# SEN9110 Simulation Masterclass
## 13. Simulation Languages (1)

Prof.dr.ir Alexander Verbraeck
a.verbraeck@tudelft.nl

Brightspace: SEN9110

# Agenda

- Simulation Language History

  Paper: Richard E. Nance. Simulation Programming Languages: An Abridged History. In: C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldsman (Eds.). Proceedings of the 1995 Winter Simulation Conference. IEEE, 1995. pp. 1307-1313.

  Paper: Ole-Johan Dahl and Kristen Nygaard. SIMULA - An ALGOL-based Simulation Language. Communications of the ACM, Vol 9, No 9, 1966. pp. 671-678.

- Simulation Language (Environment) demonstrations and comparison [1]
  - Arena
  - Enterprise Dynamics
  - ExtendSim
  - Plant Simulation
  - AutoMod

# 1.

# History of Simulation Languages

Paper: Richard E. Nance. Simulation Programming Languages: An Abridged History. In: C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldsman (Eds.). Proceedings of the 1995 Winter Simulation Conference. IEEE, 1995. pp. 1307-1313.

Paper: Ole-Johan Dahl and Kristen Nygaard. SIMULA - An ALGOL-based Simulation Language. Communications of the ACM, Vol 9, No 9, 1966. pp. 671-678.

# Requirements for discrete event simulation languages

- Generation of pseudo random numbers
- Theoretical statistical distributions
- Statistical analysis routines
- Output generation capabilities
- A time flow mechanism

# Discrete Simulation Languages I

- A simulation language is based on a **worldview**. How does one see the world
- Discrete simulation language worldviews interrelate with the concept of "locality" (Weinberg 1971). How does one group behavioral logic?
- Discrete simulation language follow either the process interaction worldview, event scheduling worldview or activity scanning worldview

# Discrete Simulation Languages II

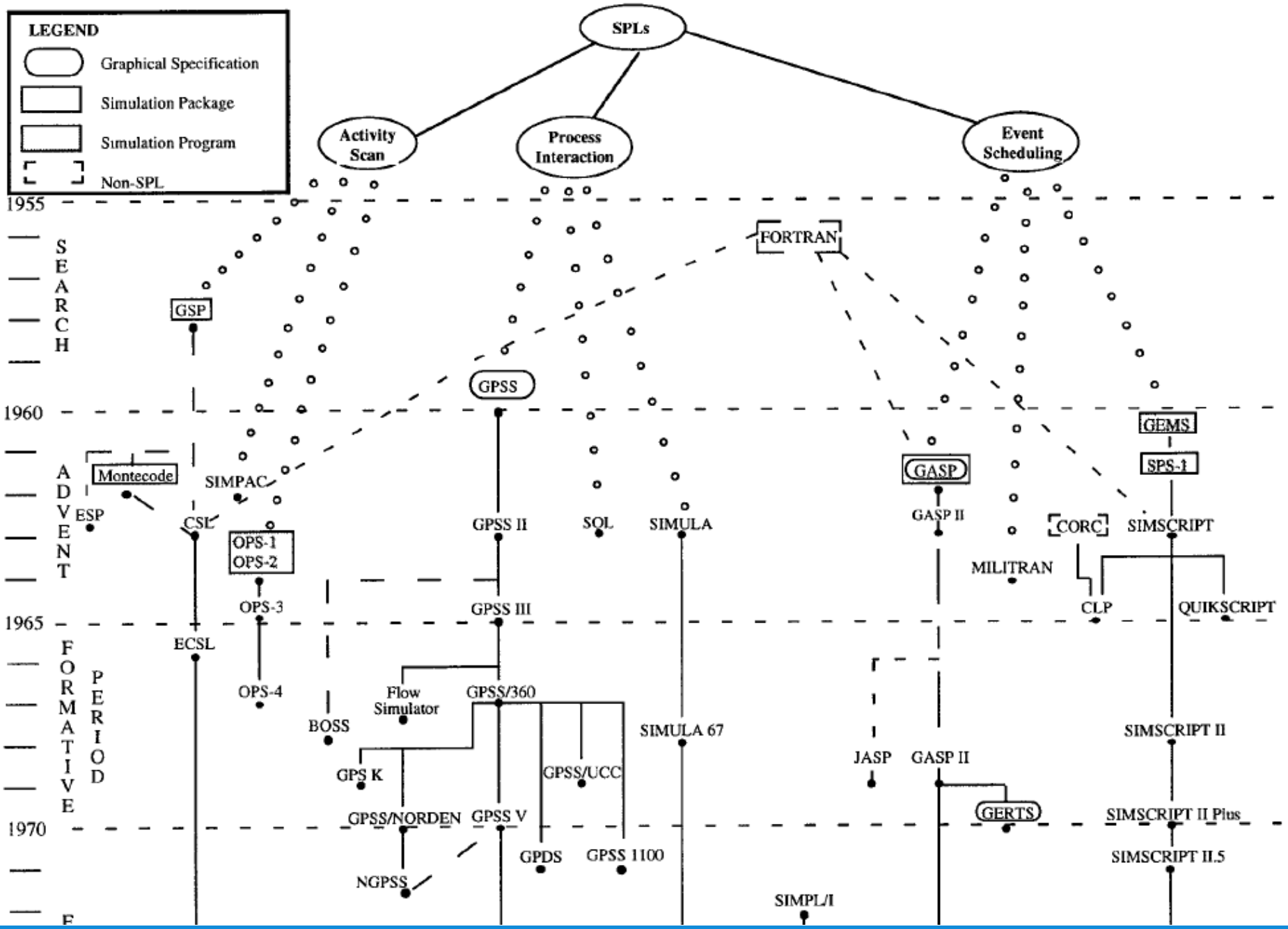
- **Event scheduling** provides locality of **time**: each event routine describes related actions that may all occur at one single instant
- **Activity scanning** provides locality of **state**: each activity routine describes all actions that occur because a particular state is reached
- **Process interaction** provides locality of **object**: each process routine describes the entire action of a particular model object

# What languages are there?

- Event scheduling languages: Arena, Plant Simulation, Automod, DSOL, Enterprise Dynamics, Simio, most modern discrete-event simulation environments
- Process interaction languages: Simula, DSOL
- Activity scanning: GSP, EDSIM (time-expensive)

# Early history of simulation languages

- Object-orientation is first implemented in a programming language by Ole-Johan Dahl and Kristian Nygaart in their simulation language called SIMULA'67
- SIMULA later became a general purpose programming language
- C++, Java, C#, Delphi, Oberon are all general purpose object-oriented programming languages


LEGEND
Graphical Specification
Simulation Package
Simulation Program
Non-SPL
SPLs
Activity Scan
Process Interaction
Event Scheduling
1955
1960
1965
1970
SEARCH
ADVENT
FORMATIVE
PERIOD
FORTRAN
GSP
GPSS
Montecode
SIMPAC
ESP
CSL
OPS-1
OPS-2
OPS-3
OPS-4
ECSL
GPSS II
GPSS III
SOL
SIMULA
BOSS
Flow Simulator
GPSS/360
GPS K
GPSS/UCC
GPSS/NORDEN
GPSS V
GPDS
GPSS 1100
NGPSS
SIMULA 67
SIMPL/I
GASP
GASP II
JASP
GERTS
MILITRAN
CORC
CLP
GEMS
SPS-1
SIMSCRIPT
QUIKSCRIPT
SIMSCRIPT II
SIMSCRIPT II Plus
SIMSCRIPT II.5

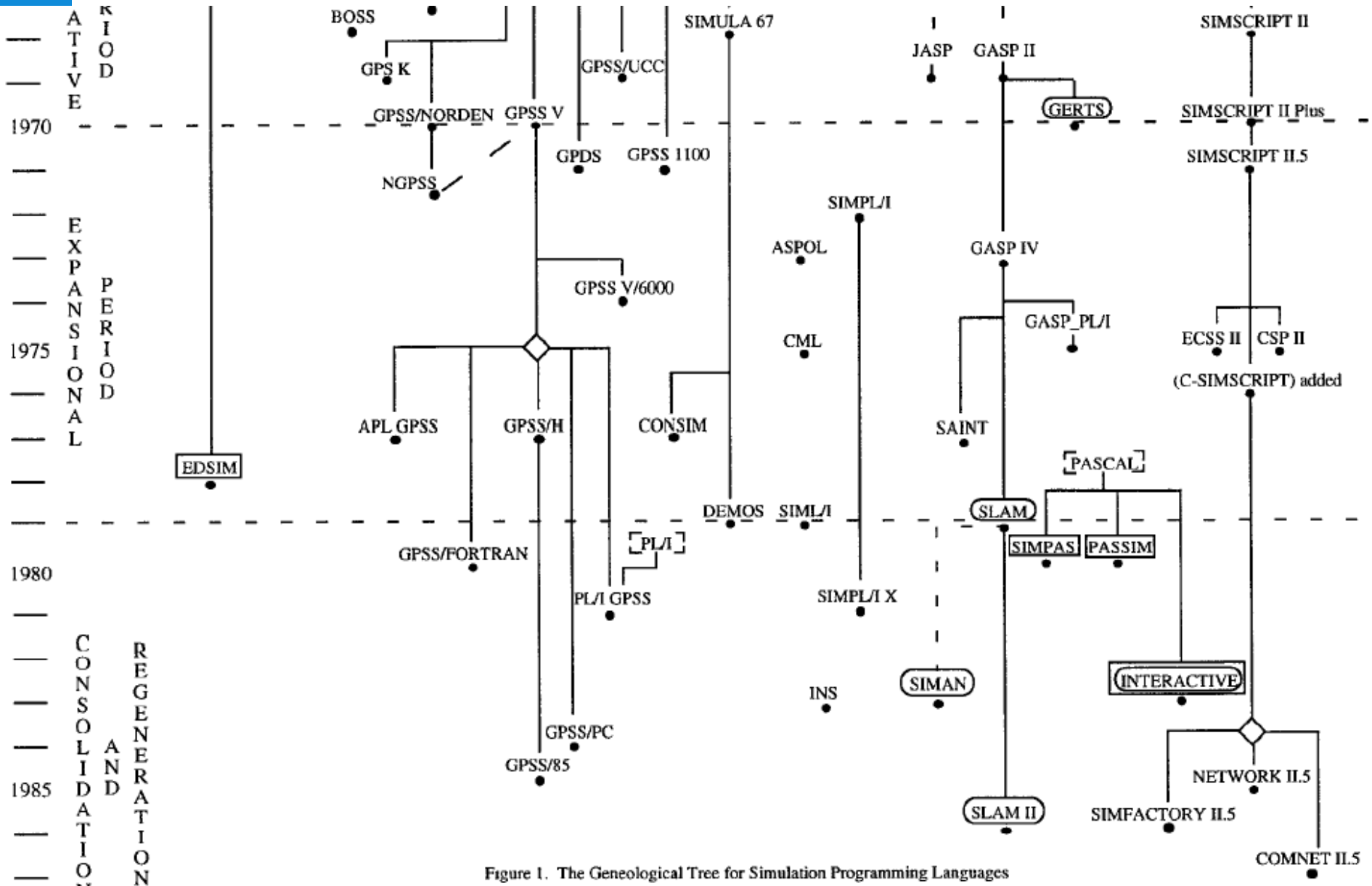
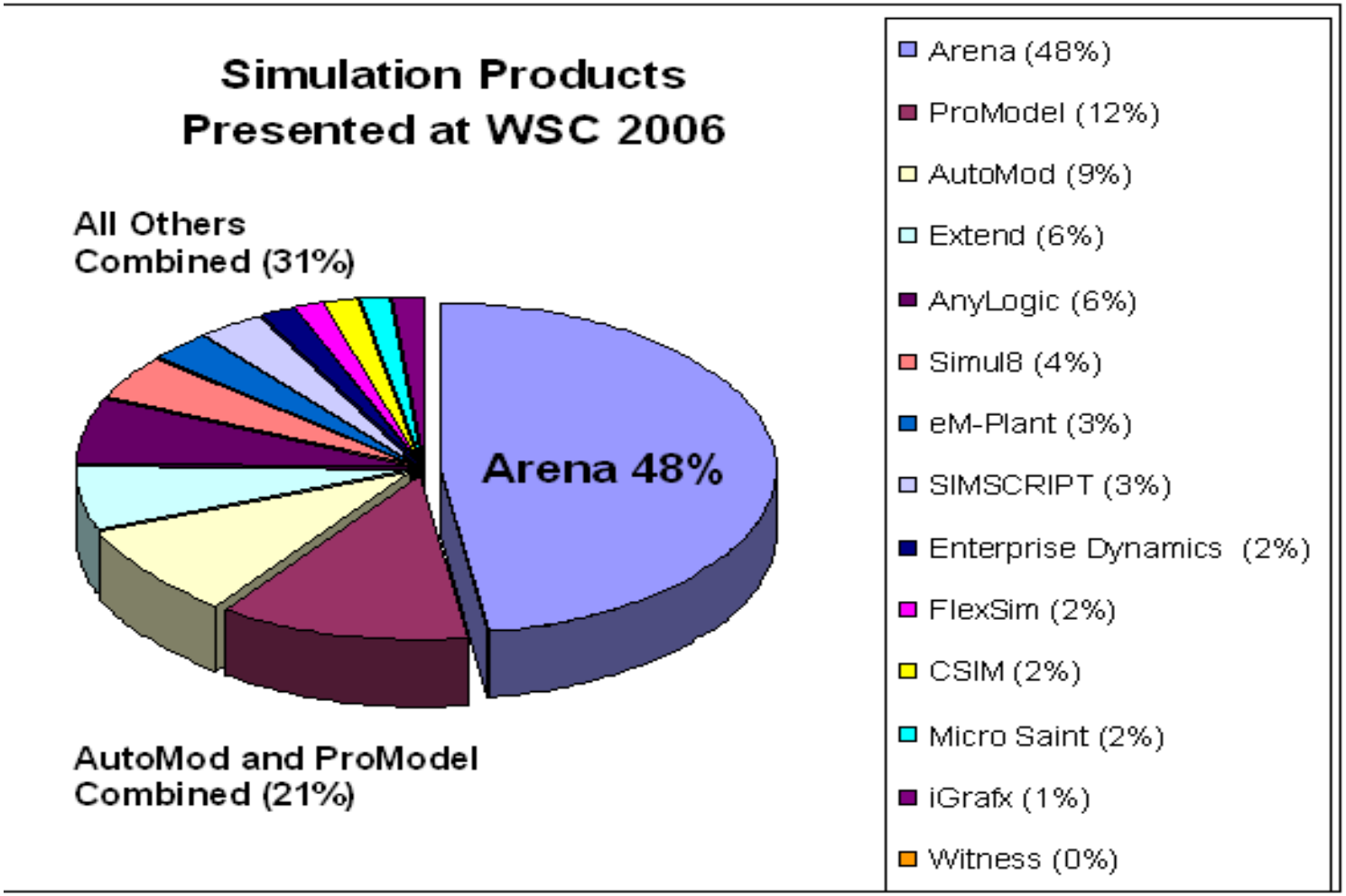



