

An Initial Glimpse of AnyLogic & Emergence: Modifying an Existing Model

Nathaniel Osgood
MIT 15.879

February 29, 2012

Announcements

- Office hours
 - Extra this week (Wednesday & Friday 10-11am)
 - Next week: Wednesday 12-1pm
- Please alert me to any AnyLogic installation/licensing issues ASAP
- First Friday class this week (in E62-446)
- Dates to watch
 - March 2: Assignment 1
 - March 9: Term project topic proposal
- Seeking expressions of interest in tutorials (email best)

Recall: Agent-Based Models

- Characteristics
 - One or more populations composed of individual agents
 - Each agent is associated with some of the following
 - State (continuous or discrete e.g. age, health, smoking status, networks, beliefs)
 - Parameters (e.g. Gender, genetic composition, preference fn.)
 - Rules for interaction (traditionally specified in general purpose programming language)
 - Embedded in an environment (typically with localized perception)
 - Communicate via messaging and/or flows
 - Local & Global Environments
- Emergent aggregate behavior

Recall: Organization in ABM

- ABM adopts the organizational style of object-oriented software engineering by clustering together the elements of state & behavior for entities
- This facilitates convenient representation of
 - Nested relationships (individuals in neighborhoods in municipalities, etc.)
 - Networked relationships (e.g. network of individuals, towns, farms, firms, etc.)

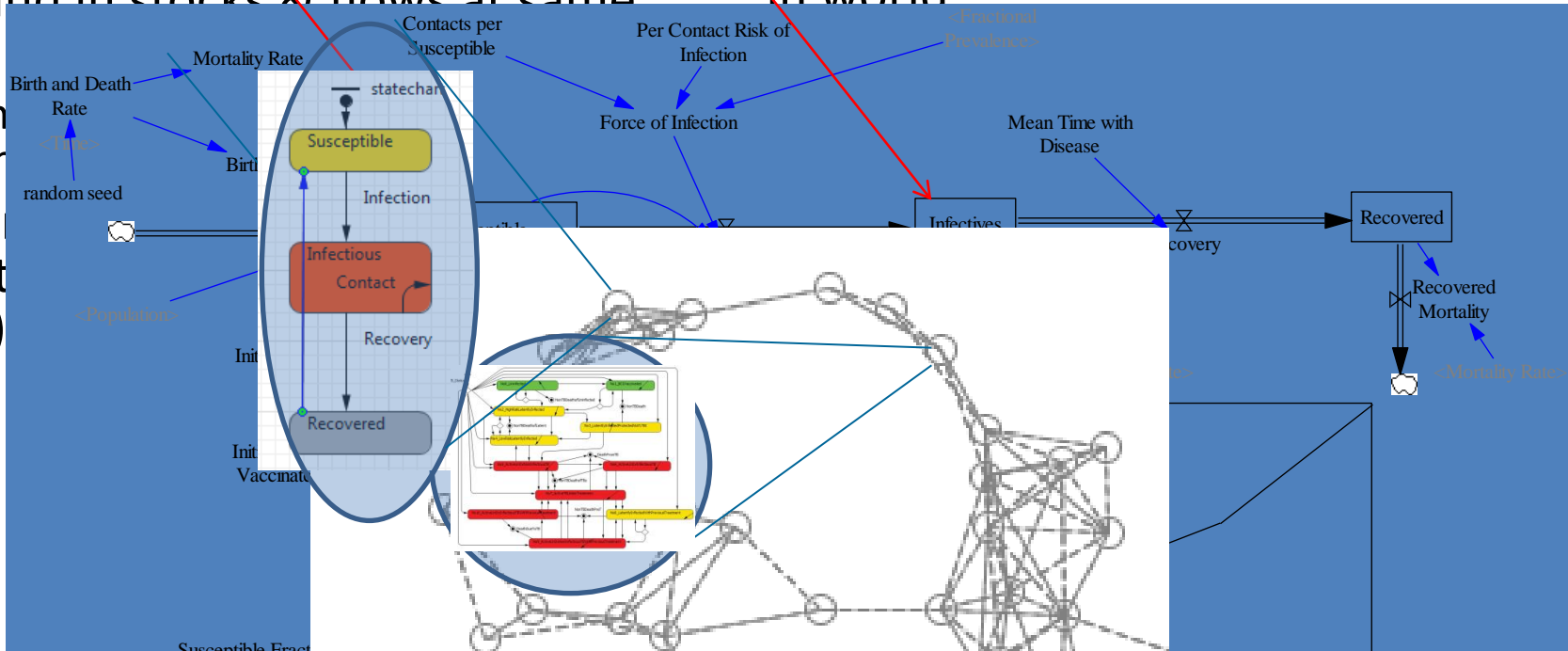
Contrasting Organization in Aggregate Stock-Flow & ABM

Agent-based modeling

Aggregate Stock & flow models •

- Within unit (e.g. city)
 - Subdivided according to state (eg # susceptible, # infective)
 - Each stock counts # units associated with that state
- State for different units of analysis are found in stocks & flows at same “level”
 - Sum both

- Within unit (e.g. city)
 - Subdivided according to constitutive smaller units (e.g. individual people)
 - Each unit maintains its own state
- The nested or networked relations among units of analysis mimic that in world



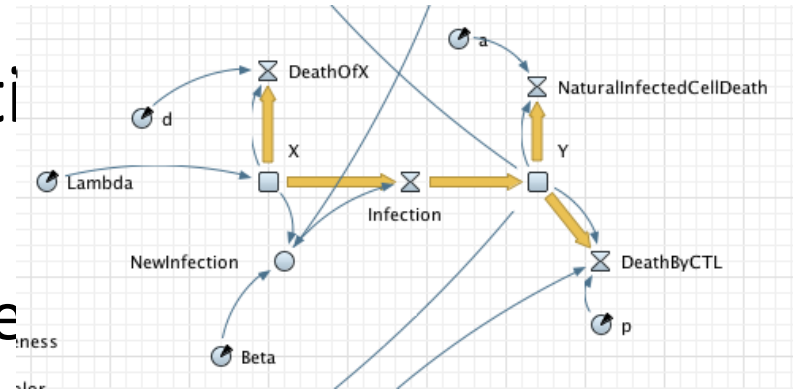
- Relatio implicit matrix)

