de estudar uma simulação não terminal para o período em que a simulação está em regime permanente (Ex. simulação de uma usina siderúrgica que opera 24 horas por dia, 7 dias por semana).
- **TERMINAL:** a simulação roda por um tempo exato e após este tempo acaba. (Ex. simulação de um que pub abre às 12:00 horas e fecha, pelas leis inglesas, pontualmente às 23:00 horas).

# Análise dos Resultados da Simulação

- Rodada:** o que ocorre quando selecionamos ou iniciamos o comando que executa a simulação no computador. Uma rodada pode envolver várias replicações.
- Replicação:** é uma repetição da simulação do modelo, com a mesma configuração, a mesma duração e com os mesmos parâmetros de entrada, mas com uma semente de geração dos números aleatórios diferente.

Apesar de os dados e dos parâmetros de entrada serem os mesmos, como os números aleatórios gerados são diferentes, cada replicação terá uma saída diferente também.

# Análise dos Resultados da Simulação

*Ex.: fila em um posto bancário*

**Uma rodada:** 6,72 pessoas  
em média na fila

<i>Replicação</i>	<i>Média de Pessoas em Fila</i>
1	6,72
2	2,00
3	0,38
4	1,28
5	0,46
6	0,19
7	0,14
8	1,30
9	0,12
10	2,85
<i>Média de 10 replicações</i>	<i>1,54</i>
<i>Desvio Padrão</i>	<i>2,03</i>

**Podemos CONFIAR nesses resultados?**

# Análise dos Resultados da Simulação

- ” **Intervalo de Confiança:** intervalo de valores que contém a média da população, com uma certa probabilidade (confiança estatística)
- ” **Precisão:** tamanho do intervalo de confiança
- ” **Confiança:** probabilidade de que o intervalo de confiança contenha a média. Valores usuais: 99%, 95% e 90%.

# Análise dos Resultados da Simulação

<i>n</i>	<i>Confiança</i> $100(1-\alpha)\%$	$\alpha$	$t_{n-1, \alpha/2}$	<i>Precisão (metade do intervalo)</i>	<i>Intervalo de Confiança da Média</i> ( $\bar{x}=1,54$ )
				$h = t_{n-1, \alpha/2} \frac{s}{\sqrt{n}}$	
10	99%	0,01	3,25	2,09	$-0,55 \leq \bar{x} \leq 3,63$
10	95%	0,05	2,26	1,45	$0,09 \leq \bar{x} \leq 3,00$
10	90%	0,10	1,83	1,18	$0,37 \leq \bar{x} \leq 2,72$
10	80%	0,20	1,38	0,89	$0,65 \leq \bar{x} \leq 2,43$

**Por que, ao aumentarmos a confiança, a precisão diminui?**

# Análise dos Resultados da Simulação

- ” Método 1: cálculo quando **se conhece** o desvio padrão da população
- ” Método 2: cálculo quando **NÃO se conhece** o desvio padrão da população

# Análise dos Resultados da Simulação

” Método 1: cálculo quando **se conhece** o desvio padrão da população:

$$P(\bar{x} - e_0 \leq \mu \leq \bar{x} + e_0) = 1 - \alpha$$

$$P\left(\bar{x} - z_{\alpha/2} \frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{x} + z_{\alpha/2} \frac{\sigma}{\sqrt{n}}\right) = 1 - \alpha$$

$$IC = \bar{x} \pm z_{\alpha/2} \frac{\sigma}{\sqrt{n}}$$

# Análise dos Resultados da Simulação

” Método 2: cálculo quando **NÃO** se conhece o desvio padrão da população:

$$P(\bar{x} - e_0 \leq \mu \leq \bar{x} + e_0) = 1 - \alpha$$

$$P\left(\bar{x} - t_{n-1, \alpha/2} \frac{s_x}{\sqrt{n}} \leq \mu \leq \bar{x} + t_{n-1, \alpha/2} \frac{s_x}{\sqrt{n}}\right) = 1 - \alpha$$