Figure 1. The Geneological Tree for Simulation Programming Languages

# 2006/2007 Simulation Market

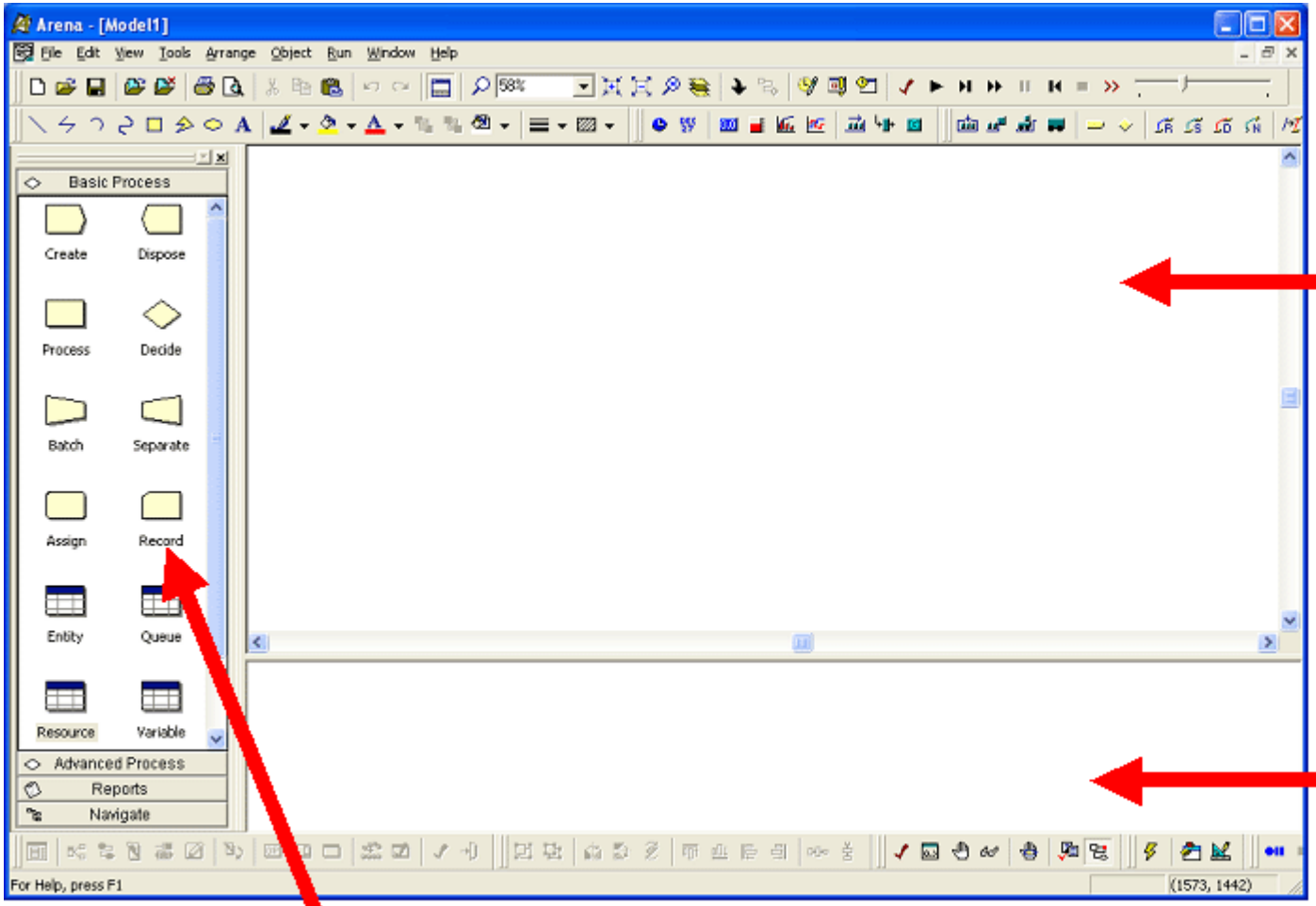



(http://www.arenasimulation.com/news/default.asp, 11 December 2007)

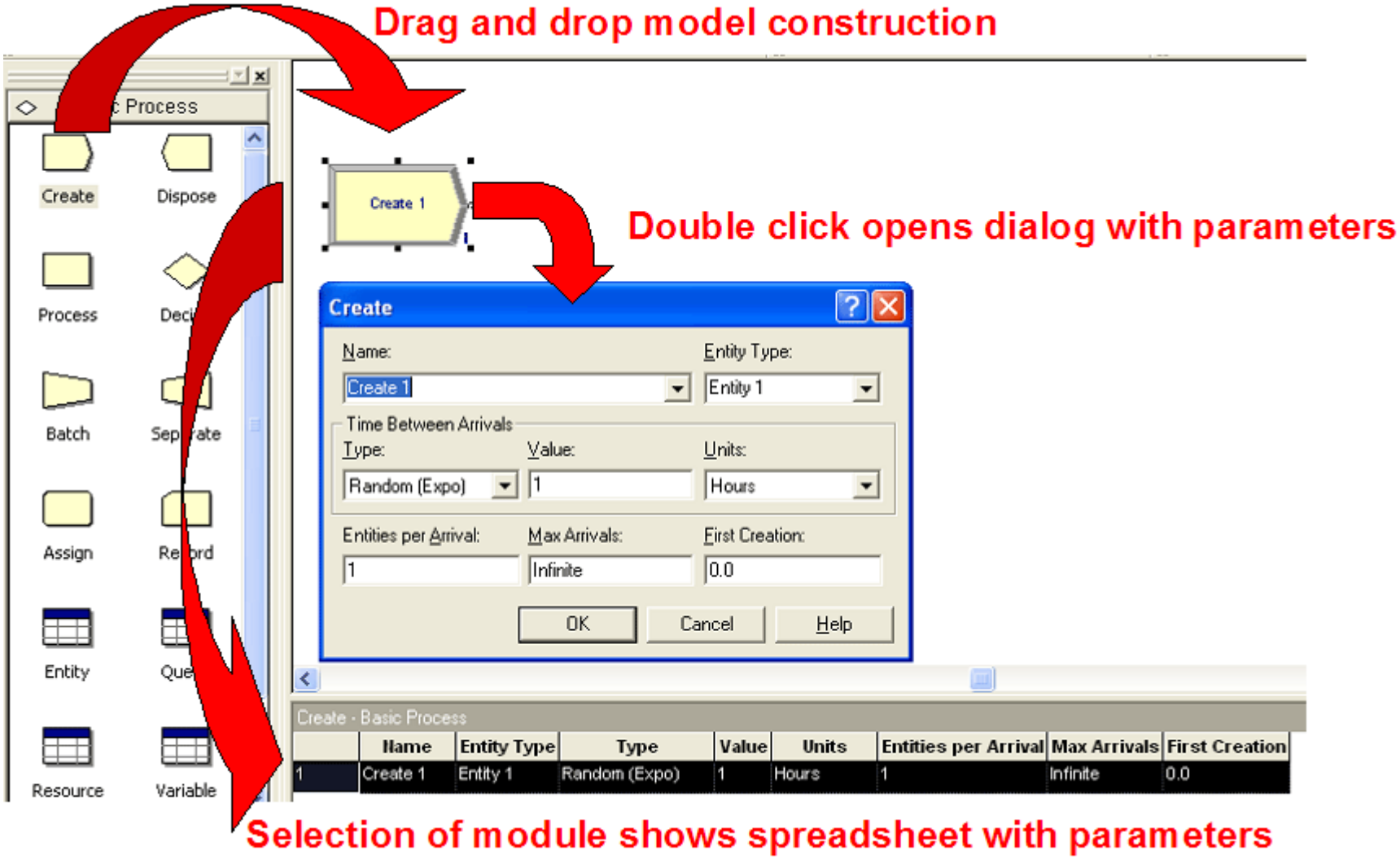
# 2.

