

Discrete Event (“Network”) Modeling in AnyLogic

Nathaniel Osgood

CMPT 858

March 10, 2011

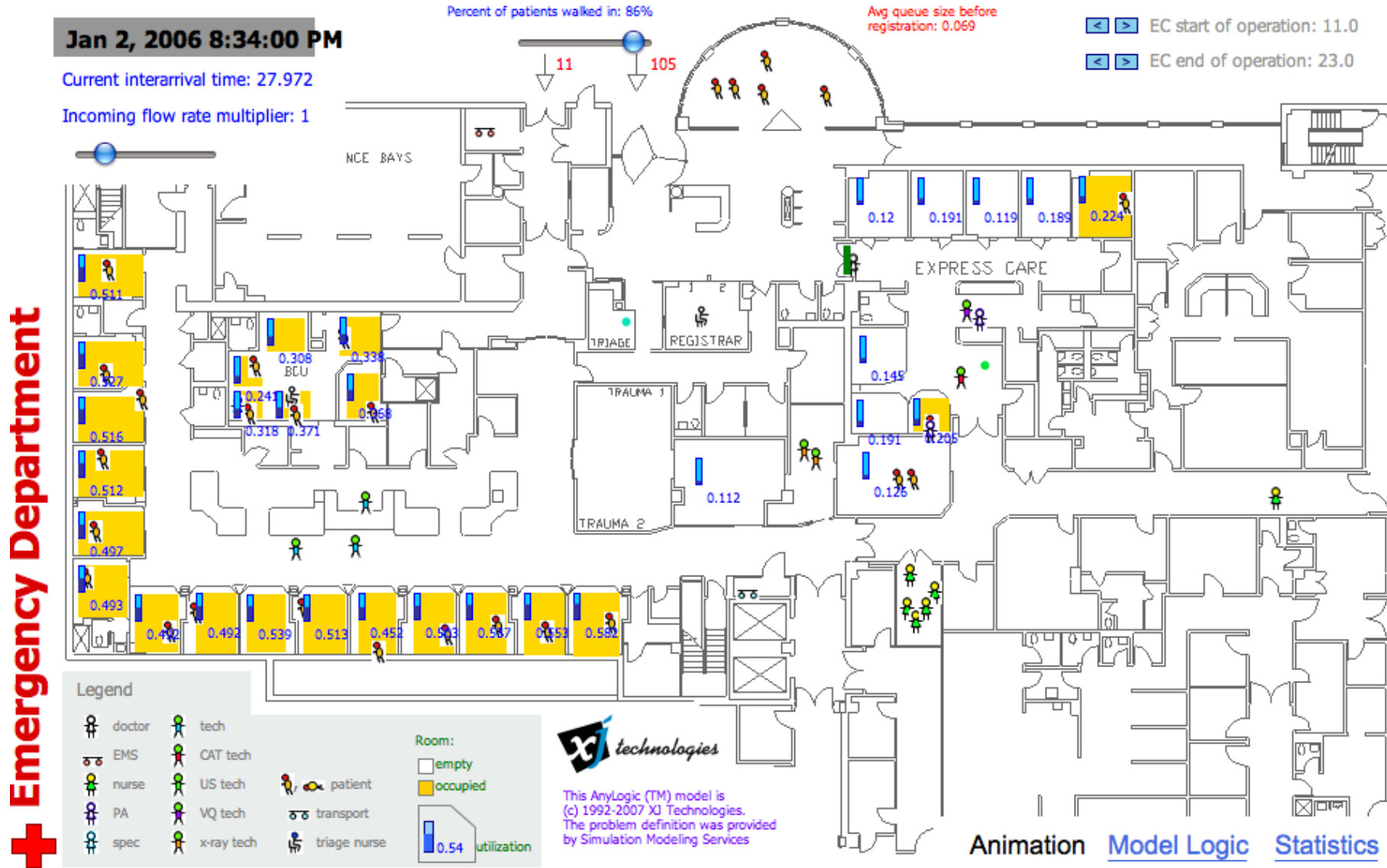


Hands on Model Use Ahead



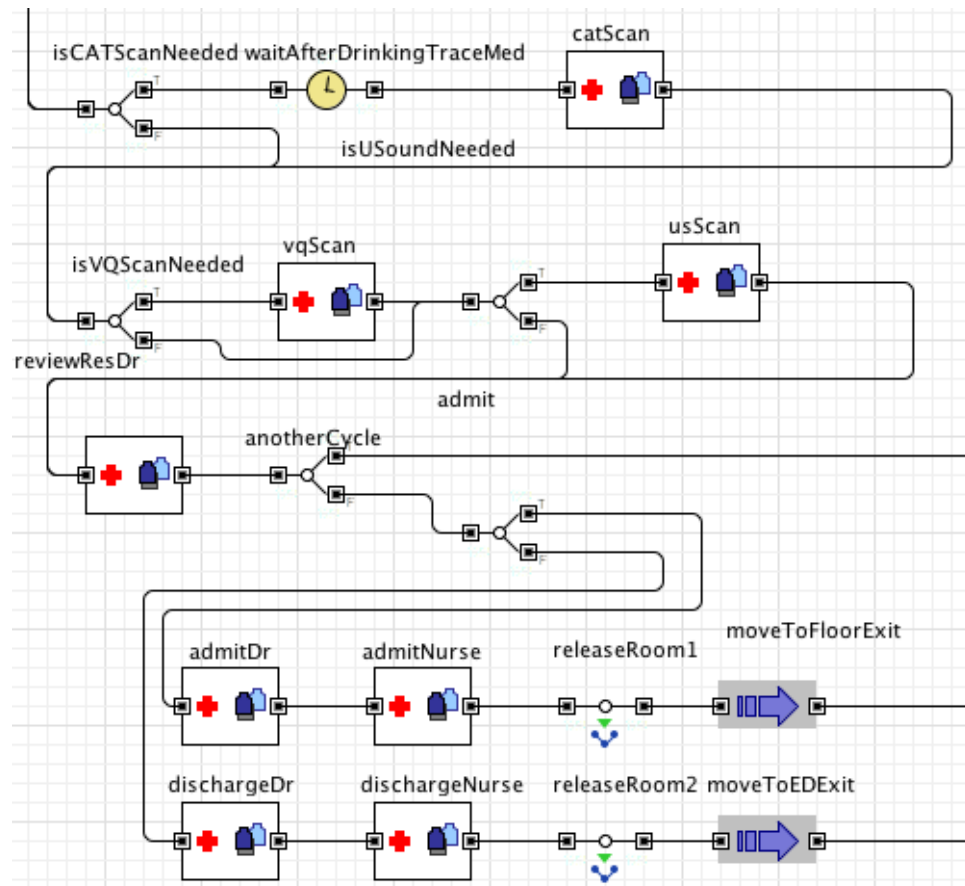
Load model:
Emergency Department Tulsa.alp