AnyLogic basics

- Multi-platform
- Declarative graphical languages
- Basic language: Java
- Rich library of built-in objects
- Continuous or discrete time/space
- Modeling approaches supported
 - System Dynamics
 - Agent-based
 - Regular & irregular spatial embedding, network embedding
 - Discrete event

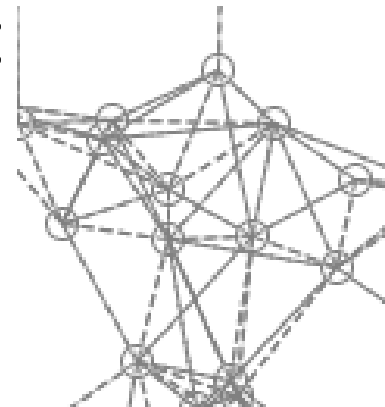
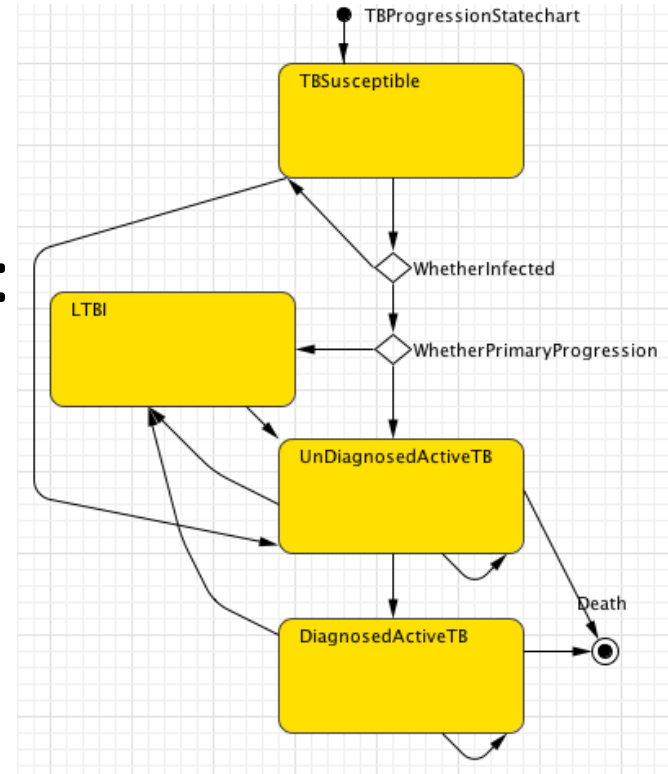
Stock & Flow Models

- Feedback-focus
- Traditional graphical depiction
 - Stocks (state of system)
 - Flows (rates of change to the
 - Continuous variation in state
- Stocks are initialized, are then change according to flows
- Values of flows are determined by stocks & any other variables