# Arena

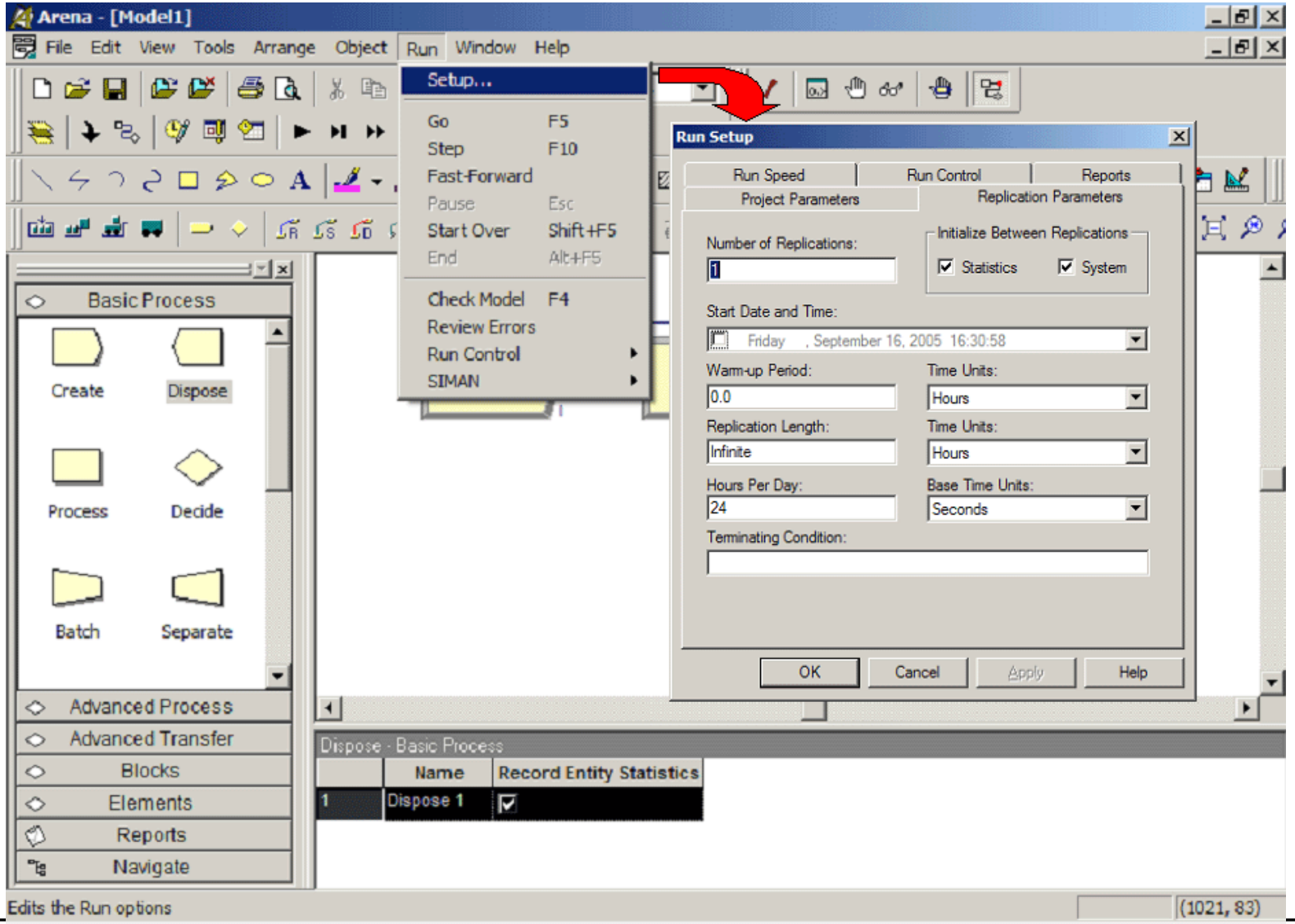
# Arena overview



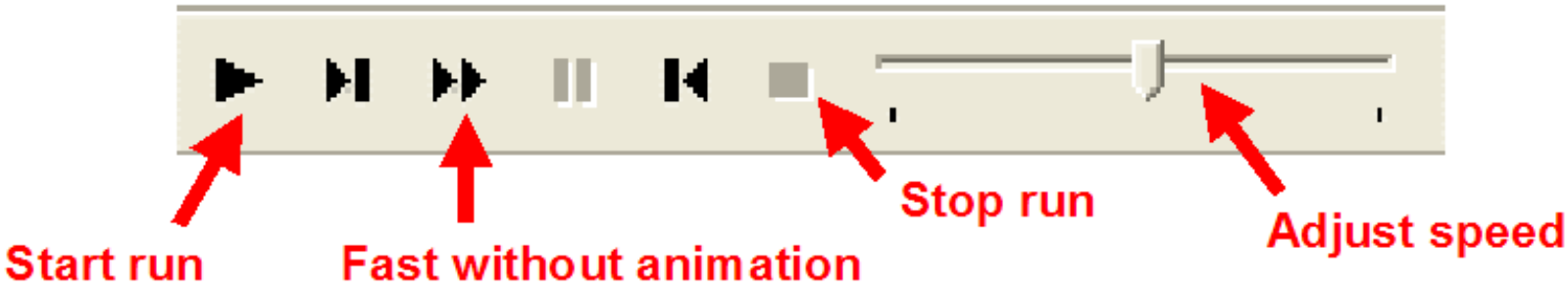
# Arena: drag - and - drop interface

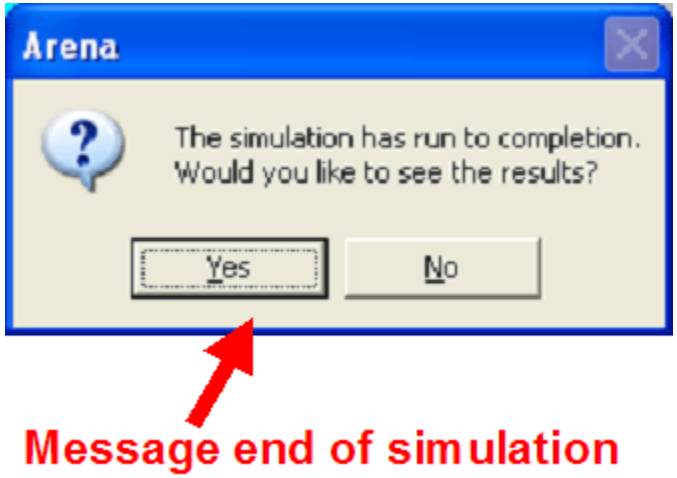


# Arena: run setup

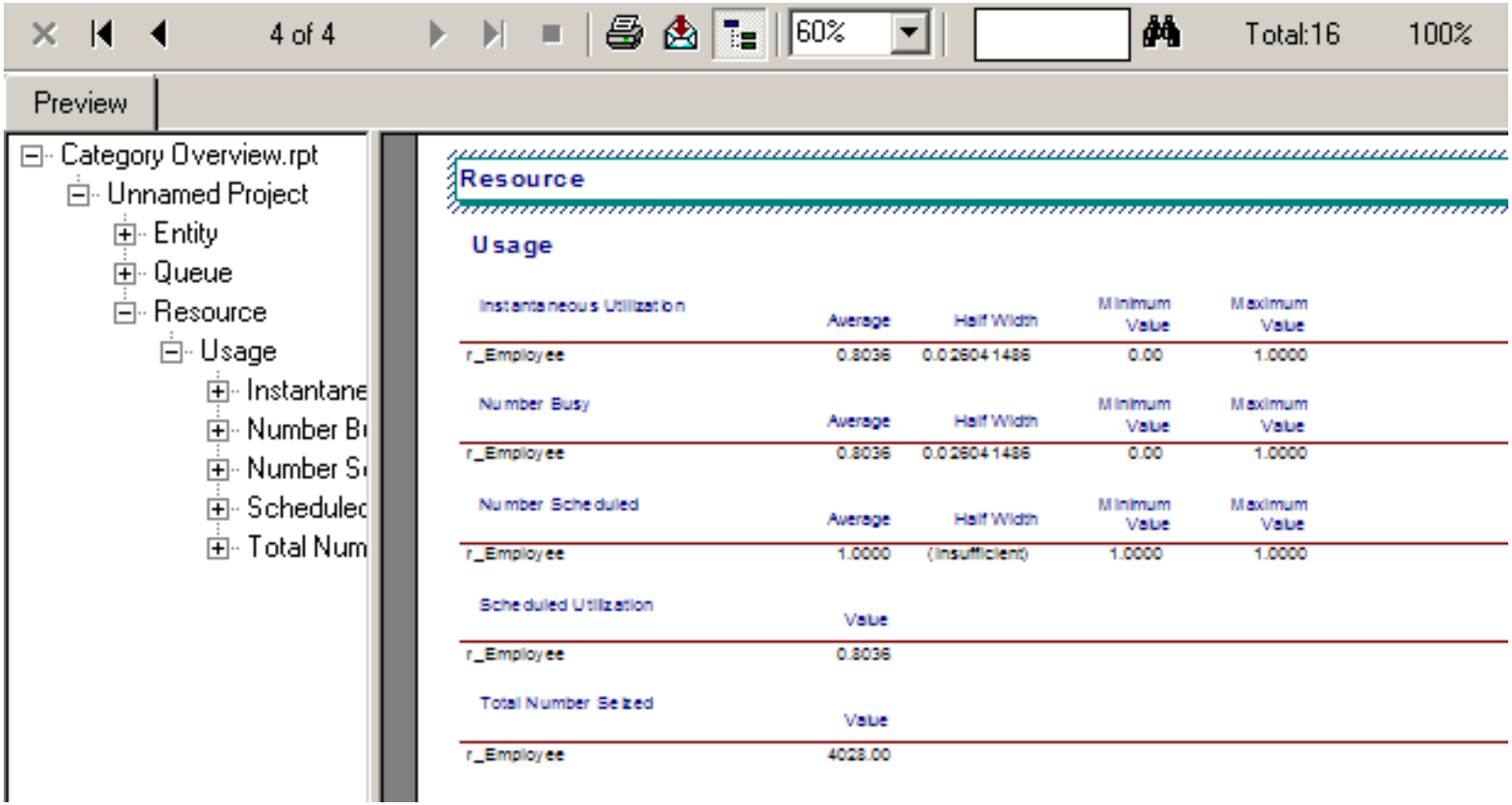


# Arena: run interaction

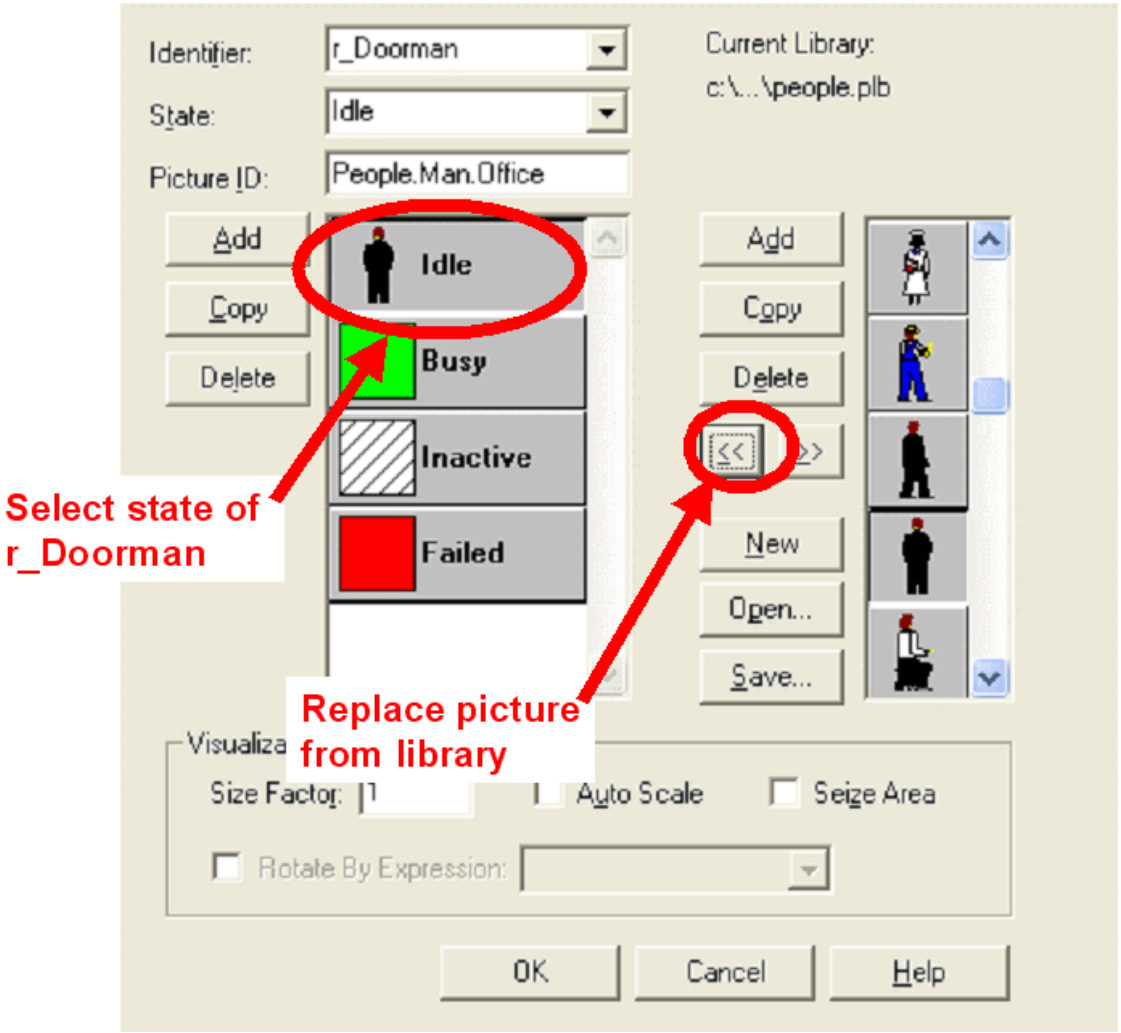


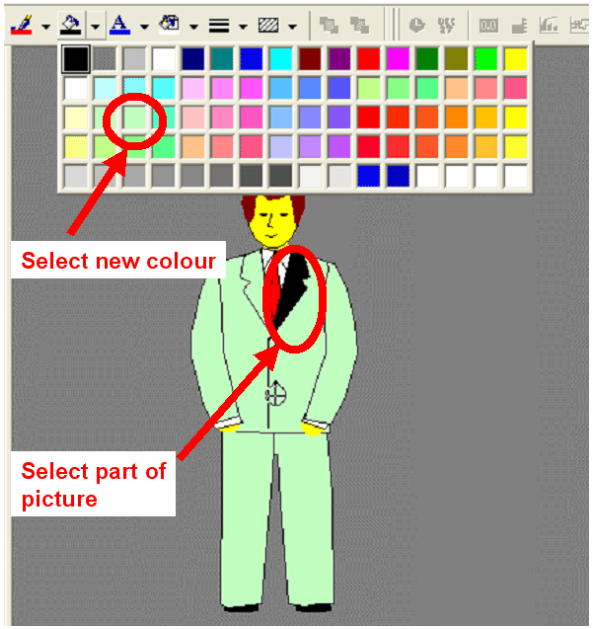



# Arena: reporting



# Arena: animation



# Arena: Modeling concepts

Basic low level blocks

- Queue -> Seize -> Delay -> Release

**Transporter**

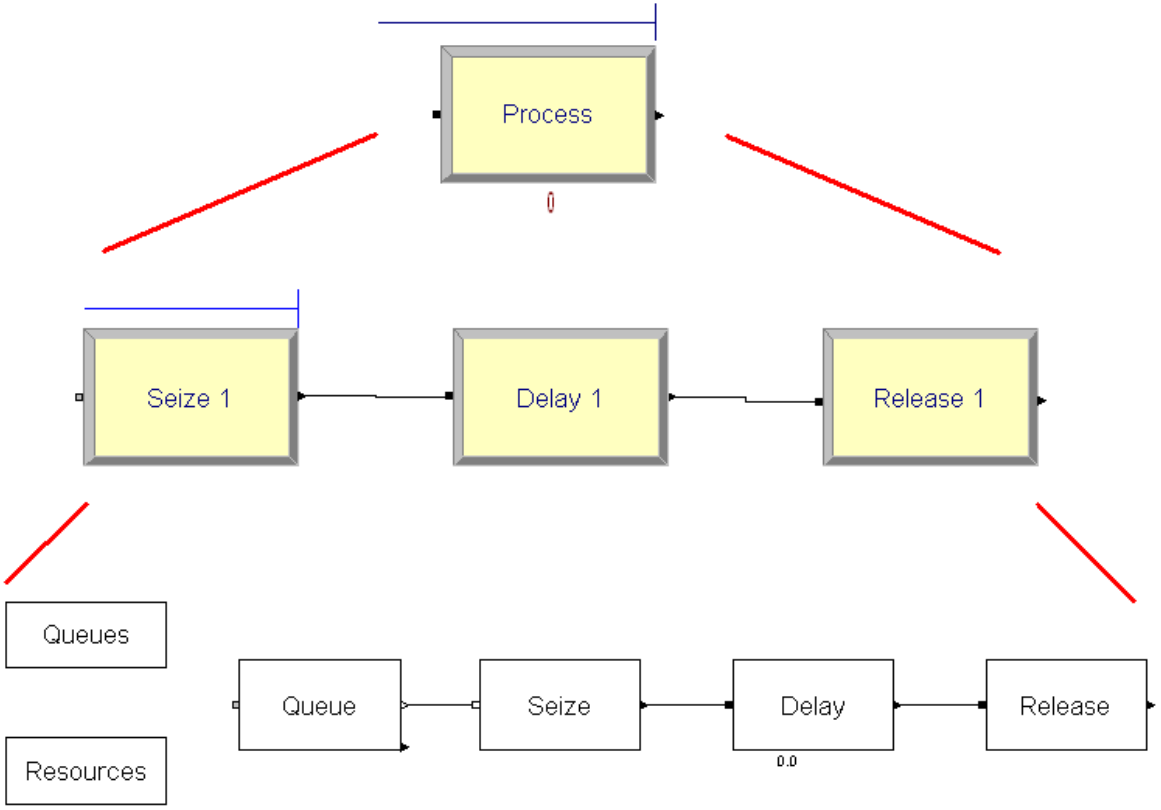
- Request
- Transport
- Free

**Conveyor**

- Access
- Convey
- Exit

# Arena basic simulation concepts: Operation

- An operation is a step carried out by or on an entity while it moves through a system

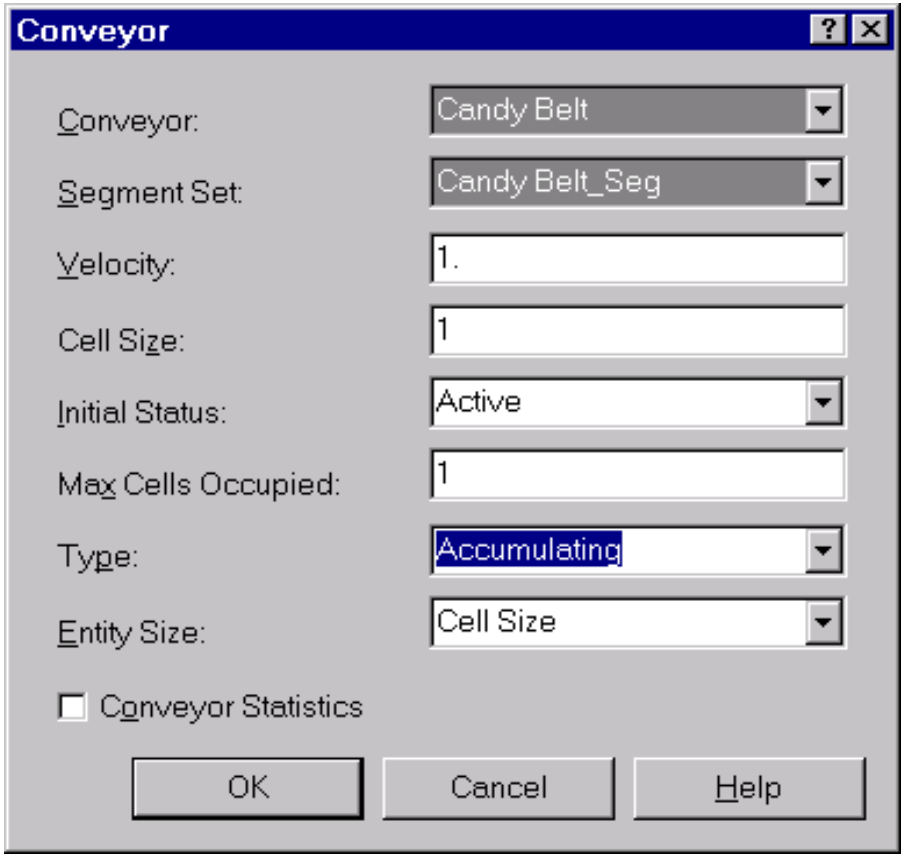


# Arena: Conveyor

- Objects must wait for sufficient space on the conveyor
- Unidirectional
- Entities can not pass one another
- Two different types
  - Accumulating
  - Non-accumulating

# Arena: Conveyor

- High level programming




# Arena: Conveyor

Variables: current information about the status of conveyors

- NEC number of entities being conveyed
- NEA number of accumulated entities
- LEC total length of entities being conveyed
- CLA total length of entities accumulated
- LC number of occupied cells
- MLC total conveyor length
- ICS conveyor status
- VC conveyor velocity

# Arena: Free-path transporter

- Transporter can freely move about system without encountering any obstruction
- Travel time depends on the fixed distance traveled and the speed of the transporter

# Arena: Guided transporter

- Movement is restricted to a pre-defined network of intersections and connecting links
- Travel time depends on:
  - vehicle's speed characteristics
  - path, congestion on paths
- Automated Guided Vehicles

# 3.

# Enterprise Dynamics

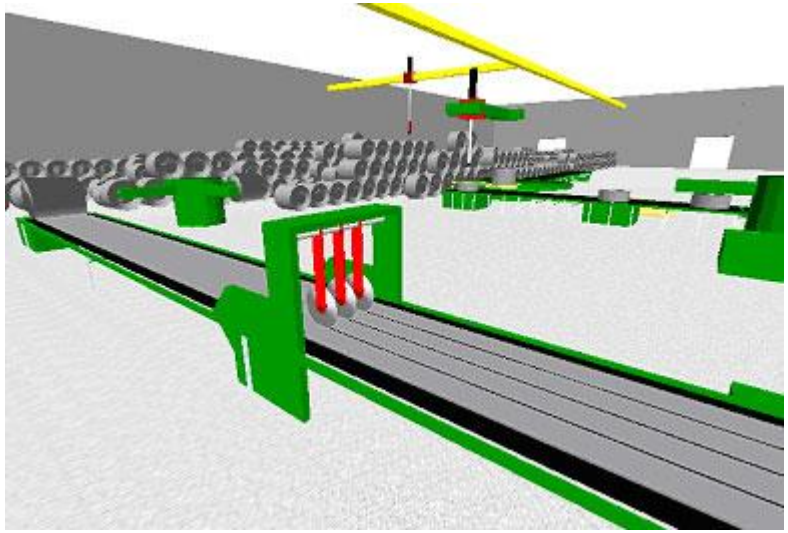
# Enterprise Dynamics: Characteristics (1)