Recall: "Network Modeling" Irregular Spatial Embedding



Discrete Event Modeling

- Resource-based modeling
 - Queues
 - Processes
 - Flow charts
 - Capacitated resource pools
 - Send to
 - Attachment/detachment



Central Concepts in Discrete Event Modeling

- Entities flowing through processes & being processed at successive stages
- Flow charts guide entity progress
- Resources required for processing
 - Queues for waiting entities
- Capacitated resource pools from which resources are drawn
- Entity interaction with resources
 - Attachment/detachment
 - Seizing
- Physical “homes” for resources
- Movement paths (via polygons)

Entities

- Entities are the central parties on which the processes take place
 - Cf patients in a hospital or clinic
- Primarily passive – things happen “to them”
- “Flow through” (are routed around) the flow charts associated with the system
 - Only exist for the duration of time that are in the system
 - Are “injected” into the system
- Multiple entities can be in the system at one time
- If wish to maintain extra information on an entity, can “subclass” the Entity class
- Entities are often associated with a physical representation, which can travel around



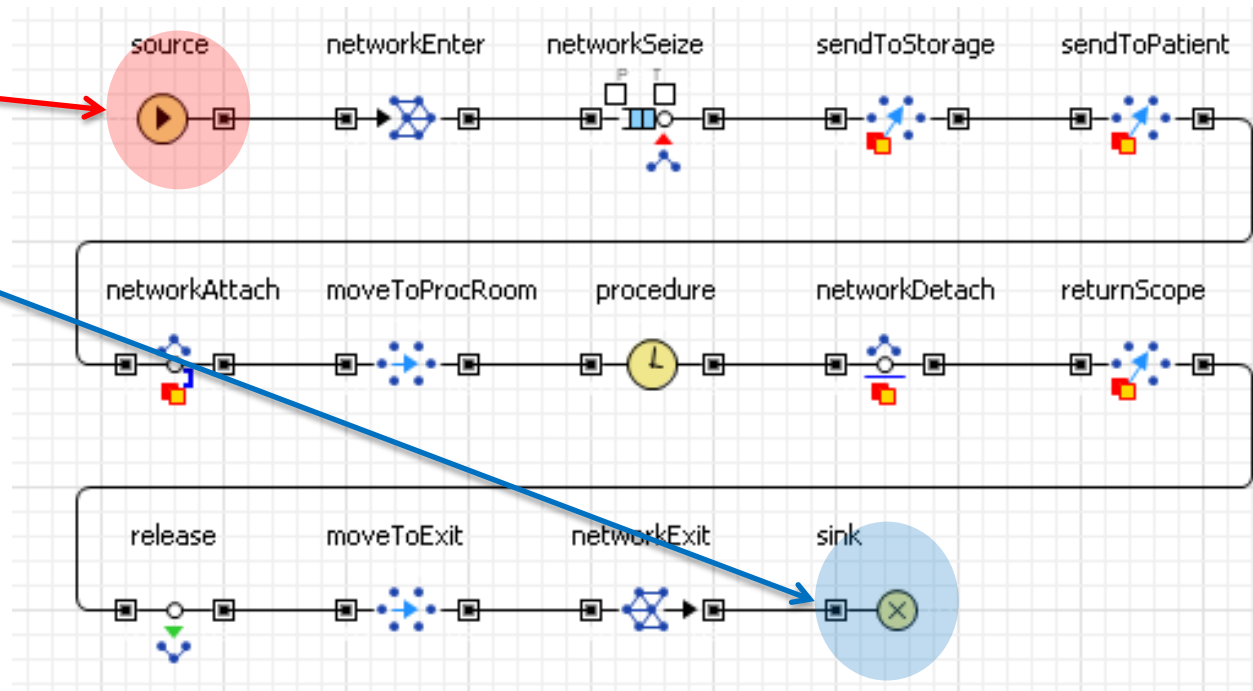
Hands on Model Use Ahead



Load model:
Ophthalmology Department.alp

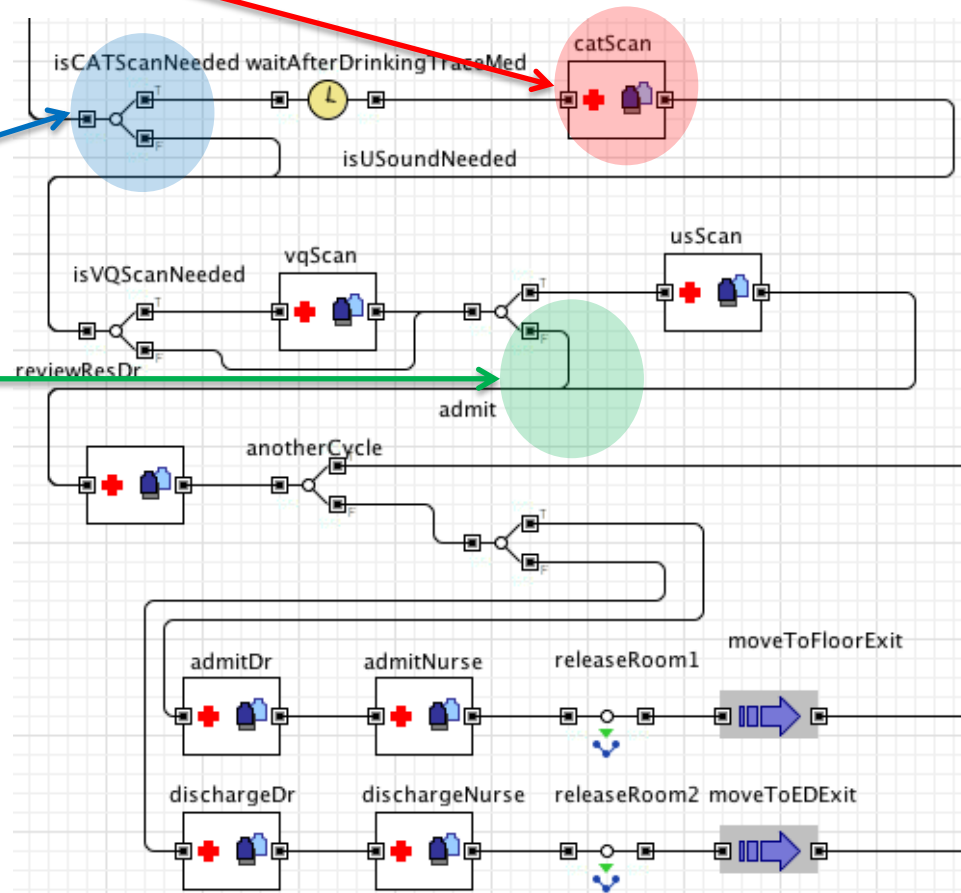
Flow Charts

- Entities flow in a single direction on flow charts
- **Start**
- **Finish**
- Can contain a variety of substeps



Flow Charts

- Flow charts can be **hierarchical**
- Frequently not linear e.g.
 - Branches
 - Joins



Elements to Build/Edit Flow Charts

The screenshot displays the AnyLogic Advanced software interface, titled "AnyLogic Advanced [EDUCATIONAL USE ONLY]". The main workspace shows a flow chart with several elements: a "release" element, a "moveToExit" element, a "networkExit" element, and a "sink" element. A "network" element is also visible, containing sub-elements "doctor", "procRoom", and "scope". The "doctor" element is highlighted with a blue box.

On the left side, there is a "Project" pane showing a tree structure of elements under "MainPhase1", including "Ports", "Embedded Objects", "source", "sink", "networkEnter", "networkExit", "network", "doctor", "procRoom", "scope", "procedure", "moveToExit", and "moveToProcRoom". Below this is a "Problems" pane with a table for "Description" and "Location".

On the right side, there is a "Palette" pane containing various elements categorized under "Model", "Action", "Analysis", "Presentation", and "Connectivity". The "Enterprise" category is expanded, showing elements like "Source", "Sink", "Hold", "Delay", "Queue", "Match", "Select Output", "Split", "Combine", "Resource Pool", "Seize", "Release", "Service", "Enter", "Exit", "Clock", "Conveyor", "Batch", "Unbatch", "Dropoff", "Pickup", "Restricted Ar...", "Network", and "Manual Enter".

At the bottom, there is a "Properties" pane for the selected "doctor" element, titled "doctor - NetworkResourcePool". The "General" tab is active, showing the following properties:

- Name: doctor