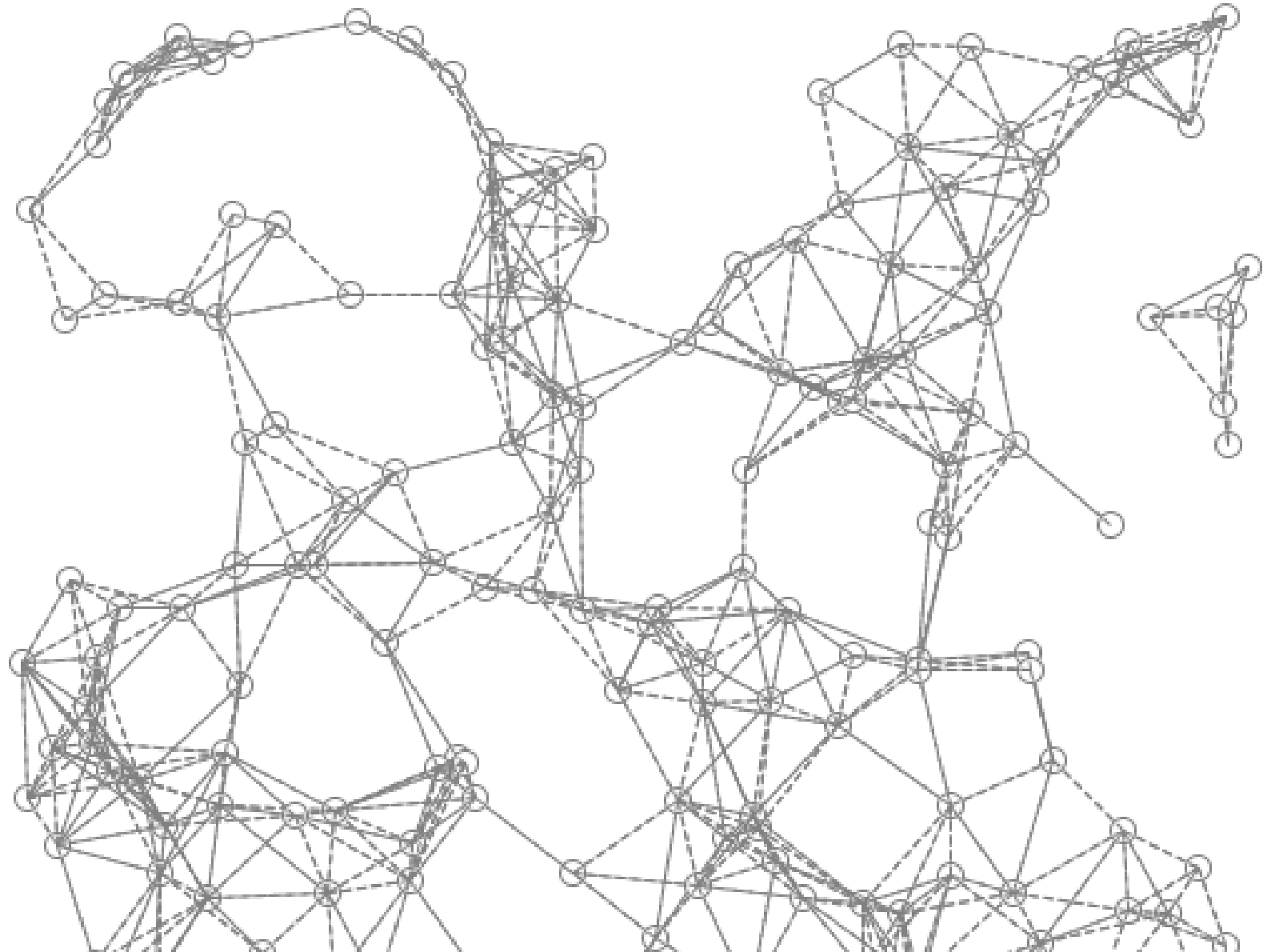


Agent-Based Modeling

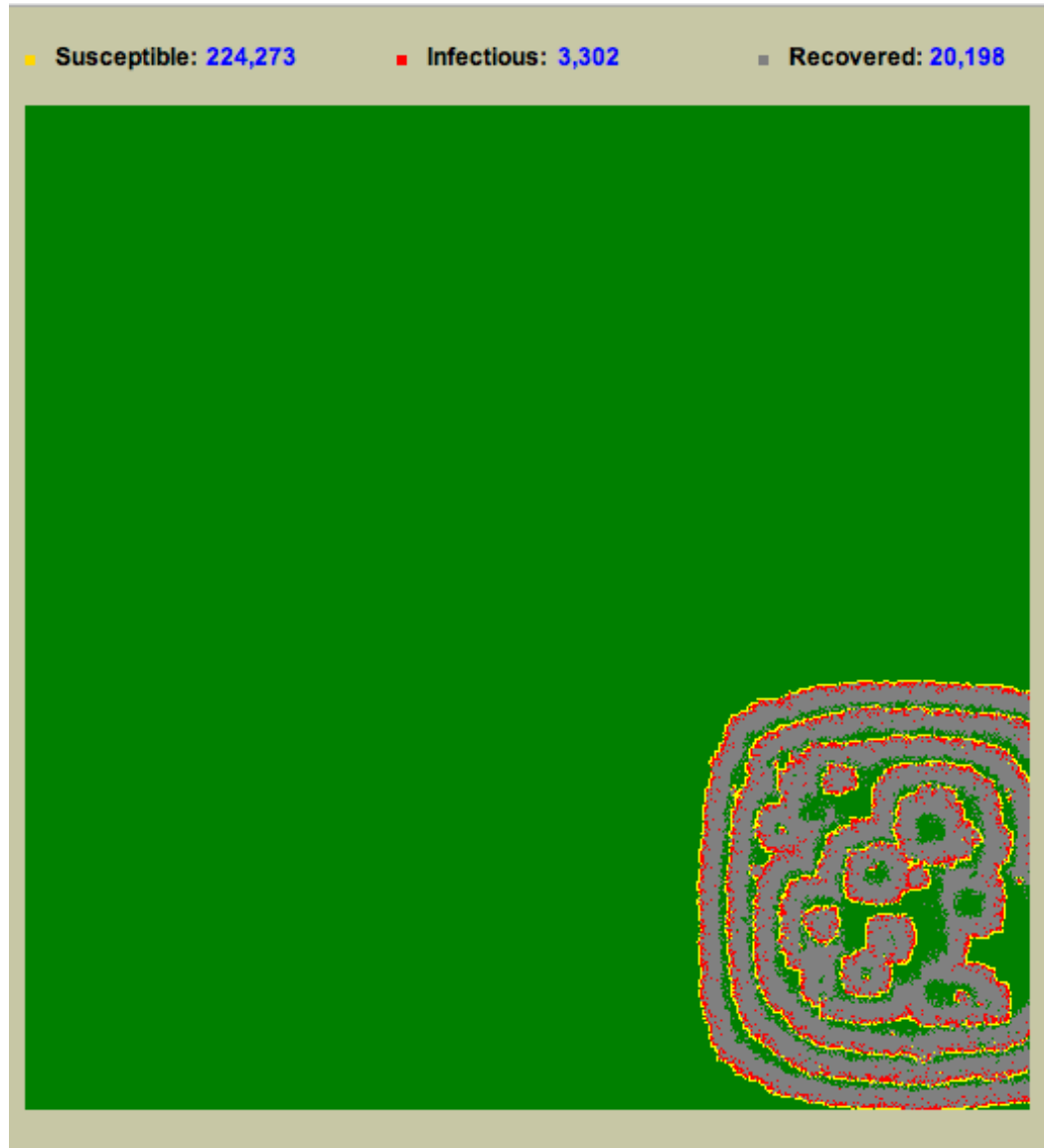
- Agent (actor) focused
- Traditional graphical depiction:
State transition diagram
 - States
 - Transitions
 - Discrete variation in state
- Regular or irregular topologies connect between agents
 - Messages sent via connections