$$IC = \bar{x} \pm t_{n-1, \alpha/2} \frac{s_x}{\sqrt{n}}$$

Desvio Padrão da  
Amostra

# Análise dos Resultados da Simulação

Para se atingir uma precisão desejada em determinado valor, necessita-se rodar o modelo várias vezes, gerando uma AMOSTRA PILOTO de tamanho  $n$  e com precisão  $h$ . Utilizando-se a expressão a seguir, onde  $h^*$  é a precisão desejada, pode-se estimar o número de replicações necessárias  $n^*$ :

$$n^* = n \left( \frac{h}{h^*} \right)^2$$

# Análise dos Resultados da Simulação

Considere-se que foram realizadas 20 replicações de um modelo de simulação. Para essa amostra piloto, a precisão obtida foi de 0,95 minutos para a média do tempo em fila. Qual o número de replicações necessárias caso necessite-se de uma precisão de 0,5 minutos?

Neste caso,  $n=20$ ,  $h=0,95$  e  $h^*=0,5$ :

$$n = 20 \left( \frac{0,95}{0,5} \right)^2 = 72,2 \simeq 73$$

# Análise dos Resultados da Simulação

7 Etapas:

- ” Estabelecer as medidas de desempenho adequadas;
- ” Escolher a confiança estatística e a precisão com que se pretende trabalhar;
- ” Definir, a partir da observação do sistema real, o tempo de simulação;
- ” Construir a “amostra piloto”;
- ” Determinar o número de replicações necessárias;
- ” Rodar o modelo novamente;
- ” Calcular o novo intervalo de confiança.

# Análise dos Resultados da Simulação

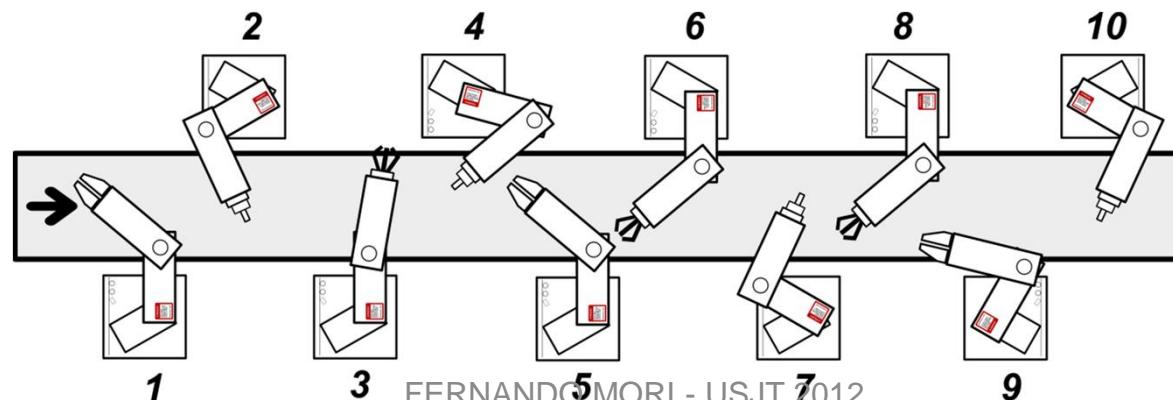
## 3 Técnicas

- ” Começar a simulação em um estado próximo daquele esperado em regime permanente;
- ” Rodar o modelo por um tempo de simulação longo;
- ” Eliminar, dos dados de saída, todos os valores gerados durante o período transitório.