- Show Name:
- Ignore:
- Public:
- Show At Runtime:
- Create Presentation:
- Type: NetworkResourcePool<T extends ResourceUnit>
- Generic parameters: ResourceUnit
- Package: com.xj.anylogic.libraries.enterprise
- Resource type: Moving
- Capacity defined: Directly By home shape By table over time
- Capacity*: 5
- Speed: 10
- New resource unit: new ResourceUnit ()
- On new unit: (empty field)

Major Operators of Interest

- Source
- Sink
- Network enter/exit
(enter into a particular network)
- Select output (based on predicate)
- Split
- Delay
- Network move to
 - Move to a physical node or resource (see below)
- Resource-related (see following slides)
 - Network seize/release
 - Network attach/detach
 - Network send to

Determining factor can either be deterministic (e.g. based on condition) or stochastic (based on probability)

Select Output

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model for an Emergency Department. The model includes an 'EC Scenario ver. 1.2' block, an 'InitExam_PA' block, and a flow of actions: 'assAndPrep_PA_Tech', 'isXRayNeeded', 'seizeXRayRoom', 'moveToXRay', 'process', 'releaseXRayRoom', and 'moveFromXRay'. A blue arrow points from the title 'Select Output' to the 'isXRayNeeded' block in the model.

The 'isXRayNeeded - SelectOutput' configuration panel is open, showing the following settings:

- Name: isXRayNeeded
- Parameters: Show Name, Ignore, Public, Show At Runtime
- Type: SelectOutput<T extends Entity>
- Package: com.xj.anylogic.libraries.enterprise
- Select True output: If condition is true, With specified probability [0..1]
- Probability: $\frac{xrayPatients}{100.0}$
- On enter:
- On exit (true):
- On exit (false):

Network Delay

The screenshot displays the AnyLogic Advanced software interface for an educational use only. The main workspace shows a network simulation model. On the left, a tree view lists the model's structure, including 'Ophthalmology Department', 'MainPhase1', 'Ports', 'Embedded Objects', and various objects like 'source', 'sink', 'networkEnter', 'networkExit', 'network', 'doctor', 'procRoom', 'scope', 'procedure', 'moveToExit', and 'moveToProcRoom'. The main workspace contains a floor plan of a department and a network flow diagram. The flow starts at a 'source' (yellow circle), goes to 'networkEnter' (blue square), then to 'moveToProcRoom' (blue square with dots), then to a 'procedure' (yellow circle with a clock icon), then to 'moveToExit' (blue square with dots), 'networkExit' (blue square), and finally to a 'sink' (yellow circle). The 'procedure' is highlighted with a purple box. Below the workspace, the 'Properties' panel is open for the 'procedure - Delay' object. It shows the following settings:

- Name: procedure
- Show Name:
- Ignore:
- Public:
- Show At Runtime:
- Create Presentation:
- Type: Delay<T extends Entity>
- Generic parameters: Entity
- Package: com.xj.anylogic.libraries.enterprise
- Delay time is: Specified explicitly Path length / speed
- Delay time*: uniform(10)
- Capacity*: 5
- Maximum capacity:
- On enter:
- On exit:

The bottom status bar shows 'Selection'.