Network Embedded Individuals

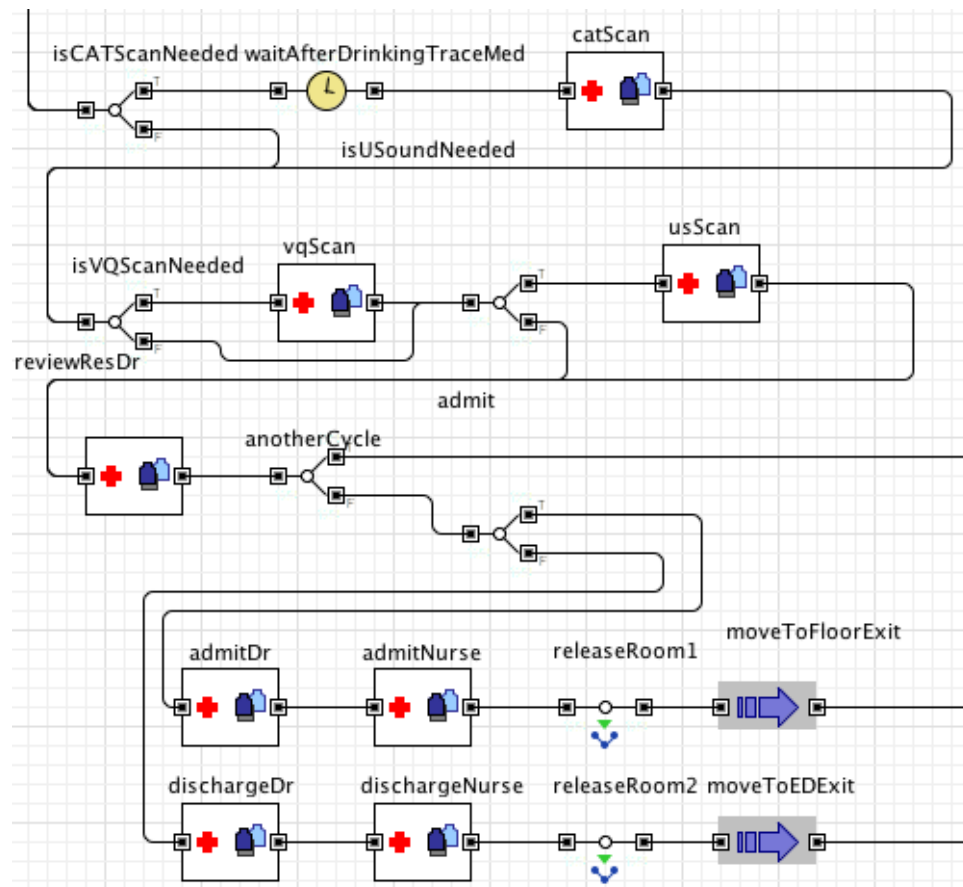


Regular Spatial Embedding



Discrete Event Modeling

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



“Network Modeling” Irregular Spatial Embedding

Emergency Department

Jan 2, 2006 8:34:00 PM

Current interarrival time: 27.972

Incoming flow rate multiplier: 1

Percent of patients walked in: 86%

Avg queue size before registration: 0.069

EC start of operation: 11.0

EC end of operation: 23.0

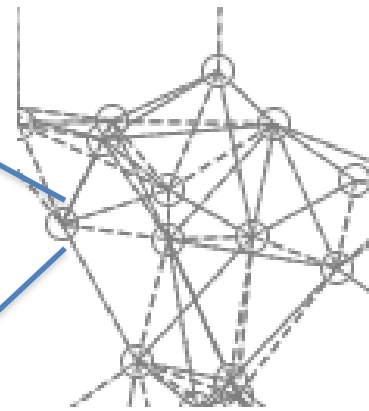
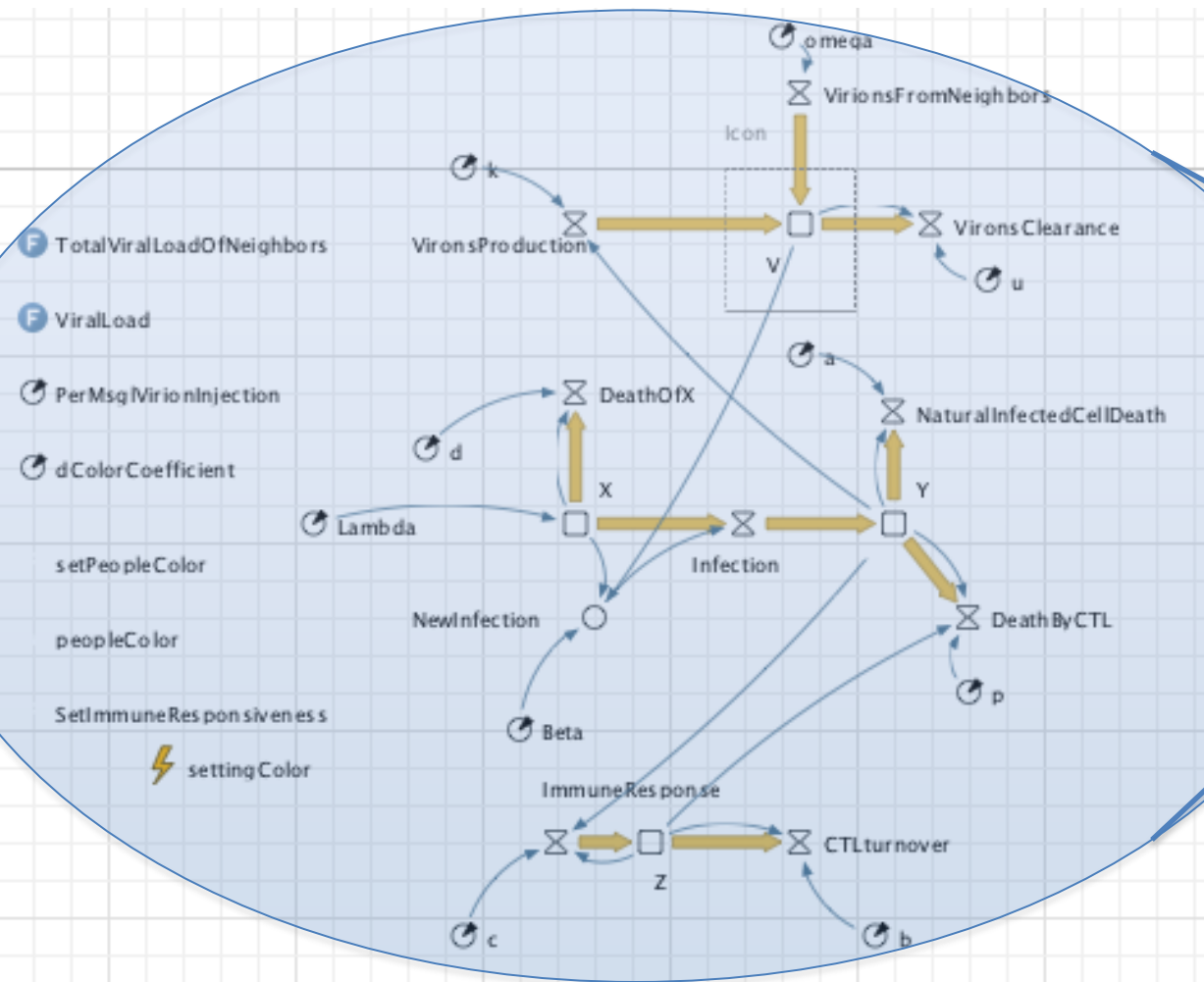