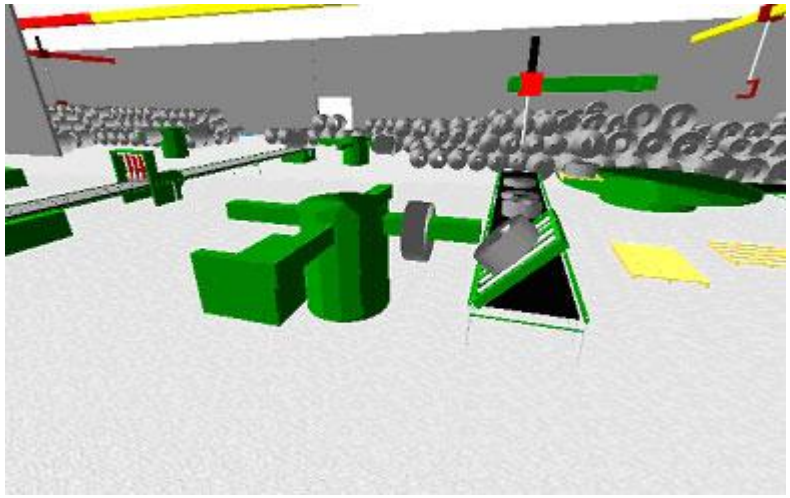
- Discrete Event Simulation package (DEVS)
- "Object Oriented" (Atoms)
  - Atom = object with 4 dimensions (x, y, z, time)
  - Each atom can have a **location**, **speed, rotation** and **dynamic behavior**
- Strong graphical representation
  - Immediate visualization in 3D/VR
  - Different Templates + Building Blocks
  - Logistics Suite
  - Airport Suite
  - Educational Suite

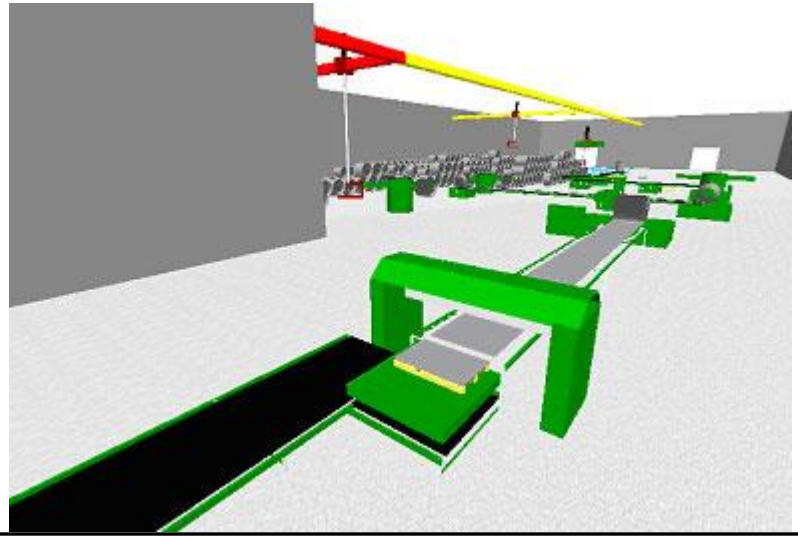
# Enterprise Dynamics: Characteristics (2)

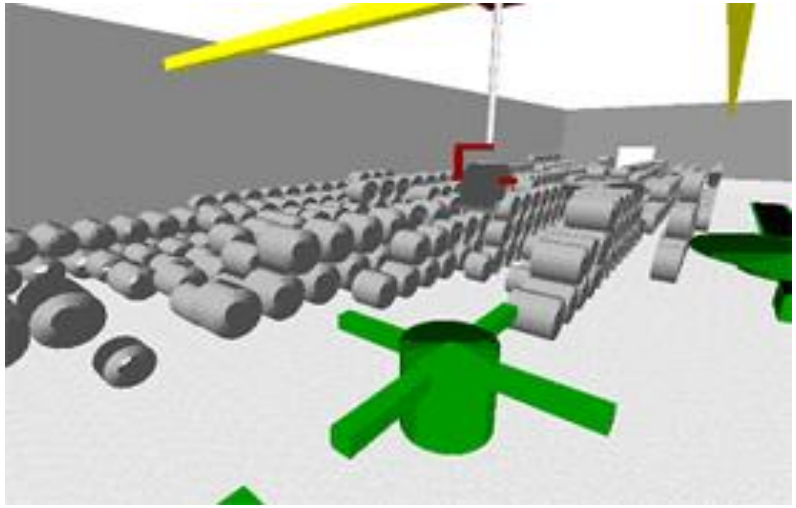
- Compatible with a lot of databases
  - MS Access, Oracle, SQL Server etc.
- Communication with external sources
  - MS Word, MS Excel, .txt-files, .xml-files etc.
  - Socket communication through TCP/IP
  - Movie creation is possible: .avi-files and .mpg-files
- HLA / IEEE 1516 implementations

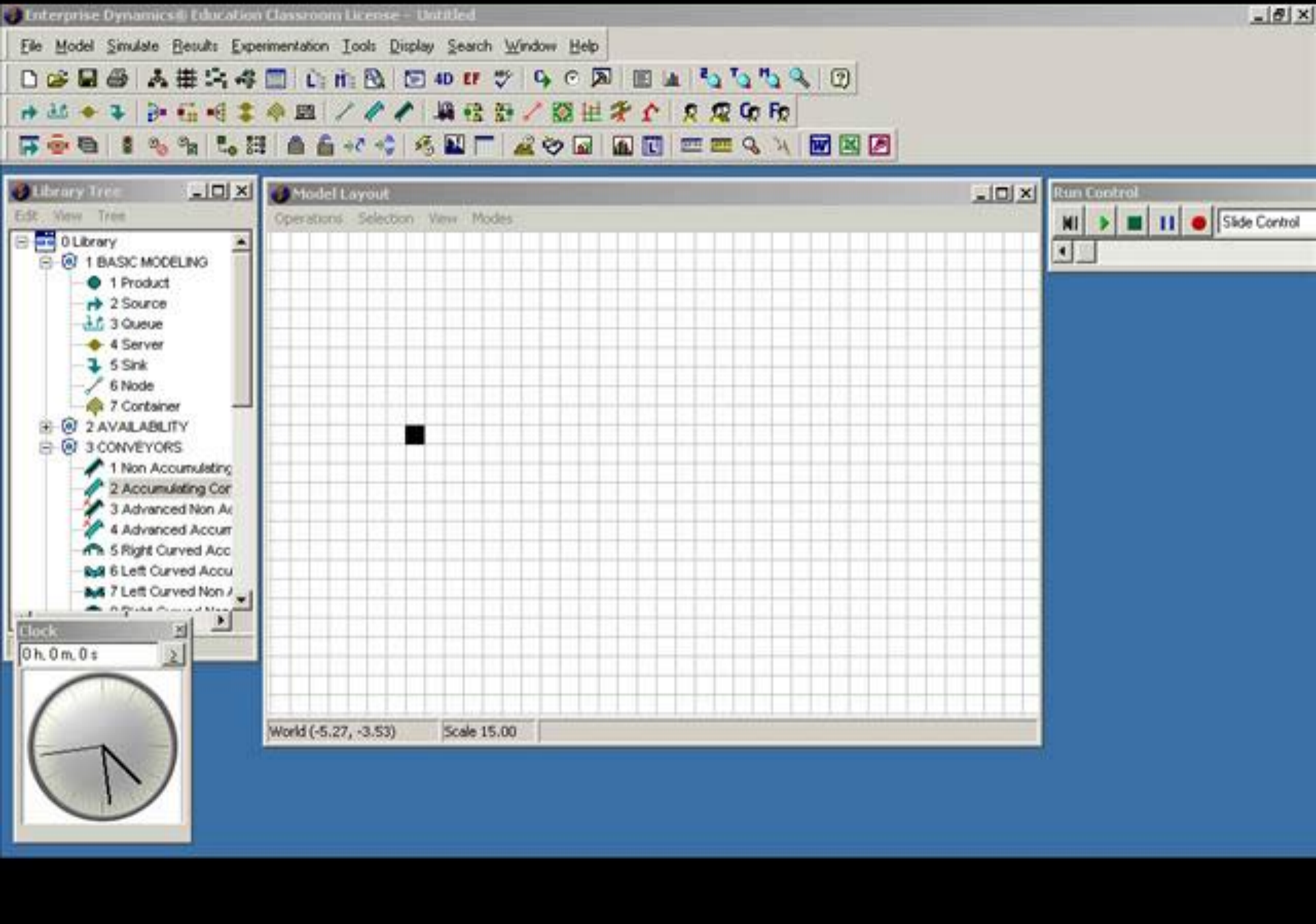
# Enterprise Dynamics










Enterprise Dynamics® Education Classroom License - Untitled
File Model Simulate Results Experimentation Tools Display Search Window Help
Library Tree
Edit View Tree
0 Library
1 BASIC MODELING
1 Product
2 Source
3 Queue
4 Server
5 Sink
6 Node
7 Container
2 AVAILABILITY
3 CONVEYORS
1 Non Accumulating
2 Accumulating Cor
3 Advanced Non Ac
4 Advanced Accum
5 Right Curved Acc
6 Left Curved Accu
7 Left Curved Non A
Clock
0 h, 0 m, 0 s
Model Layout
Operations Selection View Modes
World (-5.27, -3.53)
Scale 15.00
Run Control
Slide Control

# 4.

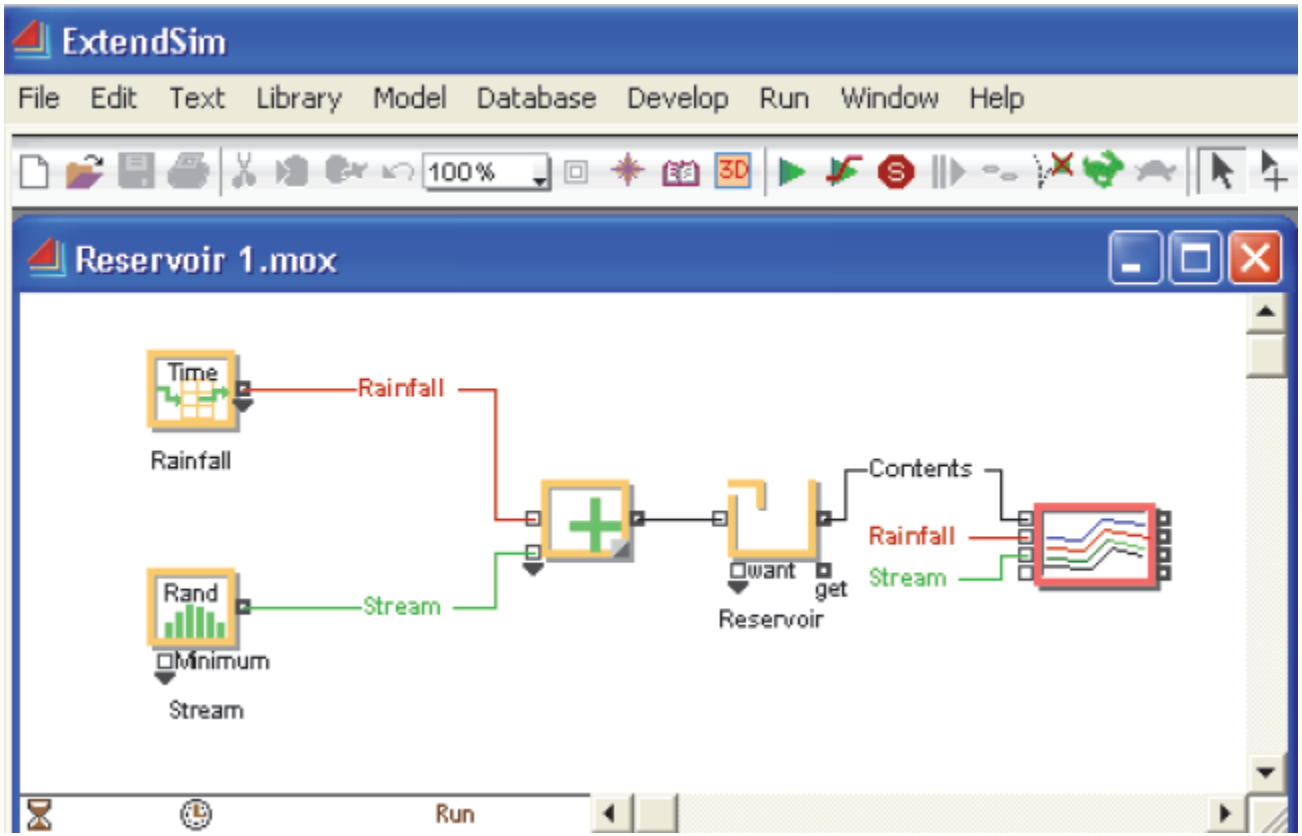
# ExtendSim

# ExtendSim

- Started as a continuous simulation package in 1987 for Apple hw
- 1990 Discrete-event capabilities added
- 1995 Release for Windows
- 2008 Discrete-rate capabilities added

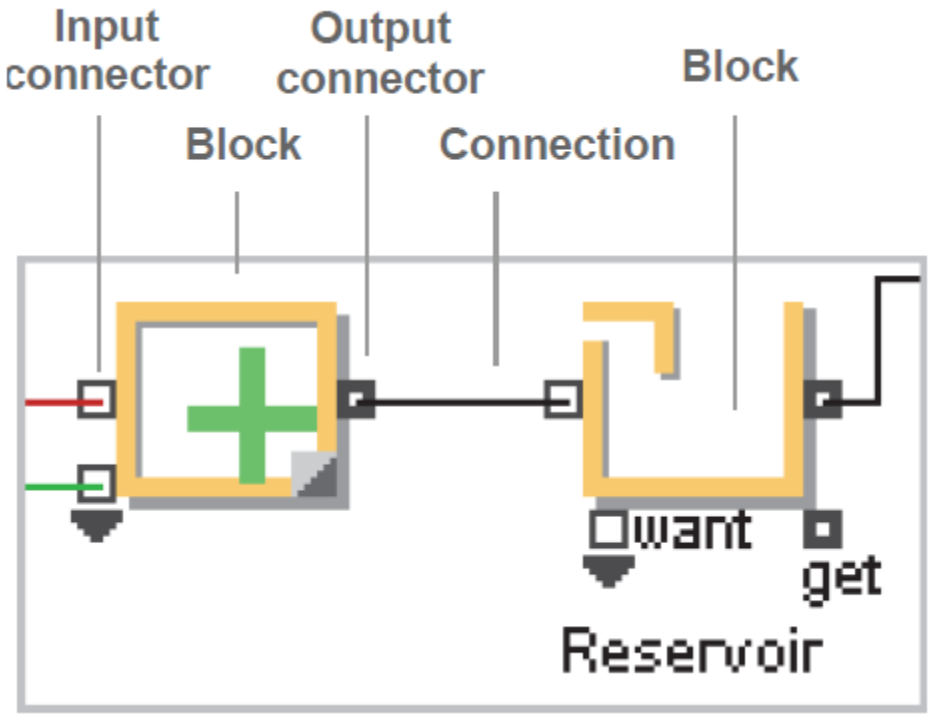
# Continuous modeling basis

Example: reservoir model; reservoir is filled by rainfall and a stream.