Resources

- Frequently resources are required to initiate a particular phase of processing
 - A doctor (resource) to administer surgery to a patient (entity)
 - A piece of diagnostic equipment (resource) to image a patient (entity)
 - An EKG to (resource) to record from a patient (entity)
 - A gurney or bed (resource) for a patient (entity)
- Distinctions amongst these resources
 - Portable vs. fixed
 - Mobile (with agency)

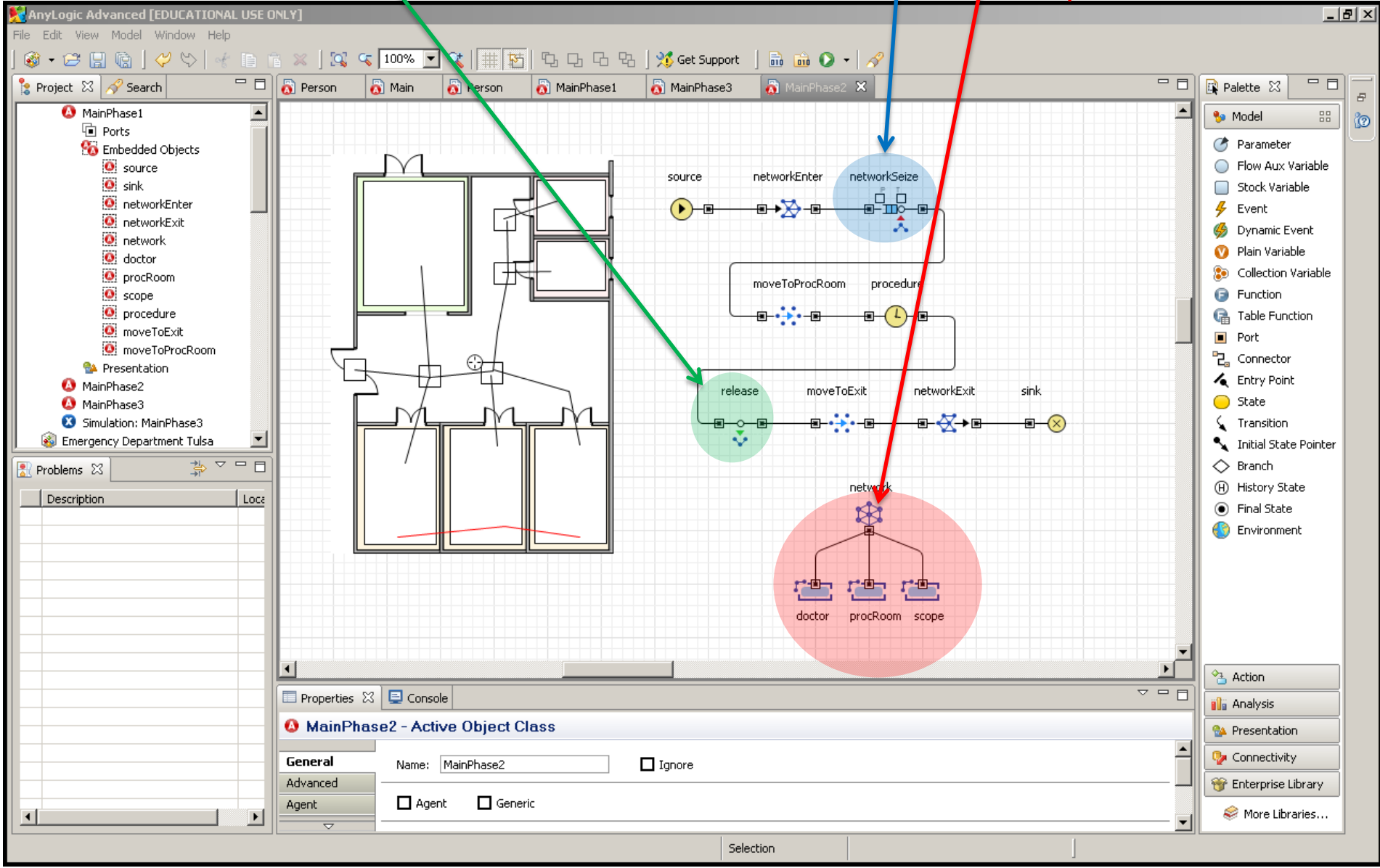
Resources 2

- A network is often associated with multiple types of resources
- When an agent cannot obtain (“seize”) a resource, they “enqueue” and wait for that resource to be released by another entity
 - These resources live in “pools” of interchangeable “resource units”
 - A “seized” resource comes from the pool
 - A “released” resource returns to the pool
 - If wish to be able to choose particular resources from a pool, create in *different pools*, and select desired pool

“Seizes” (seeks to achieve exclusive association with) a resource (otherwise waits)
“Releases” association with a resource, so others can be associated with/use it

Resources

Types of resources associated with the network, each in a resource pool



Main Flow Operators Associated with Resources

- All resources
 - Network Seize
 - Network Release
- Portable resources
 - Network Attach (NetworkAttach)/Detach (NetworkAttach)
- Mobile resources
 - Network SendTo (NetworkSendTo)

Defining Resource Pools

Capacity of Pool (number of units of resource present)