This AnyLogic (TM) model is (c) 1992-2007 XJ Technologies. The problem definition was provided by Simulation Modeling Services

Hybrid Models

- Much of the power of AnyLogic lies in its ability to integrate multiple types of modeling in a single model
- Attractive schemes
 - Agent-based using system dynamics for continuous agent state (c.f. age)
 - System dynamics using agent-based to determine flows
 - Agent-based using system dynamics for global dynamics
 - Agents entering into process-based health services

Example Hybrid Model



Advantages of AnyLogic

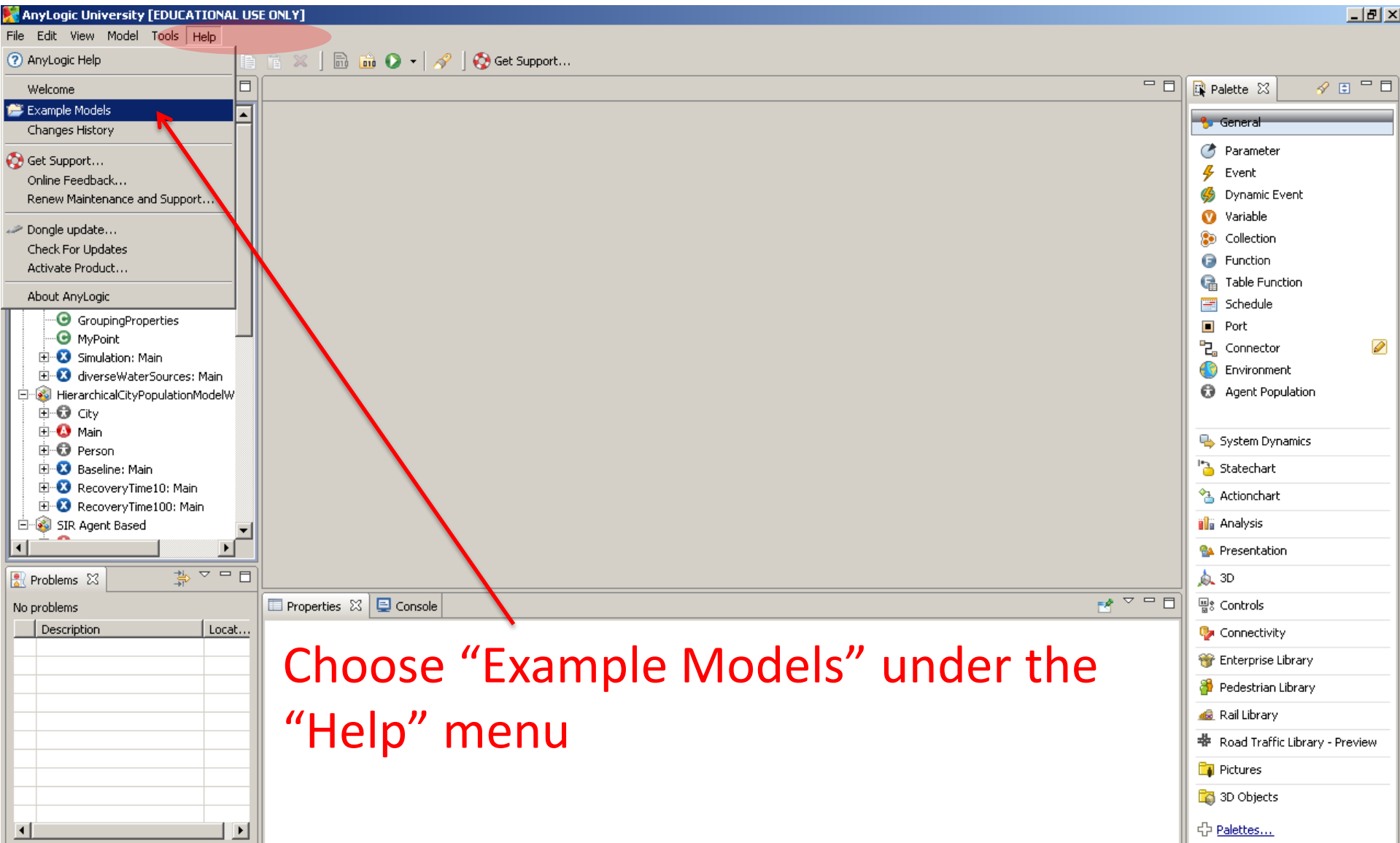
(as compared to other Agent-Based Modeling Software)

- Primarily declarative specification
- Less code
- Great flexibility
- Access to Java libraries
- Support for multiple modeling types
- Support for mixture of modeling types

Painful Sides of AnyLogic Education/Advanced

- Export of model results: Very limited support for retrospection on model results
- Need for bits of Java code
- Many pieces of system
- Pricy debugger

Opening an AnyLogic Example Model



The screenshot shows the AnyLogic University [EDUCATIONAL USE ONLY] interface. The 'Help' menu is open, and 'Example Models' is highlighted. A red arrow points from the text 'Choose "Example Models" under the "Help" menu' to the 'Example Models' option in the menu.