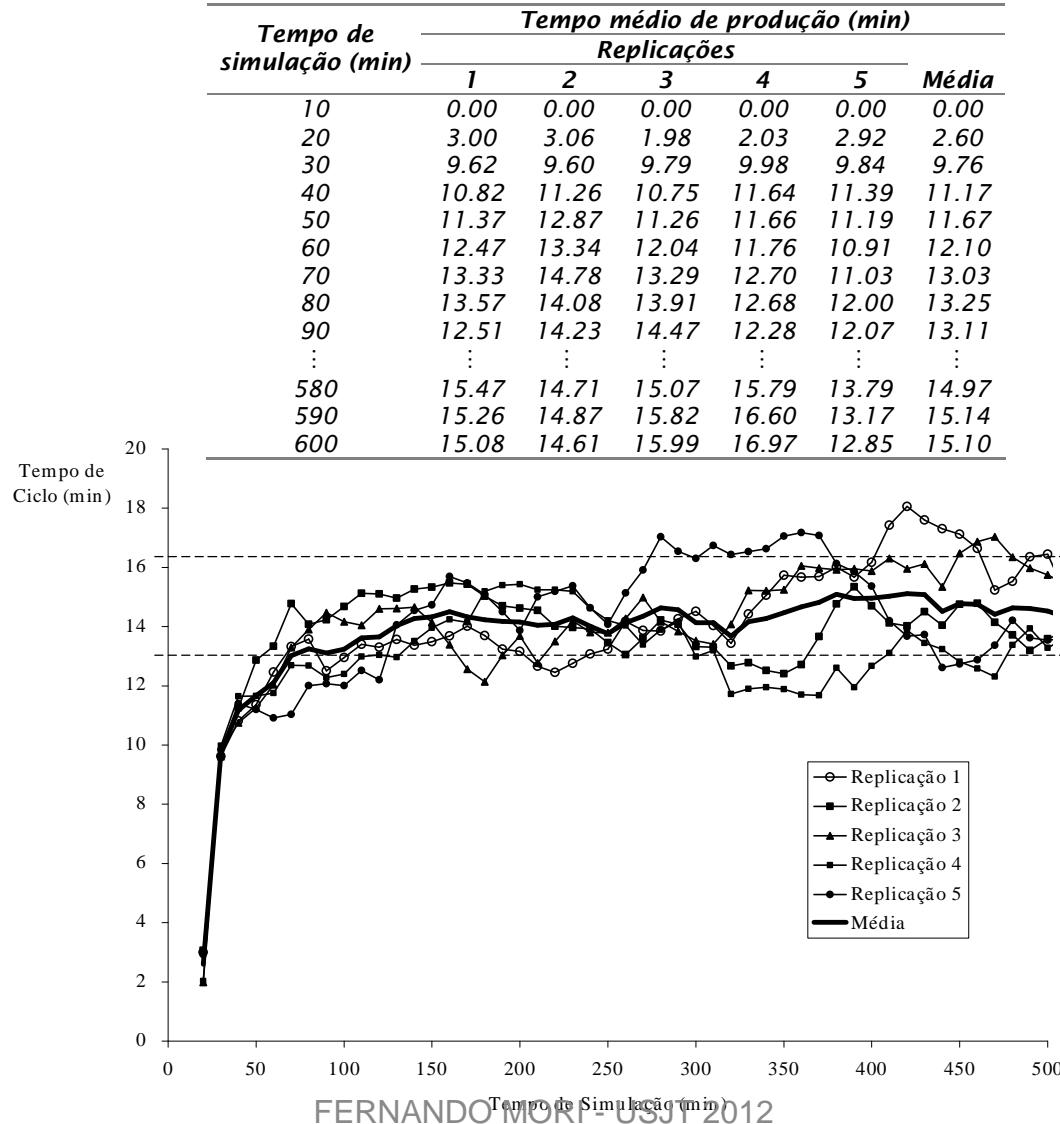
# Análise dos Resultados da Simulação

*Em uma linha de produção de um determinado produto, para que o produto possa ser produzido, uma peça deve passar por 10 operações executadas em máquinas automáticas distintas. Os tempos de operação nas máquinas são todos normalmente distribuídos com média de 0,9 minuto e desvio padrão de 0,3 minuto. As peças chegam à linha em um intervalo constante de tempo igual a 1 minuto. O gerente da linha está preocupado com o tempo total de produção. Nas palavras dele:*