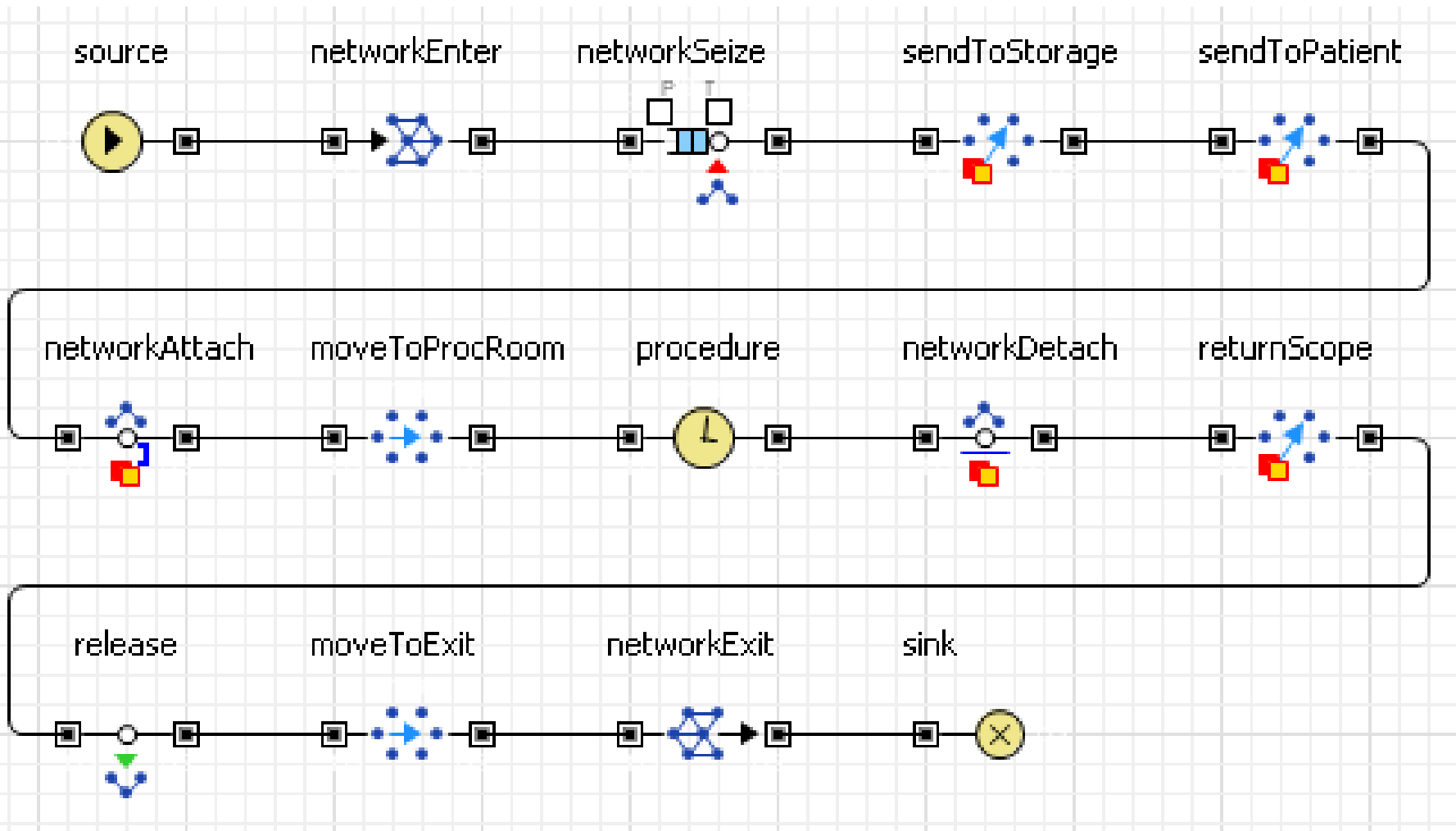
Resource pool type
(Static [Fixed], Moving [Mobile],
Portable [Can be carried])

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model with a network diagram and a flowchart. The Properties window for the 'doctor - NetworkResourcePool' is open, showing the following configuration:

- Name: doctor
- Type: NetworkResourcePool<T extends ResourceUnit>
- Package: com.xj.anylogic.libraries.enterprise
- Resource type: Moving (selected from a dropdown menu)
- Capacity defined: By home shape By table over time
- Capacity*: 5
- Speed: 10
- New resource unit: new ResourceUnit ()

A blue arrow points from the text 'Capacity of Pool' to the 'Capacity*' field in the Properties window. A red arrow points from the text 'Resource pool type' to the 'Resource type' dropdown menu in the Properties window.

Another Flow Chart



Source: Source of Entities

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model with a network diagram and a flowchart. The flowchart starts with a 'source' block, which is highlighted with a red box. The flowchart includes blocks for 'networkEnter', 'networkSeize', 'sendToStorage', 'sendToPatient', 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'. The 'source' block is connected to the 'networkEnter' block. The 'network' block is also visible in the diagram.

The Properties window for the 'source' block is open, showing the following settings:

- Name: source
- Show Name:
- Ignore:
- Public:
- Show At Runtime:
- Create Presentation:
- Type: Source <T extends Entity>
- Generic parameters: Entity
- Package: com.xj.anylogic.libraries.enterprise
- Arrivals defined by: Rate Interarrival time Rate table Arrival table Manual (call inject() method)
- Arrival rate*: 0.05
- Entities per arrival: 1
- Limited number of arrivals:
- New entity: new Entity()
- On exit:

A red arrow points from the text 'Defines rules governing origination of Entities to enter into network' to the 'Arrival rate*' field in the Properties window.

Defines rules governing origination of Entities to enter into network

Network Enter: Informing Newly Created Entities of the Available Resources

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model with a network diagram and a flowchart. The flowchart includes entities like `source`, `networkEnter`, `networkSeize`, `sendToStorage`, `sendToPatient`, `networkAttach`, `moveToProcRoom`, `procedure`, `networkDetach`, `returnScope`, `release`, `moveToExit`, `networkExit`, and `sink`. The `networkEnter` entity is highlighted with a blue box.