Choose "Example Models" under the "Help" menu



Hands on Model Use Ahead



Load AnyLogic Example Model: SIR Agent
Based.alp

Finding the Example Model

AnyLogic University [EDUCATIONAL USE ONLY]
File Edit View Model Tools Help

Welcome

AnyLogic® 6 NB: May have to use scroll bar

Getting Started with AnyLogic | What's New in AnyLogic | Professional Features | **Example Models** | Ask Question/Get Support | XJ Technologies Website

Examples

- All models (62)
- Market and Competition (8)
- Healthcare (10)
- Manufacturing (4)
- Supply Chains (5)
- Logistics (5)
- Business Processes (4)
- Social and Eco Dynamics (9)
- Defence (3)
- Project and Asset Management (2)
- IT Infrastructure (1)
- Road Traffic (3)
- Miscellaneous (8)

How-To Models

'Business Dynamics' Book Models

Competition in Global Pulp Market

Exposure to Radiation

Fun City Driving

Gas Station

Harvest Simulator

Hump Yard

Insurgency Dynamics

Inventory Workforce

Leader Election Protocol

Petri Nets

Predator Prey

Product Portfolio Management

Restaurant Business

SIR

SIR Agent Based Calibration

Schelling Segregation

Steel Converter Process

Supply Chain

The Three Modeling Methods Work Together

Train Unloading

Emergency Department

Flocks of Boids

GIS Example

HIV Diffusion and Syringe Usage

Highway Junction

Influenza

Interconnected Call Centers

Job Shop

Lorenz Weather Model

Population

Predator Prey Agent Based

Railroad Crossing

Robot for Golem Manufacturing

SIR Agent Based

James Funnel

Serial Killers

Supply Chain

The Game of Life

Three Body Problem

Trauma Center

Click on "SIR Agent Based"
Exact location will vary with screen resolution

Model Focus: Spatial Spread of an Infectious Disease

- This model simulates the spread of an infectious disease in a regular space
- The simulation starts with a single index infective case (towards lower right of space)
- Natural history of infection involves progression from Susceptible to Infected (& Infective) to Recovered
 - There is no waning of immunity in the original model
- If a given person is infective, the infection can spread from that person to their neighbours in the 4 cardinal directions (“North”, “South”, “East”, “West”) (i.e. Up, Down, Left, and Right)

Viewing the Model Structure

Double click on “Person” to see the associated state transition diagram. This diagram represents in a stylized fashion the progression of infection

The screenshot displays the AnyLogic software interface. On the left, the 'Project' tree shows a hierarchy: 'SIR Agent Based' > 'Main' > 'Person'. A red arrow points from the text 'Double click on "Person"' to the 'Person' folder. The main workspace shows a state transition diagram for the 'Person' agent, which is highlighted with a blue oval. The diagram starts with a yellow 'Susceptible' state, followed by a transition labeled 'Infection' to a red 'Infectious Contact' state, and finally a transition labeled 'Recovery' to a grey 'Recovered' state. A 'statechart' label is at the top of the diagram. The bottom of the interface shows the 'Properties' panel for the 'Person - Active Object Class', with the 'Agent' checkbox checked. The 'Console' panel is also visible.

Run the Model (**Right Click** the Experiment “Simulation” & select **“Run”**)

The screenshot displays the AnyLogic University software interface, titled "AnyLogic University [EDUCATIONAL USE ONLY]". The main workspace shows a statechart diagram for a "Person" agent, with states: Susceptible (yellow), Infectious Contact (orange), and Recovered (grey). Transitions are labeled "Infection" and "Recovery". A "color" variable is also visible.

The "Projects" panel on the left shows a tree structure with "SIR Agent Based*" containing "Main", "Person", and "Simulation - Main". A right-click context menu is open over "Simulation - Main", with the "Run" option highlighted in blue. A red arrow points from the text "Right Click the Experiment" to the "Simulation - Main" item, and a blue arrow points from "select 'Run'" to the "Run" button in the menu.

The "Simulation - Simulation Experiment" configuration window is open at the bottom, showing the following settings:

- Name: Simulation
- Main active object class (root): Main
- Ignore:
- Random number generation:
 - Random seed (unique simulation runs)
 - Fixed seed (reproducible simulation runs) Seed value: 1
 - Custom generator (subclass of Random): new Random()
- Average illness duration: 15

The "Palette" on the right lists various components like Parameter, Event, Dynamic Event, Variable, Collection, Function, Table Function, Schedule, Port, Connector, Environment, Agent Population, System Dynamics, Statechart, Actionchart, Analysis, Presentation, 3D, Controls, Connectivity, Enterprise Library, Pedestrian Library, Rail Library, Road Traffic Library - Preview, Pictures, and 3D Objects.

Press this button to start model execution

The screenshot displays the AnyLogic software interface for an SIR model simulation. The main window is titled "SIR Agent Based : Simulation - AnyLogic Advanced [EDUCATIONAL USE ONLY]". The simulation area shows three states: "Susceptible" (yellow), "Infectious" (red), and "Recovered" (grey). A red arrow points to a button labeled "Run the model and switch to Main view" located under the "Infectious" state. The statechart area on the right shows a grid with various model elements. The bottom status bar indicates the simulation is in "Idle" mode, with "Step: 0 [0]", "EPS: 0", "FPS: 0.0", and "Memory: 7M of 297M".