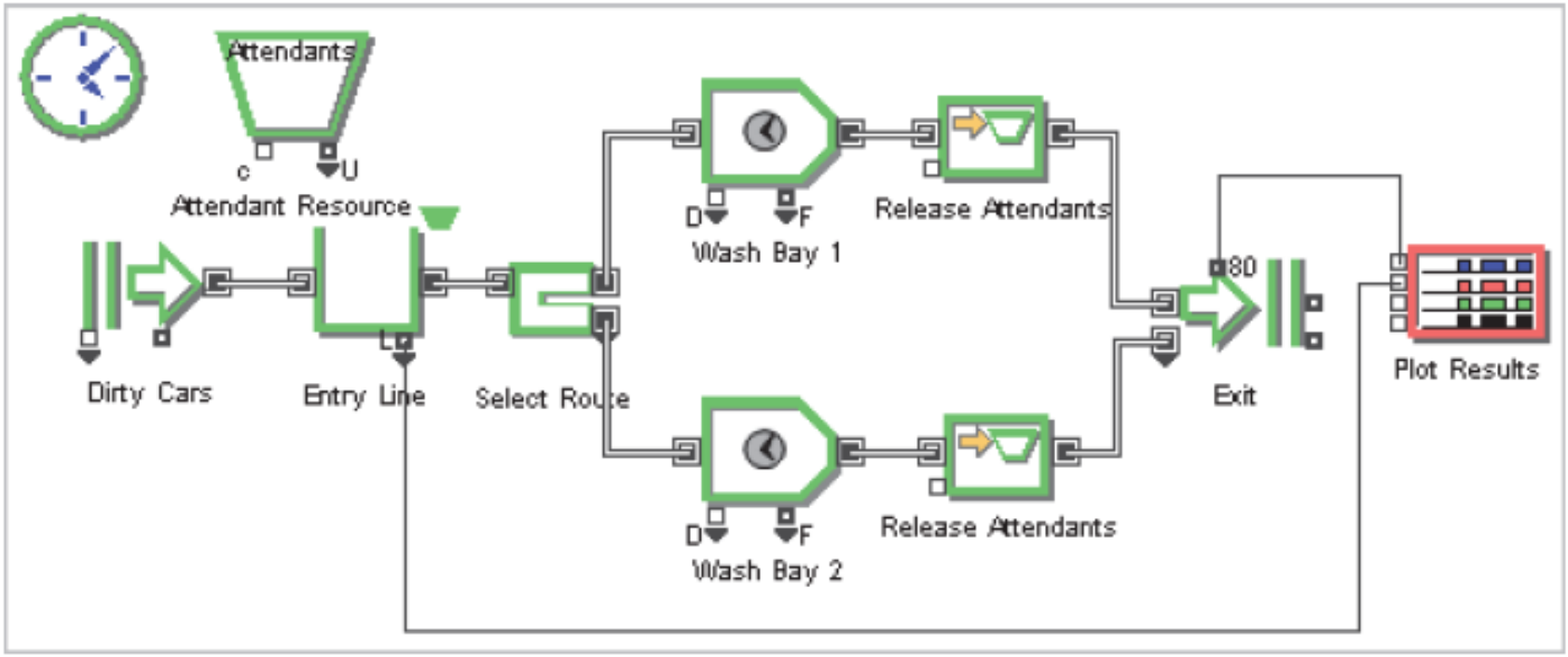
# Blocks work with ports & connectors

- Library-based; many blocks available in a library
- Different ports (called connectors in ExtendSim) per block



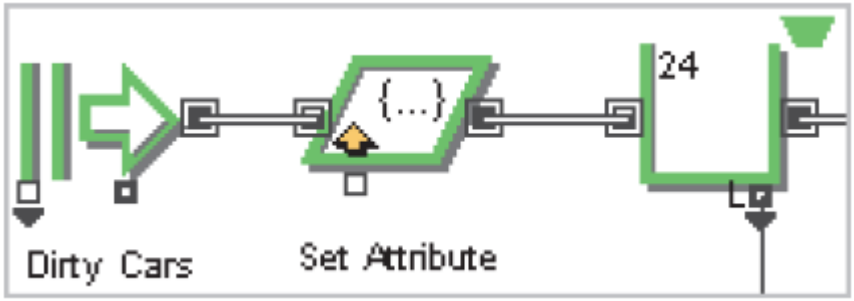
# Discrete-event models

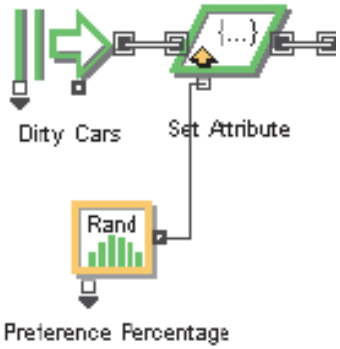
- Blocks for creating entities, queueing, selecting, resources, processes, leaving the model, graphs, etc.
- Model is purely logical – no physical space



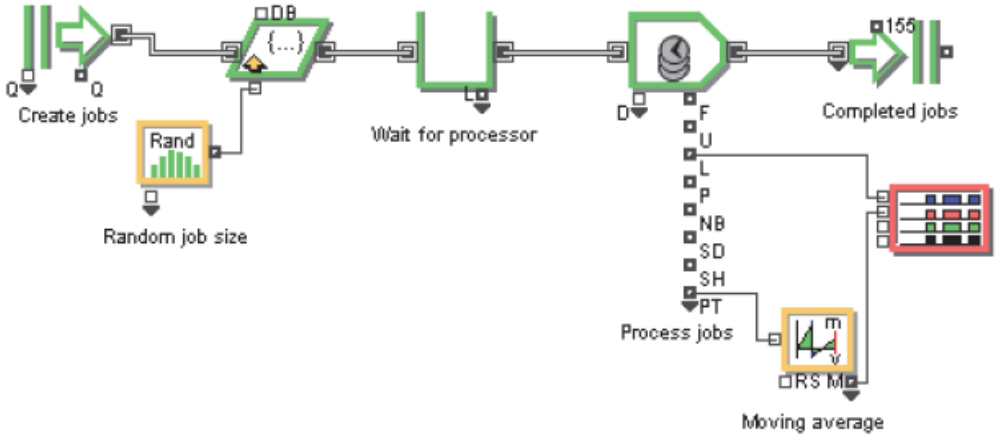
# Advanced modeling

- Interacting with attributes goes with blocks between the simulation logic. Random numbers as well:



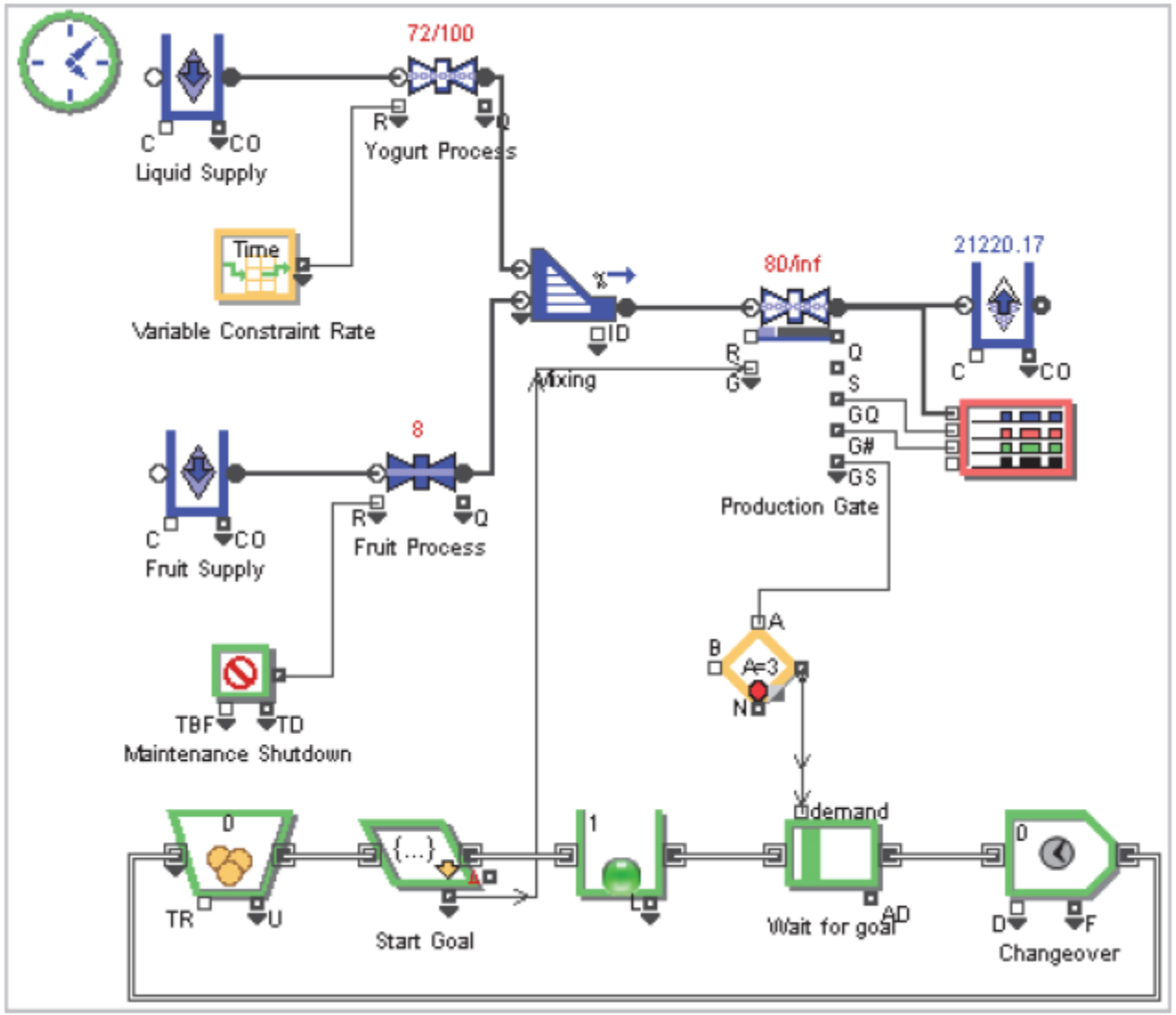




- Plotting and outputting also:



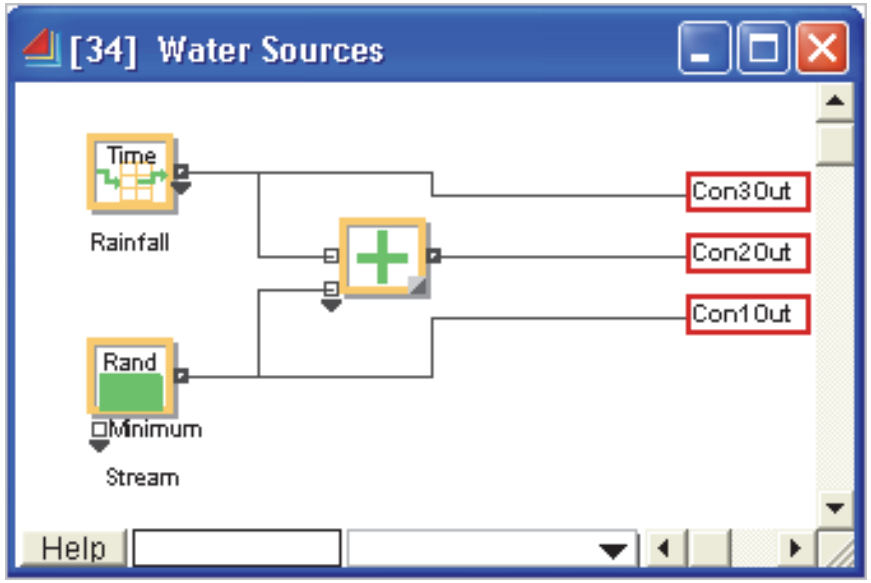
# Discrete-rate models

- flow
- valves
- pipes
- tanks
- mixing
- etc.



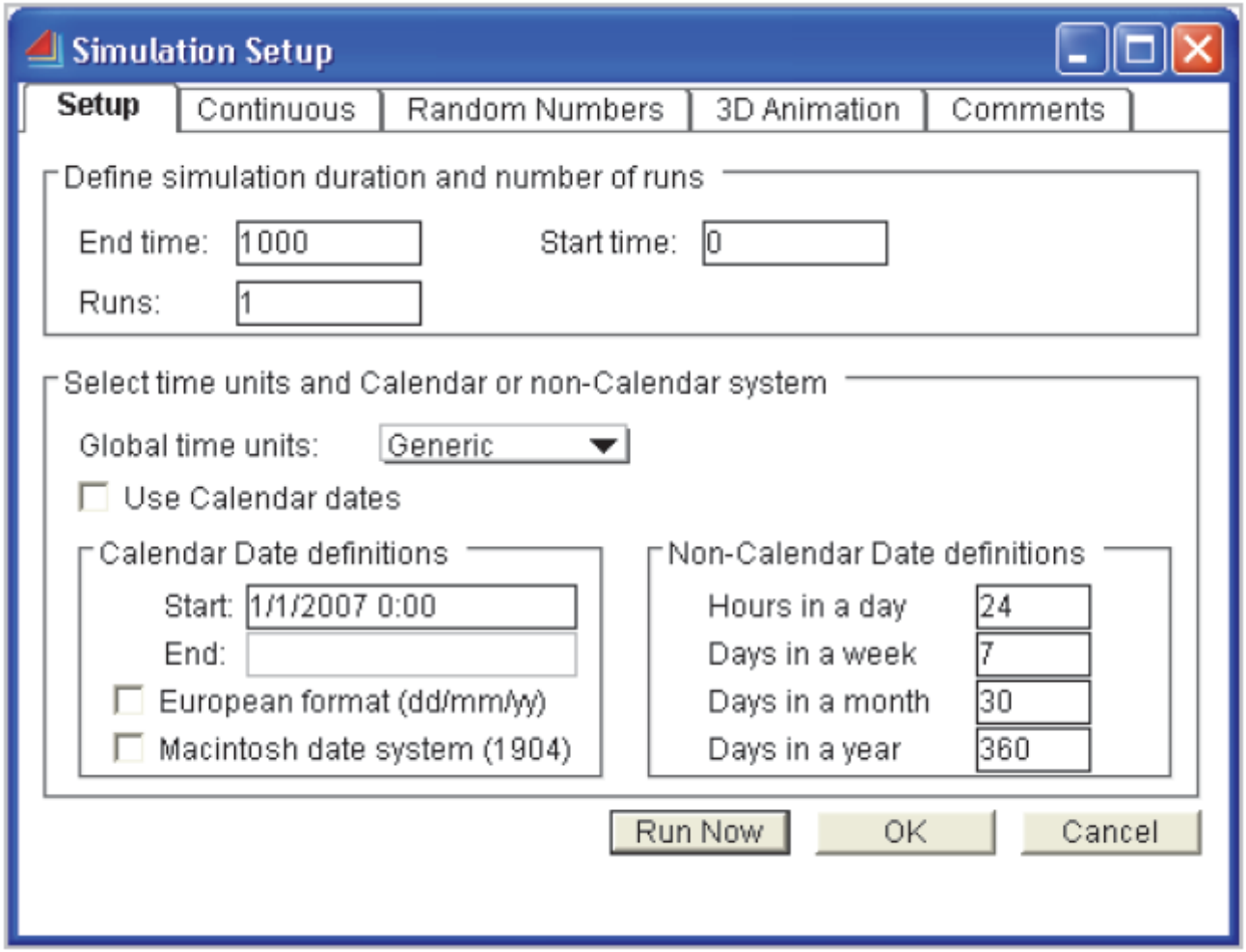
# Hierarchical modeling is standard

- Grab a couple of blocks and make a submodel out of it:




- Very elegant way of making new blocks

# Pretty complete run-setup



# But limited experimental design

A Scenario Manager block is available to help with running scenarios, but it is **part of the model**, instead of acting **on the model**.

| | Factor Name | Block Name | Block Number | Block Label | H-Block Name | H-Label[num] | Dialog Variable | Row,Column | Variable Value | Min Value | Max Value | Step |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Attendants | Resource P | 21 | Attendant | | [-1] | NumServ | -1,-1 | 2 | 1 | 3 | 1 |
| 1 | Wash Time | Activity | 4 | Wash Bay | | [-1] | waitDelta_prm | -1,-1 | 6 | 5 | 7 | 1 |
| 2 | Wash&Wax Time | Activity | 13 | Wash/Wax | | [-1] | waitDelta_prm | -1,-1 | 8 | 6 | 8 | 1 |
| 3 | Car Interarrival T | Create | 1 | Dirty Cars | | [-1] | md_Arg1_prm | -1,-1 | 4 | 3 | 5 | 0.5 |

Run results:

| | Select | Scenario Name | Attendants | Wash Time | Wash&Wax Time | Car Interarrival Time | (M)Average queu | (M)Average wait | Details |
|---|---|---|---|---|---|---|---|---|---|
| 0 | ☑ | Scenario 1 | 1 | 5 | 6 | 3 | 35.417179189399 | 101.889953021 | Show |
| 1 | ☑ | Scenario 2 | 1 | 5 | 6 | 3.5 | 24.054957756948 | 84.5324201843 | Show |
| 2 | ☑ | Scenario 3 | 1 | 5 | 6 | 4 | 15.370510260768 | 60.8295415103 | Show |
| 3 | ☑ | Scenario 4 | 1 | 5 | 6 | 4.5 | 10.941054253974 | 48.0812516130 | Show |
| 4 | ☑ | Scenario 5 | 1 | 5 | 6 | 5 | 6.1907290672059 | 30.2761953476 | Show |
| 5 | ☑ | Scenario 6 | 1 | 5 | 7 | 3 | 29.547632786837 | 91.7816165137 | Show |
| 6 | ☑ | Scenario 7 | 1 | 5 | 7 | 3.5 | 26.613653357253 | 91.5871396797 | Show |

# 5.

# Plant Simulation

# Plant Simulation

- Developed as SiMPLE++ at the Fraunhofer Institute for Manufacturing Engineering and Automation
- Commercial release in 1992 by AESOP
- 1997 Takeover by Tecnomatix Technologies LTD
  SIMPLE++ renamed to eM-Plant
- 2005 Takeover by UGS
- 2007 Takeover by Siemens

# Plant Simulation

- Designed for:
  - Discrete event simulation
  - Production and logistics
  - Planning and optimization

(Heinicke & Hickman, 2000)

- Plant Simulation is used by many industries:
  - Car manufacturers
  - Baggage handling at airport
  - Shipyard
  - International Finance

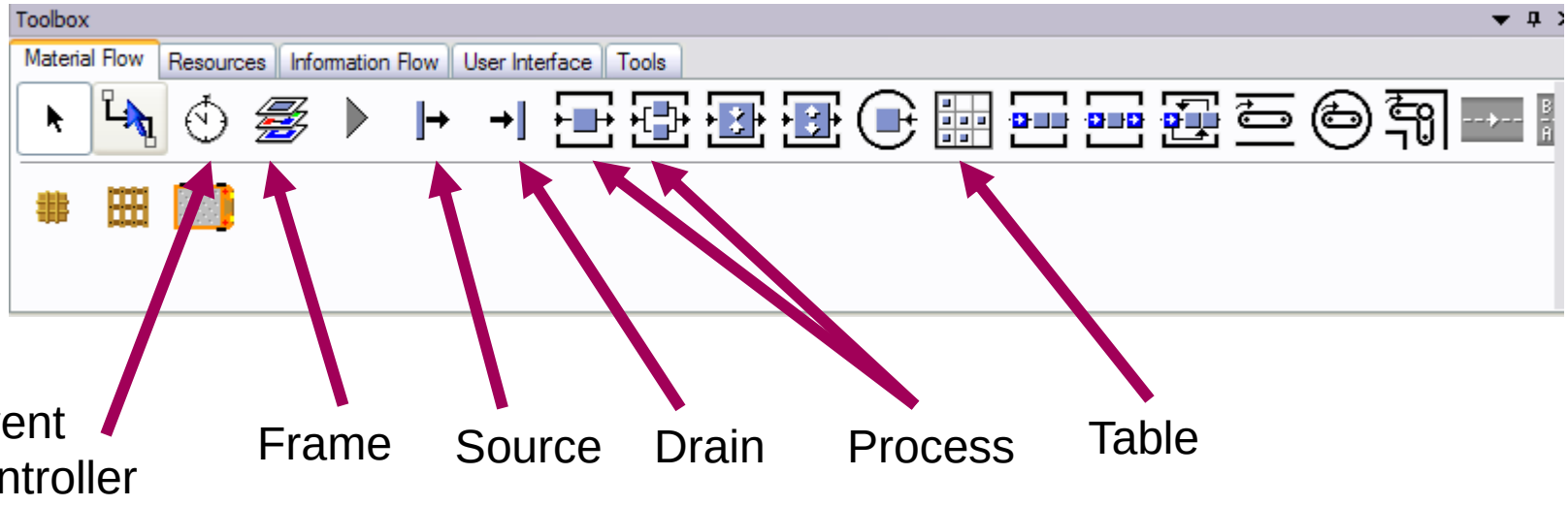
(Heinicke & Hickman, 2000)

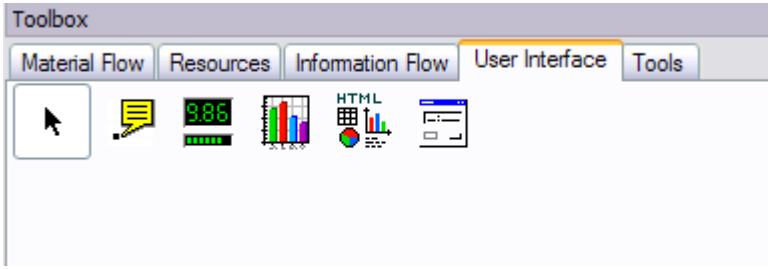
Heinicke, M. U., & Hickman, A. (2000). Eliminate bottlenecks with integrated analysis tools in eM-Plant. Winter Simulation Conference Proceedings.

# Characteristics

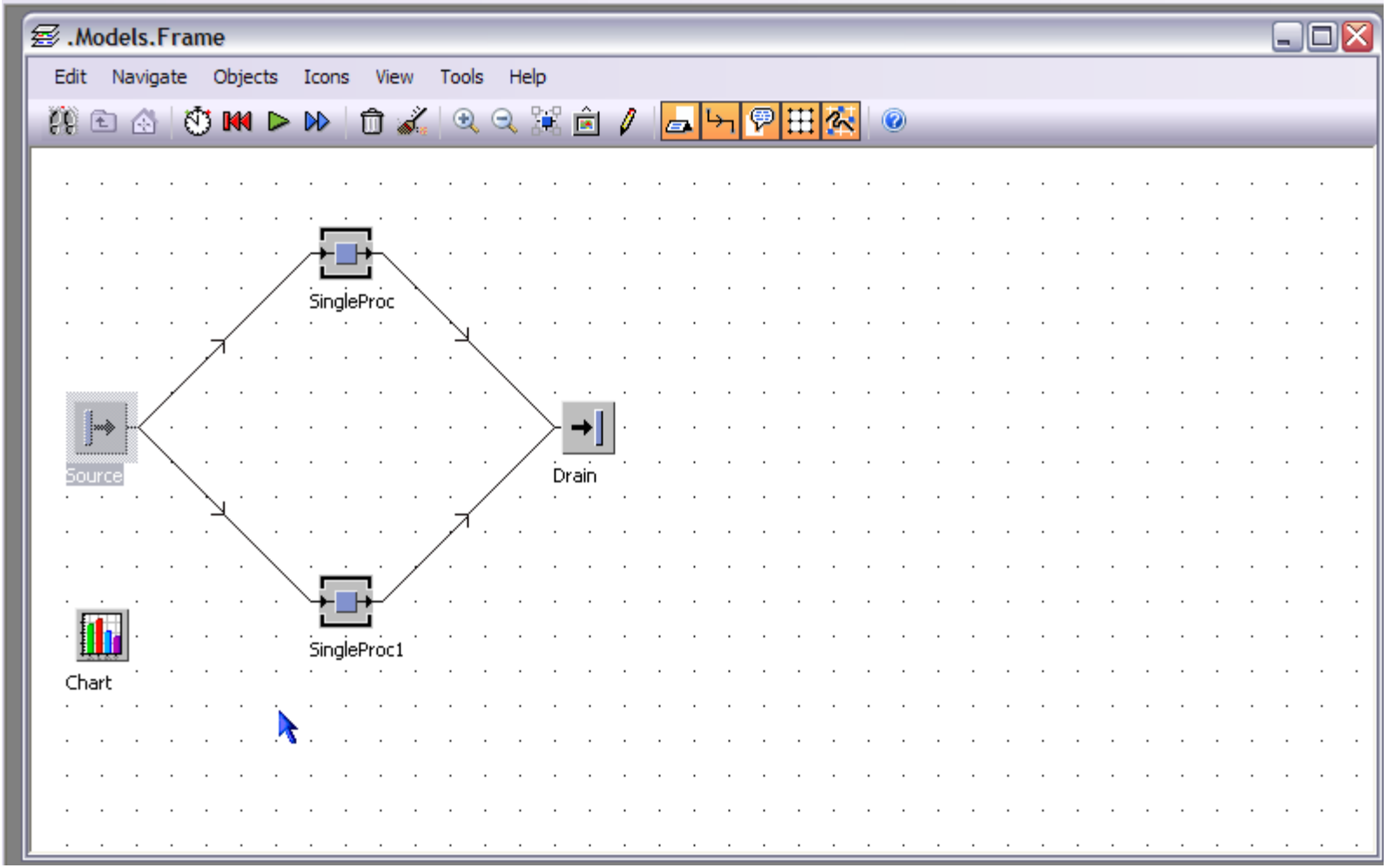
- Object oriented
- Discrete event simulation
- Object libraries enable hierarchical modeling
- Distributive use is possible
- Possibility to program new methods
- Visualisation: Easy 2D/3D