The Properties window for the selected `networkEnter` entity is shown below:

networkEnter - NetworkEnter

General

Name: `networkEnter` Show Name Ignore Public Show At Runtime

Parameters

Type: `NetworkEnter<T extends Entity>` Generic parameters: `Entity`

Package: `com.xj.anylogic.libraries.enterprise`

Network*: `network`

Entry node*: `waitingHall`

On enter:

Speed: `10`

Replication:

Securing Association with 1 or More Resources: Network Seize

AnyLogic Advanced [EDUCATIONAL USE ONLY]

File Edit View Model Window Help

Project Search

Ophthalmology Department*

- MainPhase1
 - Ports
 - Embedded Objects
 - source
 - sink
 - networkEnter
 - networkExit
 - network
 - doctor
 - procRoom
 - scope
 - procedure
 - moveToExit
 - moveToProcRoom
 - Presentation
 - MainPhase2
 - MainPhase3
 - Simulation: MainPhase3

networkSeize - NetworkSeize

General Name: networkSeize Show Name Ignore Public Show At Runtime Create Presentation

Parameters Type: NetworkSeize<T extends Entity> Generic parameters: Entity

Description Package: com.xj.anylogic.libraries.enterprise

List of resources { pool1, ... } {procRoom, doctor, scope}

On enter

On exit

Queue capacity

Maximum queue capacity

Enable exit on timeout

Enable preemption

Resource pools With whose Resources Entity is seeking association

This seizes one resource unit from each pool One resource may be seized while waiting for ("blocking for") the other.

Network Send To: Moving a (Seized) Resource to a Resource, Entity, or Place

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model with a network diagram and a flowchart. The flowchart includes nodes for 'source', 'networkEnter', 'networkSeize', 'sendToStorage', 'sendToPatient', 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'. The 'sendToStorage' node is highlighted with a blue box. The Properties window for 'sendToStorage - NetworkSendTo' is open, showing the following configuration:

- Name: sendToStorage
- Parameters: Type: NetworkSendTo<T extends Entity>, Package: com.xj.anylogic.libraries.enterprise
- Resources to send*: { doctor }
- Destination is*: Seized resource unit
- Resource*: scope

A red arrow points from the text 'Here, sending an already Seized resource to another resource' to the 'Resources to send*' field in the Properties window.

Here, sending an already Seized resource to another resource

Example of Simultaneously Moving Multiple Resources Together via SendTo

AnyLogic Advanced [EDUCATIONAL USE ONLY]

File Edit View Model Window Help

100%

Project Search

Ophthalmology Department

- MainPhase1
 - Ports
 - Embedded Objects
 - source
 - sink
 - networkEnter
 - networkExit
 - network
 - doctor
 - procRoom
 - scope
 - procedure
 - moveToExit
 - moveToProcRoom
 - Presentation
 - MainPhase2
 - MainPhase3
 - Simulation: MainPhase3

Person Main MainPhase1 MainPhase3 MainPhase2 ECProcess

source networkEnter networkSeize sendToStorage sendToPatient

networkAttach moveToProcRoom procedure networkDetach returnScope

release moveToExit networkExit sink

Palette

- Model
- Action
- Analysis
- Presentation
- Connectivity
- Enterprise...

Source Sink Hold Delay Queue Match Select Output Split Combine Resource Pool Seize Release Service Enter Exit Clock Conveyor Batch Unbatch Dropoff Pickup Restricted Area... Restricted Area... Network Manual Entry More Libraries...

Problems

Description	Location
-------------	----------

sendToPatient - NetworkSendTo

General Name: sendToPatient Show Name Ignore Public Show At Runtime Create Presentation

Parameters Type: NetworkSendTo<T extends Entity> Generic parameters: Entity

Description Package: com.xj.anylogic.libraries.enterprise

Resources to send: {doctor, scope}

Destination is* Specified node Entity Seized resource unit Home of seized resource unit

On enter:

On exit:

Replication:

Here, sending 2 already Seized resource s to the entity (the patient)

Network Attach: Associating Entity with Specified Seized Resources, or those Nearby (So move together henceforth)

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a network diagram with several interconnected blocks. A purple box highlights the `networkAttach` block, which is connected to a `networkEnter` block and a `networkExit` block. The diagram also includes blocks for `source`, `networkSeize`, `sendToStorage`, `sendToPatient`, `moveToProcRoom`, `procedure`, `networkDetach`, `returnScope`, `release`, `moveToExit`, `networkExit`, and `sink`.

The left sidebar shows the project structure, including the Ophthalmology Department, MainPhase1, Ports, Embedded Objects, and Presentation. The bottom panel shows the configuration for the `networkAttach` block.

networkAttach - NetworkAttach

General

Name: `networkAttach` Show Name Ignore Public Show At Runtime

Parameters

Type: `NetworkAttach<T extends Entity>` Generic parameters: `Entity`

Package: `com.xj.anylogic.libraries.enterprise`

Attach* All seized non-static resources at entity location Specified resources

On enter

On exit

Replication:

The right sidebar shows the Palette with various blocks and actions, including Source, Sink, Hold, Delay, Queue, Match, Select Output, Split, Combine, Resource Pool, Seize, Release, Service, Enter, Exit, Clock, Conveyor, Batch, Unbatch, Dropoff, Pickup, Restricted Area, Network, and Network Enter.

Network Move To: Moving an *Entity* to a Resource (or Node)

The screenshot displays the AnyLogic Advanced interface. On the left, a project tree shows the 'Ophthalmology Department*' model with phases MainPhase1, MainPhase2, and MainPhase3. The main workspace contains a network diagram with nodes like 'source', 'networkEnter', 'networkSeize', 'sendToStorage', 'sendToPatient', 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'. A blue box highlights the 'moveToProcRoom' node. Below the diagram, the 'Properties' window for 'moveToProcRoom - NetworkMoveTo' is open. It shows the following configuration:

- Name: moveToProcRoom
- Parameters: Show Name, Ignore, Public, Show At Runtime
- Type: NetworkMoveTo<T extends Entity>
- Generic parameters: Entity
- Package: com.xj.anylogic.libraries.enterprise
- Destination is*: Specified node Seized resource unit
- Resource*: procRoom

Red annotations are present:

- A red arrow points from the text 'Resource to which agent should move (here, already seized unit from this Resource pool)' to the 'procRoom' value in the 'Resource*' field.
- Red text at the bottom reads: 'NB: Because resources are attached (seized), this MoveTo will Move Entity & but also bring moving & portable resources along (doctor & scope)'.

Network Detach

So **entity** can be physically Separated from **resources** (while remaining associated w/them)

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model of an ophthalmology department, including a network diagram with various components like 'source', 'networkEnter', 'networkSeize', 'sendToStorage', 'sendToPatient', 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'. The 'networkDetach' component is highlighted with a blue box.