Run the model and switch to Main view

Run: 0 Idle Step: 0 [0] EPS: 0 FPS: 0.0 Memory: 7M of 297M 0.0 sec

Example of Emergent Behaviour

SIR Agent Based : Simulation - AnyLogic Advanced [EDUCATIONAL USE ONLY]

■ Susceptible: 236,782 ■ Infectious: 6,167 ■ Recovered: 7,051

Run: 0 Running Step: 687,896 [14,322] EPS: 247,415 FPS: 1.0 Memory: 1 of 297

es (x86)\AnyLogic 6\jre\bin\javaw.exe (Jul 11, 2010 6:05:05 PM)

Selection

Make Sure Model Time is Visible

The image displays the AnyLogic software interface for an SIR model simulation. The main window is titled "SIR Agent Based : Simulation - AnyLogic Advanced [EDUCATIONAL USE ONLY]". At the top, it shows the current state: Susceptible: 193,137, Infectious: 9,635, and Recovered: 47,228. The simulation area shows a large yellow area representing susceptible agents and a smaller red area representing infectious agents. The statechart on the right shows the flow from Susceptible to Infectious Contact via Infection, and from Infectious Contact to Recovered via Recovery. The bottom status bar shows the simulation is running, with Step: 2,922,431 [19,269], EPS: 218,350, FPS: 0.0, and Memory: 1 of 297M. A red circle highlights a button in the bottom status bar, which is used to add a "model time" output.

If no model time is visible on the bottom of the window, press **this button** to add a "model time" output

Select "Model Time" here (so a check mark appears)
(If a checkmark is already present, just click back on the

output window

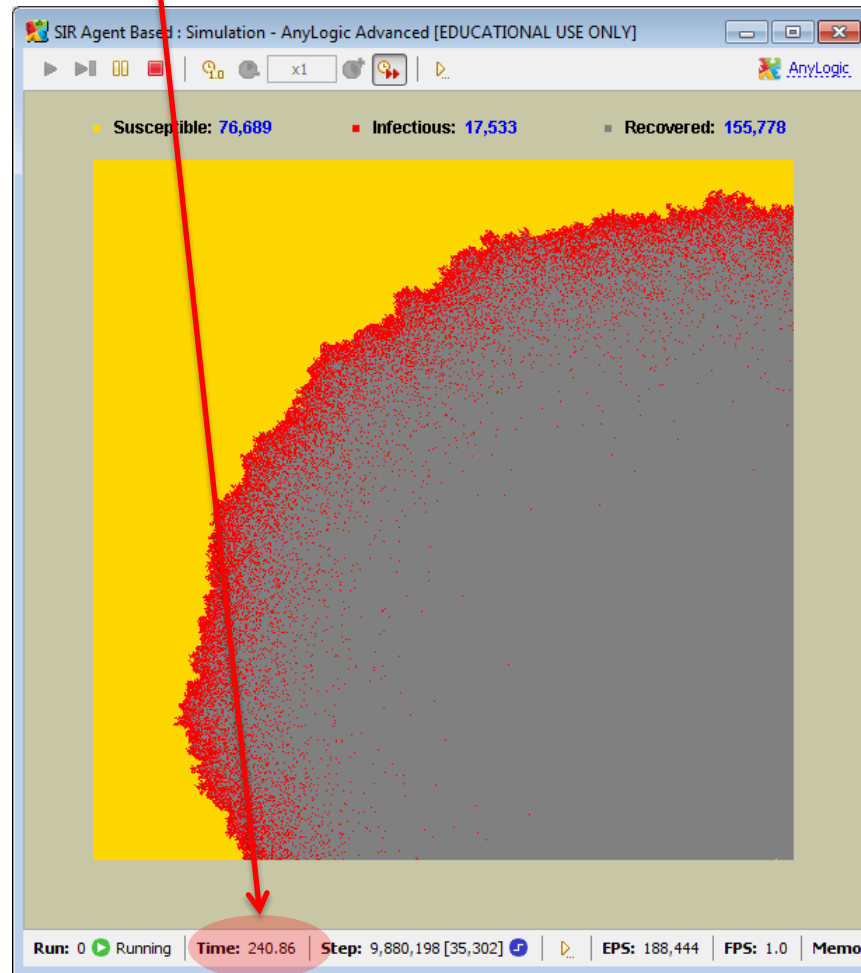
The screenshot shows the AnyLogic software interface for an SIR simulation. At the top, the status bar displays: Susceptible: 166,449, Infectious: 11,499, and Recovered: 72,052. The main area is a large yellow rectangle representing the susceptible population, with a red and grey area representing the infectious and recovered populations. A context menu is open over the simulation area, with the 'Model time' option highlighted in red. The menu items are: Status (checked), Model time (highlighted), Model step (checked), Experiment, Simulation, Model date, Events per second (checked), Frames per second (checked), Memory (checked), and Real time (checked). The bottom status bar shows 'Run: 0 Running', 'Step: 4,656,702 [23,534]', and 'Memory: 741 of 297M'. The bottom left corner shows 'Slide 16 of 56', 'Office Theme', and 'English (Canada)'.

- Status
- Model time
- Model step
- Experiment
- Simulation
- Model date
- Events per second
- Frames per second
- Memory
- Real time

Run: 0 Running | Step: 4,656,702 [23,534] | Memory: 741 of 297M

Slide 16 of 56 | Office Theme | English (Canada)