# Plant Simulation basic library

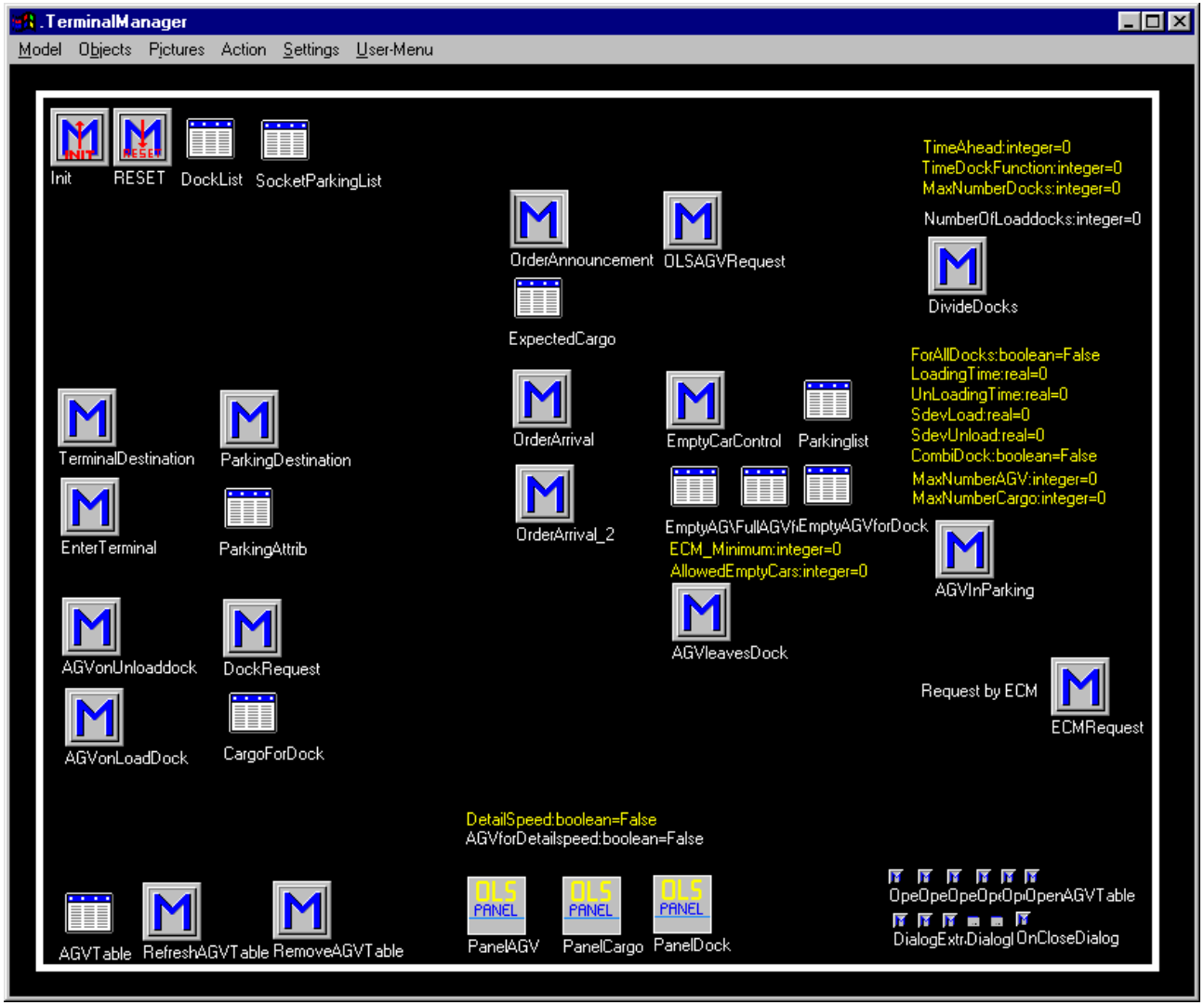


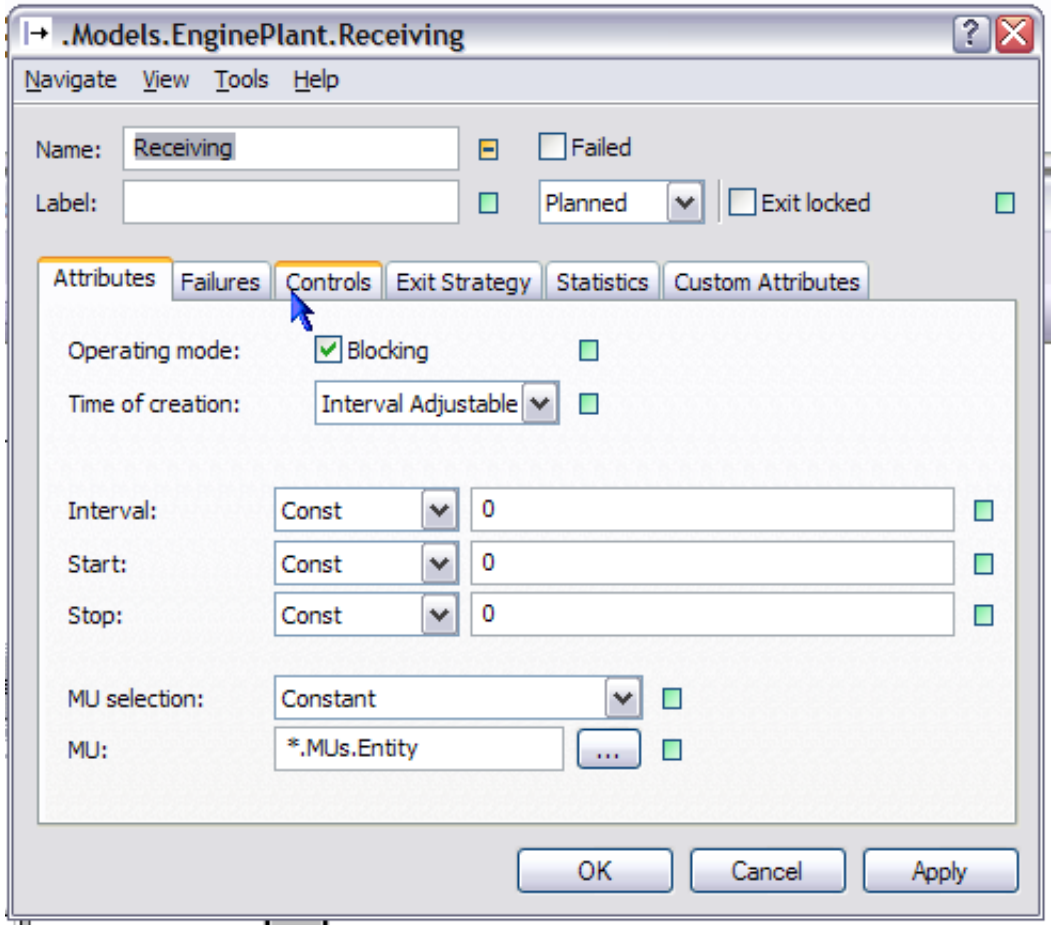
# Connecting processes in a frame



# Methods in a frame



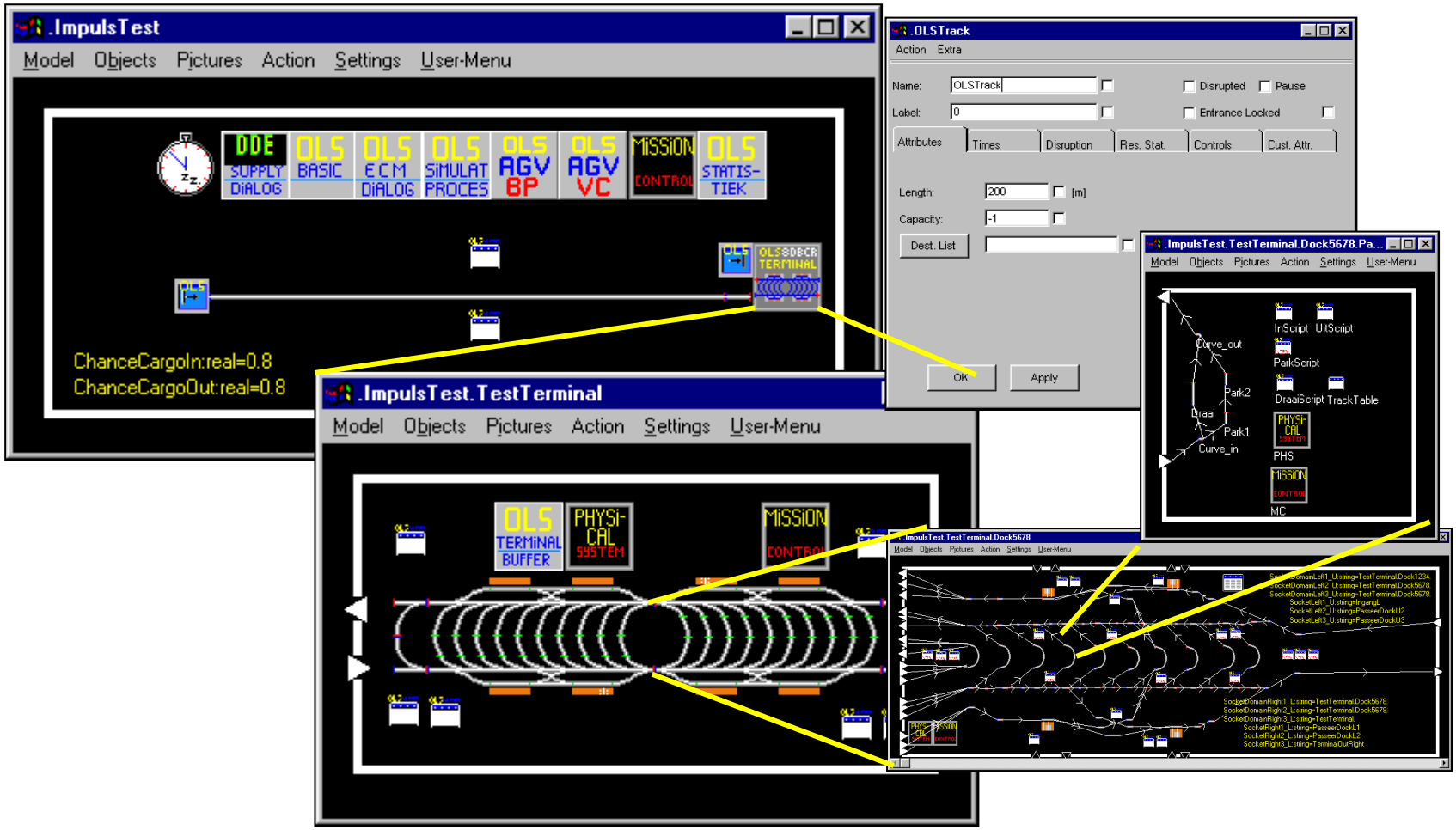
# Source set-up




Several attributes for the source:

- name
- inheritance
- interarrival time
- frequencies
- object generated
- disruptions
- statistics
- controls
- attributes of the object

# Complexity of hierarchies (2)

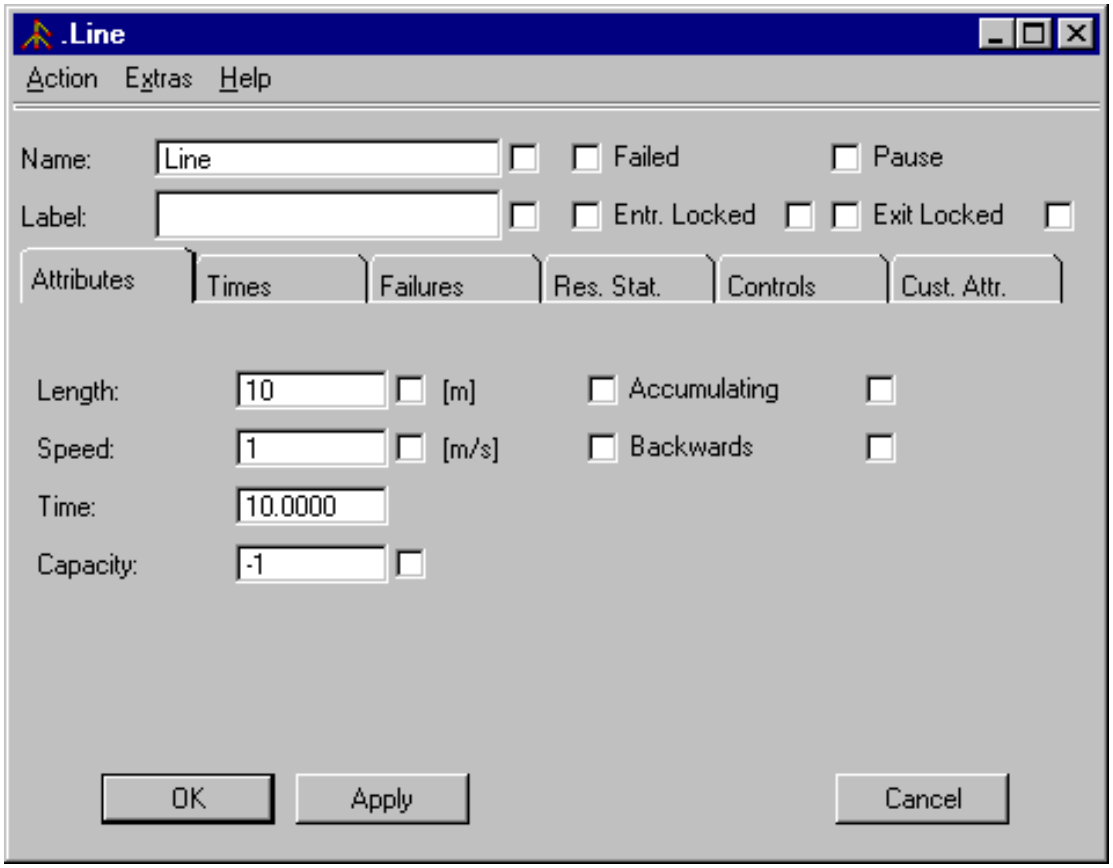


# Plant Simulation: Conveyors

- Model time delay
- Library objects
  - SingleProc, ParallelProc, Track, etc.
- Application objects
  - Frame: Straight
- Sensors that can be triggered

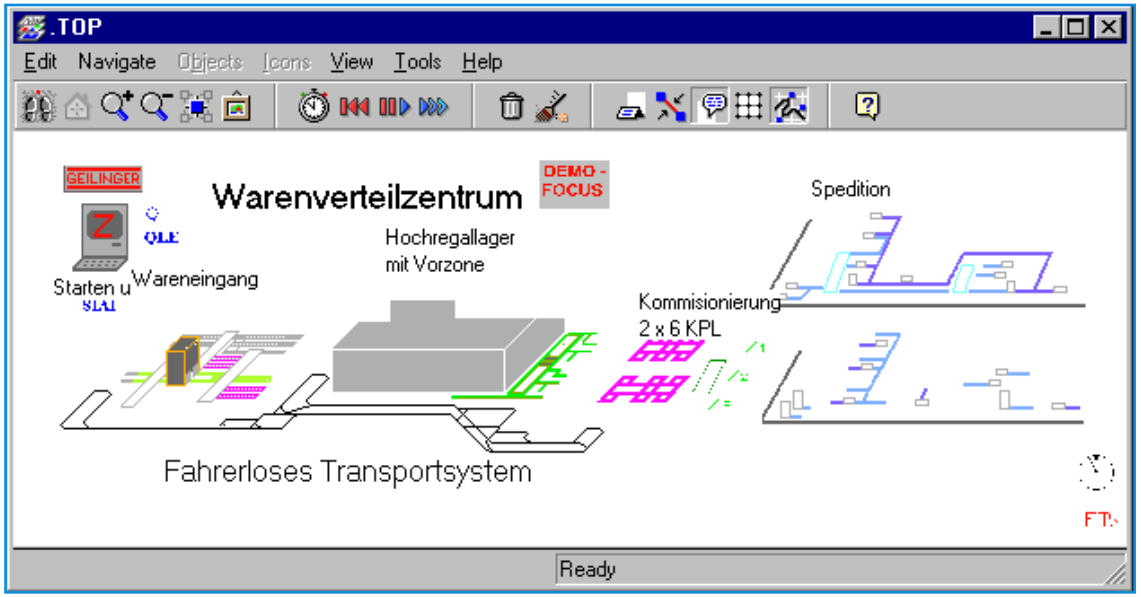
# Plant Simulation: Line

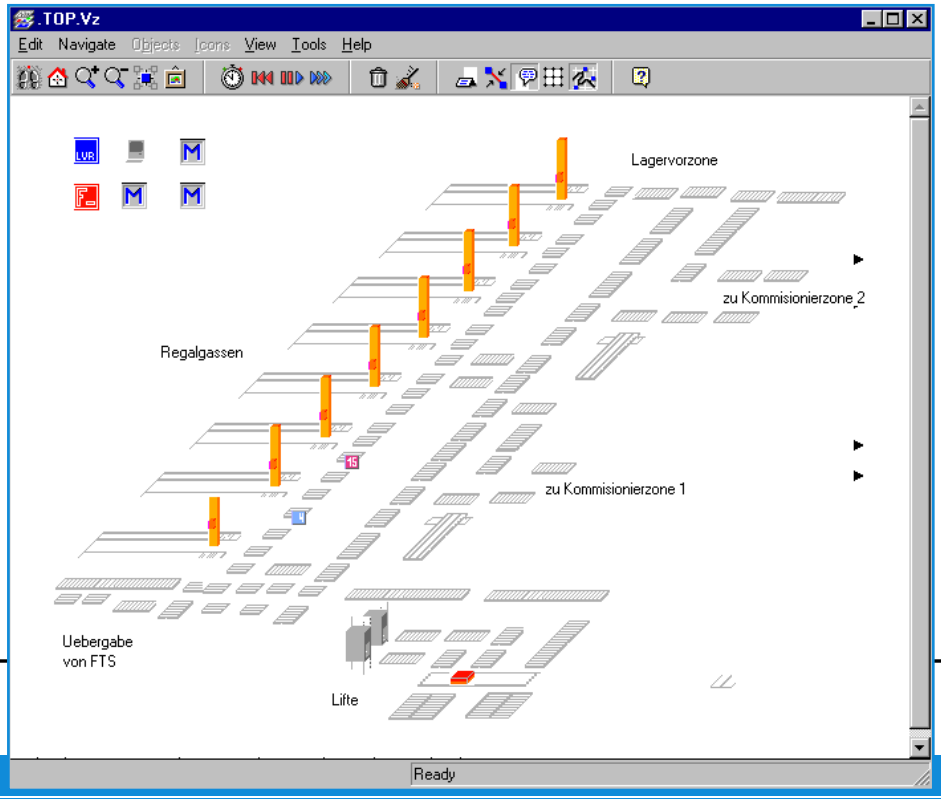
- The Line is an active material flow object.
- Use it to model conveyor systems.