The left sidebar shows the project structure, including 'Ophthalmology Department', 'MainPhase1', 'Ports', 'Embedded Objects', and 'Presentation'. The bottom-left pane shows the 'Problems' list.

The bottom-right pane shows the 'Properties' window for the 'networkDetach' component. The 'General' tab is active, showing the following details:

- Name: networkDetach
- Parameters: Show Name, Ignore, Public, Show At Runtime, Create Presentation
- Type: NetworkDetach<T extends Entity>
- Generic parameters: Entity
- Package: com.xj.anylogic.libraries.enterprise
- Detach*: All attached resources Specified resources
- On enter: [Empty text field]
- On exit: [Empty text field]
- Replication: [Empty text field]

The right sidebar shows the 'Palette' with various simulation components like Model, Action, Analysis, Presentation, Connectivity, and Enterprise... The 'Enterprise...' palette includes components like Source, Sink, Hold, Delay, Queue, Match, Select Output, Split, Combine, Resource Pool, Seize, Release, Service, Enter, Exit, Clock, Conveyor, Batch, Unbatch, Dropoff, Pickup, Restricted Area, and Network.

Releasing Associated Resources

Resource pools whose Resource Units one is releasing

release - NetworkRelease

General

Name: Show Name Ignore Public Show At Runtime

Parameters

Type: Generic parameters:

Description

Package:

Release: Specified resources All seized resources

Moving resources: Return to home location Stay where they are

On enter:

On exit:

Replication:

Moving resource (doctor) returns to home location after release of association with entity

Visual Depiction

- Entities are associated with icons
- Resources are associated with
 - Locations
 - Icons
- Networks are associated with routing paths
 - Often want to move resources or icons among different visual locations
 - Specific points (e.g. a storage closet for mobile resources)
 - Points associated with fixed resources (e.g. a MRI scanner)

Association of Network with Paths

The screenshot displays the AnyLogic Advanced software interface, titled "AnyLogic Advanced [EDUCATIONAL USE ONLY]". The main workspace is divided into several panes:

- Project Pane (Left):** Shows a hierarchical tree structure for the "Ophthalmology Department*" model, including "MainPhase1", "Ports", "Embedded Objects", and various objects like "source", "sink", "networkEnter", "networkExit", "network", "doctor", "procRoom", "scope", "procedure", "moveToExit", and "moveToProcRoom".
- Main Canvas (Center):** Contains a network diagram. On the left, a schematic of three examination rooms is shown with a red line indicating a path. On the right, a flowchart shows the network's logic, including actions like "networkAttach", "moveToProcRoom", "procedure", "networkDetach", "returnScope", "release", "moveToExit", "networkExit", and "sink". Below the flowchart, a "network" node is connected to "doctor", "procRoom", and "scope" nodes.
- Properties Pane (Bottom):** Displays the configuration for the selected "network - Network" object. The "General" tab is active, showing:
 - Name: network
 - Type: Network
 - Package: com.xj.anylogic.libraries.enterprise
 - Group of network shapes*: networkGroup
 - Hide network shapes:
 - When item is at a node: Draw at random position within node
 - Enable priorities:
 - Replication: (empty field)
- Palette (Right):** A vertical toolbar containing various modeling elements such as Source, Sink, Hold, Delay, Queue, Match, Select Output, Split, Combine, Resource Pool, Seize, Release, Service, Enter, Exit, Clock, Conveyor, Batch, Unbatch, Dropoff, Pickup, Restricted Area, and Network.

Associated “Group” of Presentation

The network will “know” about these
(e.g. forrouting)

The screenshot displays the AnyLogic Advanced software interface, which is used for building and simulating discrete event models. The main workspace is divided into several panes:

- Project Explorer (Left):** Shows a hierarchical tree of the model's components. Under the 'Presentation' folder, a 'networkGroup' is defined, containing various elements like 'waitingHall', 'staffRoom', 'storageRoom', 'procRoom1', 'procRoom2', 'exit', and several 'rectangle' and 'polyline' objects.
- Main Canvas (Center):** Displays a 2D simulation environment. It features a floor plan with three rooms. A network diagram is overlaid on the canvas, showing a central 'network' node connected to three nodes labeled 'doctor', 'procRoom', and 'scene'. Above the floor plan, a flowchart of the simulation logic is visible, including steps like 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'.
- Properties Panel (Bottom):** Shows the configuration for the selected 'networkGroup'. The 'General' tab is active, displaying the name 'networkGroup' and several checkboxes: 'Show Name' (unchecked), 'Ignore' (unchecked), 'Public' (checked), and 'Icon' (unchecked).
- Palette (Right):** A vertical toolbar containing various simulation components such as 'Source', 'Sink', 'Hold', 'Delay', 'Queue', 'Match', 'Select Output', 'Split', 'Combine', 'Resource Pool', 'Seize', 'Release', 'Service', 'Enter', 'Exit', 'Clock', 'Conveyor', 'Batch', 'Unbatch', 'Dropoff', 'Pickup', 'Restricted Area', 'Network', and 'Network Enter'.

Presentation of Entity

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model with a network of nodes and transitions. A red box highlights the 'source' entity in the network. The Properties window is open, showing the configuration for the 'source' entity. The 'Entity animation shape*' property is set to 'shapePatient', which is circled in red. A red arrow points from the text 'Appearance of the entity when Moving through the network' to this property.

Appearance of the entity when Moving through the network

Properties of the 'source' entity:

- Name: source
- Type: Source<T extends Entity>
- Package: com.xj.anylogic.libraries.enterprise
- Arrivals defined by: Rate
- Arrival rate*: 0.05
- Entities per arrival: 1
- Limited number of arrivals:
- New entity: new Entity()
- On exit:
- Entity animation shape*: shapePatient
- Unique shape for each entity:
- Enable rotation:

Presentation Properties of a Resource

The screenshot displays the AnyLogic Advanced interface. On the left, a project tree shows a resource named 'doctor' under a 'network' group. The main workspace shows a network diagram with nodes for 'doctor', 'procRoom', and 'scope'. A red arrow points from the 'doctor' node in the network to the 'doctor - NetworkResourcePool' properties window. In this window, the 'Idle unit animation shape*' and 'Busy unit animation shape*' are both set to 'shapeDoctor', which is circled in red. The 'Home node*' is set to 'staffRoom', which is circled in blue. A blue arrow points from the 'staffRoom' in the project tree to the 'Home node*' field in the properties window.