- . Se o produto passa por 10 máquinas que levam 0,9 minuto cada uma, então, era de se esperar que o tempo total de produção fosse de  $0,9 \times 10 = 9$  minutos, em média. Mas, hoje, estamos operando em 15 minutos, cerca de 50% mais lentos!*



# Análise dos Resultados da Simulação



# AULA 10

# EXERCÍCIOS FINAIS

1) Um escritório que entrega placas de automóveis dividiu os seus clientes em categorias com a finalidade de agilizar o trabalho. Os clientes chegam e entram em de três filas baseados na localização de sua residência. Modele estas chegadas como três fluxos independentes usando uma distribuição exponencial com tempo médio entre chegadas de 10 minutos. Cada tipo de cliente fica em uma fila diferente, e para cada fila existe um atendente que recebe um formulário e recebe o pagamento de uma taxa. O tempo de serviço é UNIF (8,10) minutos para todos os tipos de clientes. Após completar este processo, todos os clientes vão para uma fila única, na qual um segundo atendente verifica os formulários e entrega as placas (este atendente irá atender os três tipos de clientes que irão se misturar numa fila única e serão atendidos de acordo com a disciplina FIFO). O tempo de serviço para esta atividade é UNIF (2.66 , 3.33) minutos para todos os tipos de clientes. Desenvolva um modelo e o execute para 5000 minutos observando o tempo médio no sistema para todos os tipos de clientes. Execute o modelo para 5 replicações.

Um consultor recomendou que o escritório não diferencie entre os clientes e que use uma única fila com 3 atendentes que pode em atender um cliente de qualquer tipo. Desenvolva um modelo para este sistema e o execute para 5000 minutos. Compare com os resultados do primeiro modelo e decida qual é o melhor. Faça animações para o modelo.

2) Objetos chegam de acordo com uma distribuição exponencial de tempo entre chegadas com média 1.1 ( todos os tempos em minutos). Ao chegar os objetos são empacotados por um dos 4 empacotadores que estão em uma única fila. O tempo de empacotamento é  $\text{TRIA}( 2.75 , 3.3 , 4.0 )$ . Os pacotes são então separadas por tipo ( 20% internacional e 80% doméstico), e são enviados para serem despachadas. Existe um único atendente para pacotes internacionais e dois para pacotes nacionais com uma única fila chegando aos atendentes domésticos. O atendente internacional gasta um tempo  $\text{TRIA}( 2.3 , 3.3 , 4.8 )$  e os atendentes nacionais gastam  $\text{TRIA}( 1.7 , 2.0 , 2.7 )$ . Este sistema de empacotamento trabalha em turnos de 8 horas, cinco dias por semana. Todos os empacotadores e atendentes que despacham tem intervalos de 15 minutos a cada 2 horas de trabalho, e um intervalo de 30 minutos para o almoço. Execute a simulação para 4 semanas ( 20 dias de trabalho) para determinar o numero médio e máximo de pacotes em cada fila. Faça 5 replicações.

# Referências Bibliográficas

1. Darci Prado, Usando o Arena em Simulação; Série Pesquisa Operacional, Volume 3, INDG, Belo Horizonte, 2004.
2. Maria Cristina Fogliatti e Neli Maria Costa Mattos; Teoria das Filas, Editora Interciencia, Rio de Janeiro, 2007.
3. Darci Prado; Teoria das Filas e da Simulação; Série Pesquisa Operacional, Volume 3, INDG, Belo Horizonte, 2004.
4. Afonso C. Medina e Leonardo Schwif; Modelagem e Simulação de Eventos Discretos, Segunda Edição.