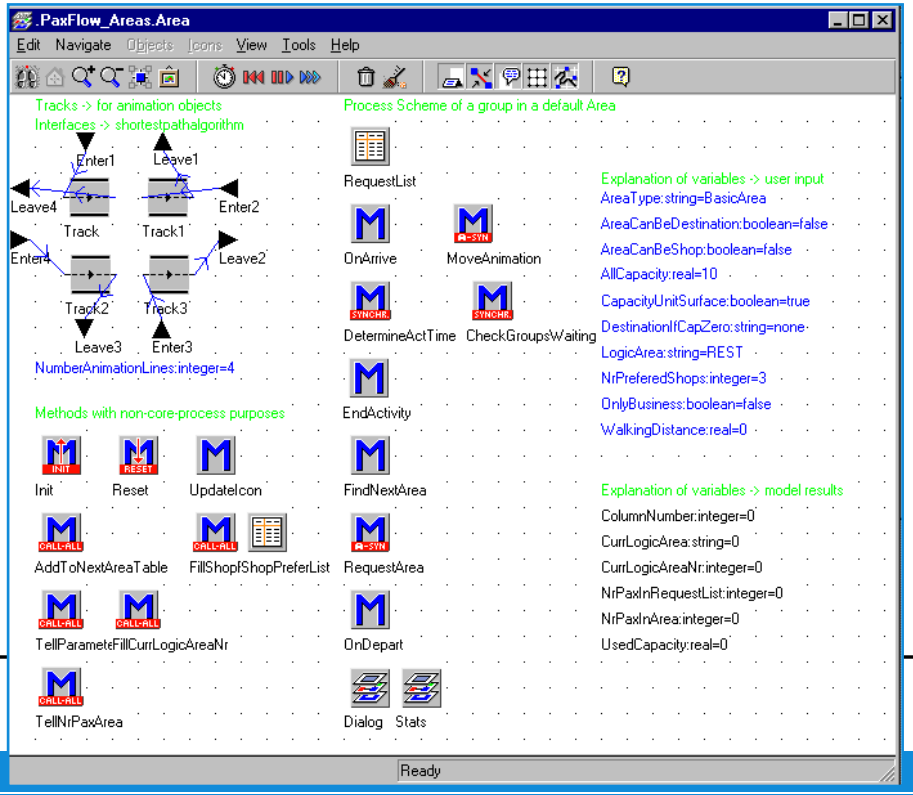


# Plant Simulation

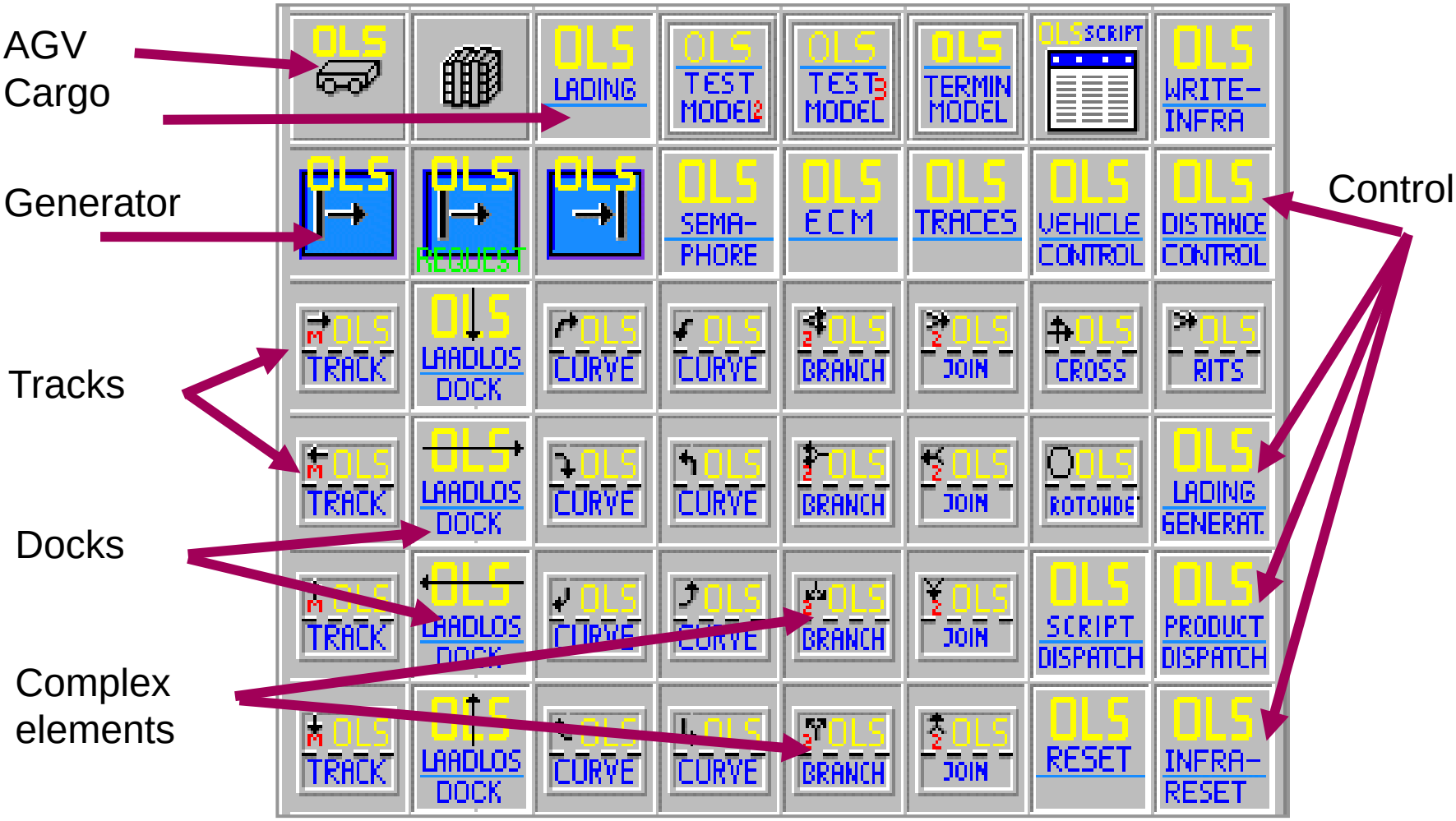
Sensors on the line -> trigger of Methods

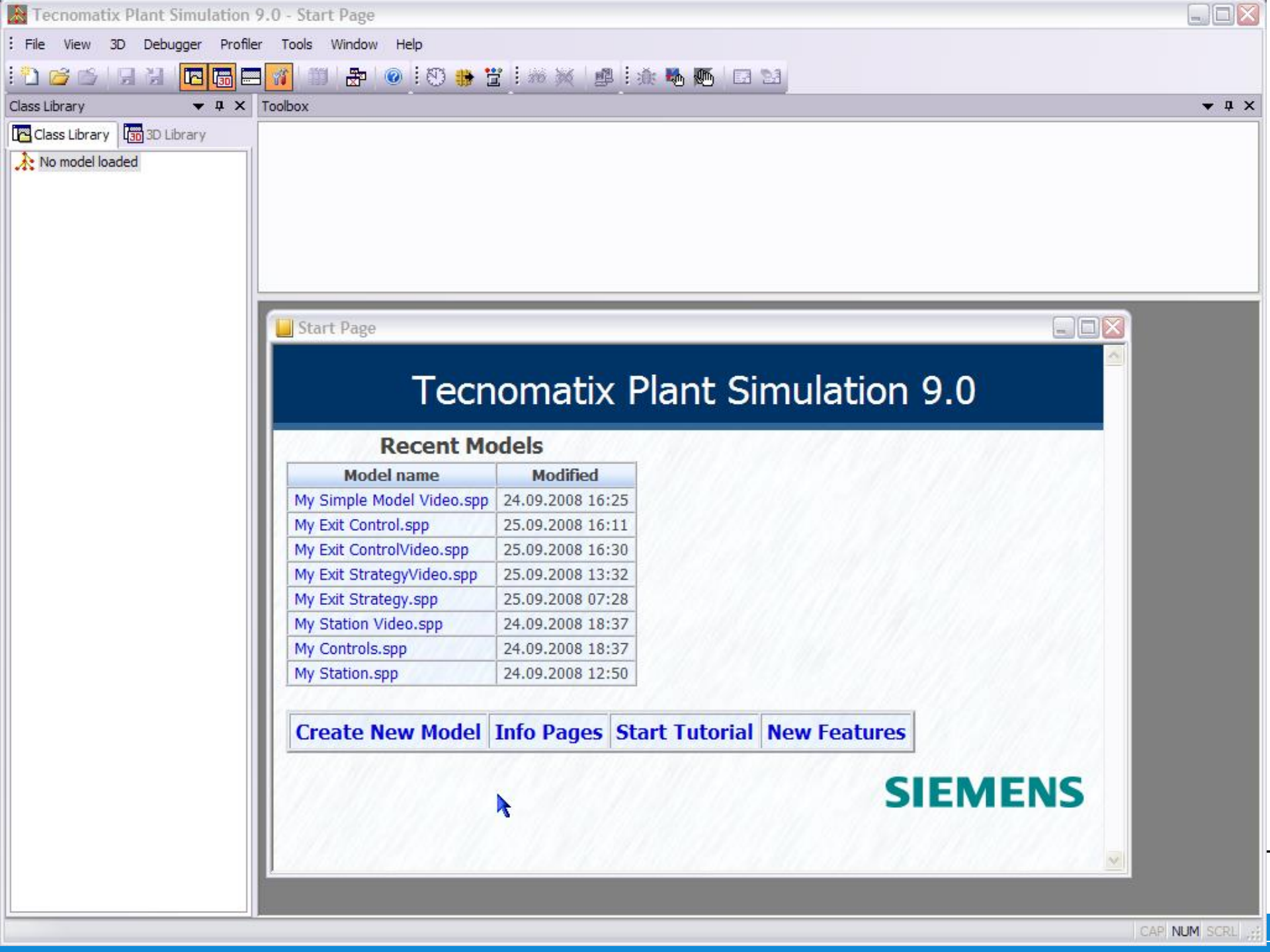


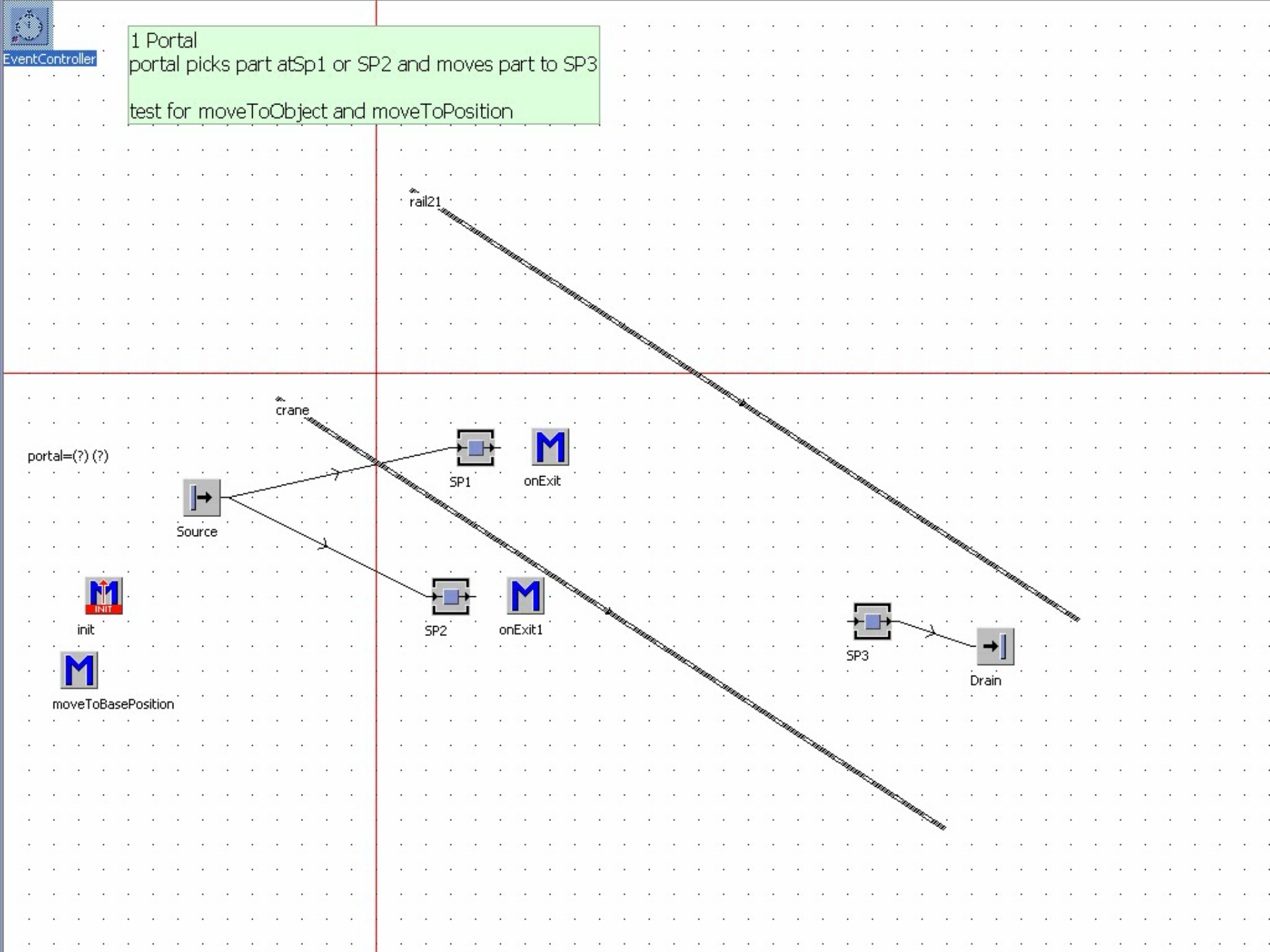



# Own libraries can be created




Tecnomatix Plant Simulation 9.0 - Start Page
File View 3D Debugger Profiler Tools Window Help
Class Library
Toolbox
Class Library
3D Library
No model loaded
Start Page
Tecnomatix Plant Simulation 9.0
Recent Models
Model name
Modified
My Simple Model Video.spp
24.09.2008 16:25
My Exit Control.spp
25.09.2008 16:11
My Exit ControlVideo.spp
25.09.2008 16:30
My Exit StrategyVideo.spp
25.09.2008 13:32
My Exit Strategy.spp
25.09.2008 07:28
My Station Video.spp
24.09.2008 18:37
My Controls.spp
24.09.2008 18:37
My Station.spp
24.09.2008 12:50
Create New Model
Info Pages
Start Tutorial
New Features
SIEMENS
CAP NUM SCRL


EventController
1 Portal
portal picks part atSp1 or SP2 and moves part to SP3
test for moveToObject and moveToPosition
rail21
crane
portal=(?) (?)
SP1
onExit
Source
init
SP2
onExit1
SP3
Drain
moveToBasePosition

# 6.

# AutoMod

# AutoMod Product Suite

- AutoMod
  - Model build and simulation execution environment
- AutoStat
  - Statistical analysis including optimizations
- AutoView
  - Animation with AVI/MPEG support
- Module Communications Module (MCM)
  - Linking to third party software and control systems

# AutoMod general information

- Modeling of:
  - Transportation, logistics
  - Material Handling Systems
- Development environment
  - Construction of models
  - Compile models
- Simulation environment
  - Run model, animation, gather statistics

# AutoMod

- True to scale
  - conveyors
  - transporters
- No room – no move
- "A meter is a meter"

# AutoMod: Systems

- Process System
  - Defines the model logic, how loads are processed in the system
  - Only one process system can be used
- Movement Systems
  - Simulation of material handling
  - Unlimited number of movement systems
- Static System
  - Background (wall, floor space)


# AutoMod: Process System

- Process
  - Define activities that take place in model
  - Loads flow from one process to another
  - Different types of processes
- Loads
  - Loads are the physical entities that move through a system
  - Loads execute logic and cause events to happen
  - Load type, Load creation rate
  - Load attributes: specific to each load


# AutoMod: Process System

- Resources
  - Represent machines, operators, tools
  - Limited capacity
  - Get-wait-free or use-command
  - Resource cycles (MTTF, MTTR)
- Queue
  - Space were loads van be stored
  - Use two queues: waiting and processing
- Order lists
  - A list of loads/vehicles that are delayed for a amount of time during the simulation

# AutoMod: Movement systems

- Movement system
  - Template for material movements
    - Path Mover (AGVs, lift trucks, trucks)
    - Conveyor
    - Automated Storage and Retrieval Systems
    - Bridge cranes
    - Power and free chain conveyors
    - Tanks and pipes