Appearance Of the resource units in this resource in Idle & Busy States

Home position of resource in Presentation network associated with network

doctor - NetworkResourcePool

- General
- Parameters
- Statistics
- Description
 - On seize
 - On release
 - Idle unit animation shape* **shapeDoctor**
 - Busy unit animation shape* **shapeDoctor**
 - Unique shape for each unit
 - Enable rotation
 - Home defined by: Single node
 - Home node* **staffRoom**
 - Enable statistics

Entering the Network: Where & with What (Logical & Presentation) Network

The screenshot displays the AnyLogic Advanced software interface. The main workspace is divided into two parts: a logical network diagram on the left and a presentation diagram on the right. The logical diagram shows a building layout with various rooms and a network node. The presentation diagram shows a flowchart with nodes like 'source', 'networkEnter', 'networkSeize', 'sendToStorage', 'sendToPatient', 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'. A red arrow points from the 'Speed' property in the configuration window to the 'networkEnter' node in the presentation diagram.

networkEnter - NetworkEnter

General

Name: networkEnter Show Name Ignore Public Show At Runtime

Parameters

Type: NetworkEnter <T extends Entity> Generic parameters: Entity

Package: com.xj.anylogic.libraries.enterprise

Network*: network

^Entry node*: waitingHall

^On enter:

^Speed: 10

Replication:

Speed to use when Entity moves around

Movement Network: Defined by Polygons & Rectangles

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a floor plan of an Emergency Department with a blue-shaded area representing a movement network. A logical network diagram is overlaid on the floor plan, showing a sequence of actions: source, networkEnter, networkSeize, sendToStorage, sendToPatient, networkAttach, moveToProcRoom, procedure, networkDetach, returnScope, release, moveToExit, networkExit, and sink. The project tree on the left lists various elements, including a networkGroup and several polylines. The palette on the right contains various modeling elements like Source, Sink, Hold, Delay, Queue, Match, Select Output, Split, Combine, Resource Pool, Seize, Release, Service, Enter, Exit, Clock, Conveyor, Batch, Unbatch, Dropoff, Pickup, Restricted Area, and Network.

Recall: This is the visual (presentation) network associated with the logical network

These "polylines" and rectangles are the elements over which the Entities & Resources move

Recall: The Location of the Rooms is Given as Being the “Path across nodes” defined by the Polyline

The screenshot shows the AnyLogic Advanced software interface. The main workspace displays a simulation model of an emergency department. The model includes a network diagram with nodes for 'doctor', 'procRoom', and 'scope', connected by a network resource pool. A red polyline is drawn across the room layout in the background. The Properties panel for 'procRoom - NetworkResourcePool' is open, showing the 'Home path*' property set to 'roomsLocation'.

procRoom - NetworkResourcePool

Property	Value
On seize	
On release	
Idle unit animation shape	
Busy unit animation shape	
Unique shape for each unit	<input type="checkbox"/>
Enable rotation	<input type="checkbox"/>
Home defined by*	Path across nodes
Home path*	roomsLocation
Enable statistics	<input type="checkbox"/>

Polyline Describes the Location of the Procedure Rooms

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a floor plan of an emergency department with a red-outlined polygon highlighting a specific area. A red arrow points from a text box to the vertices of this polygon. The text box contains the following text:

The rectangles touched by this poly ine vertices are the room locations

The properties panel at the bottom shows the following details for the 'roomsLocation - Polyline' object:

- Name: roomsLocation
- General: Show Name, Ignore, Public, Icon
- Advanced: Fill Color: No Fill
- Dynamic: Description: (empty)
- Line Color: red
- Line Width: 1 pt
- Line Style: (solid line)
- Polyline options: Close Polyline

Moving Entity to a Node

AnyLogic Advanced [EDUCATIONAL USE ONLY]

File Edit View Model Window Help

100%

Project Search

Ophthalmology Department

- MainPhase1
 - Ports
 - Embedded Objects
 - source
 - sink
 - networkEnter
 - networkExit
 - network
 - doctor
 - procRoom
 - scope
 - procedure
 - moveToExit
 - moveToProcRoom
 - Presentation
- MainPhase2
- MainPhase3
- Simulation: MainPhase3

Person Main Person MainPhase1 MainPhase3 MainPhase2 ECProcess

source networkEnter networkSeize sendToStorage sendToPatient

networkAttach moveToProcRoom procedure networkDetach returnScope

release moveToExit networkExit sink

Properties Console

moveToExit - NetworkMoveTo

General Name: moveToExit Show Name Ignore Public Show At Runtime

Parameters

Type: NetworkMoveTo<T extends Entity> Generic parameters: Entity

Package: com.xj.anylogic.libraries.enterprise

Destination is Specified node Seized resource unit

[#]Node*: exit

[#]On enter

[#]On exit

Replication:

Selection

Palette

- Model
- Action
- Analysis
- Presentation
- Connectivity
- Enterprise...
- Source
- Sink
- Hold
- Delay
- Queue
- Match
- Select Output
- Split
- Combine
- Resource Pool
- Seize
- Release
- Service
- Enter
- Exit
- Clock
- Conveyor
- Batch
- Unbatch
- Dropoff
- Pickup
- Restricted Are...
- Restricted Are...
- Network
- Network Enter
- More Libraries...

Subclassing: A Valuable Tool

- So as to customized the desired system behavior, it can be useful to customize entities & resources (resource units)
 - To e.g. carry around additional information (e.g. associated external agent in agent-based model, history information, etc.)
 - Particular specialized network types
- Because the original entities & resource units are classes, this can be accomplished via subclassing (subclass Entity & ResourceUnit)
- If do this, parameterize generics by subclass type S (e.g. NetworkResourcePool<S>)