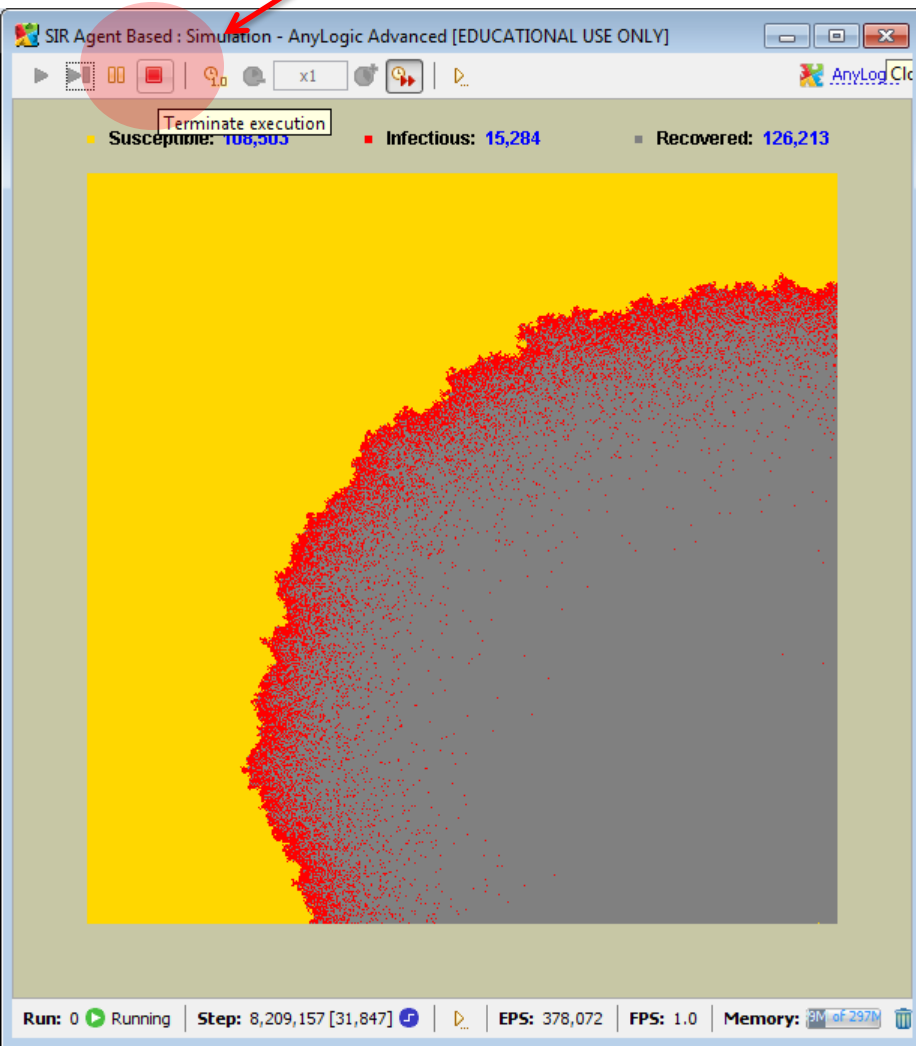
The Updated Window Should Include a Model Time Output



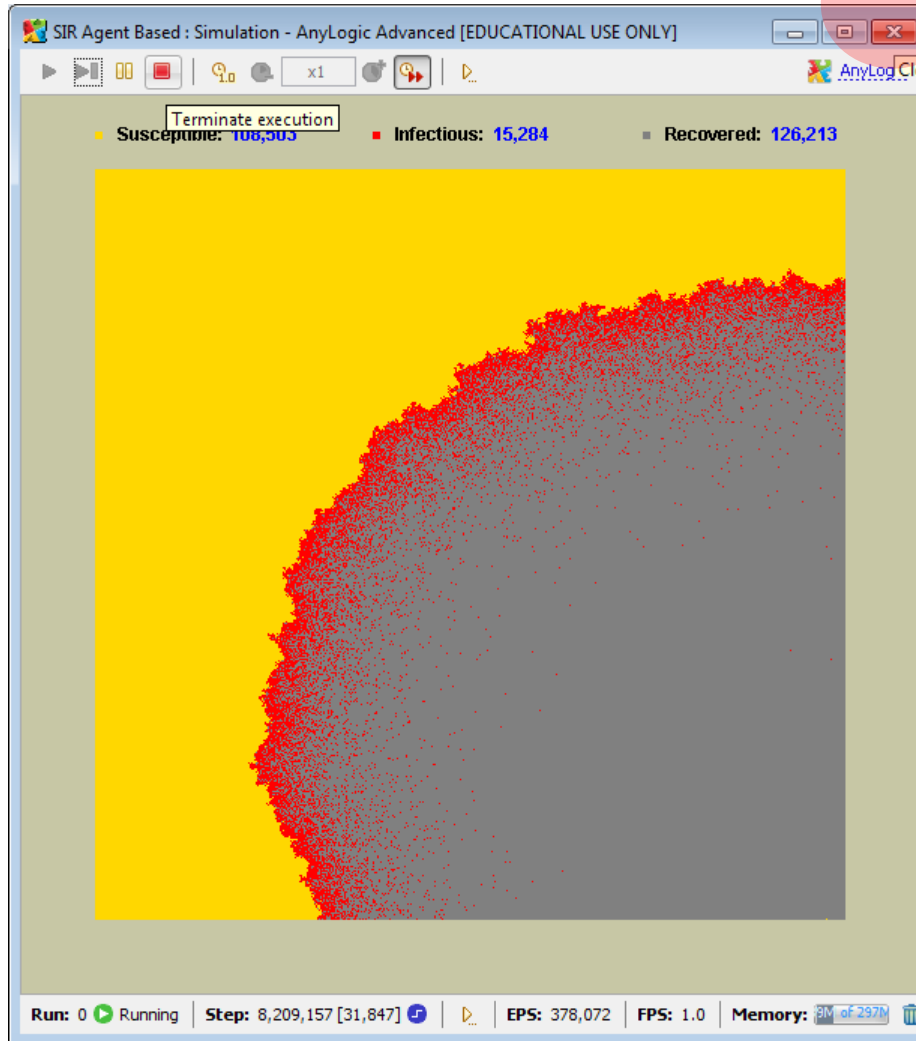
Stylized Measurement 1

- How Long Does it Take for The Infection to Reach the Top or Left Boundaries?
- We'll compare this to the situation with other assumptions regarding the progression of the infection (as encoded by model "parameters")

Press this button to stop model execution



Close the window using this button



We'll Now Modify the Model

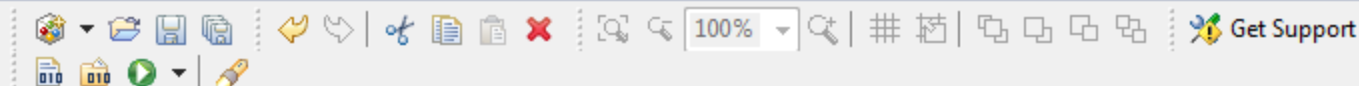
Note that May Get this Warning (Can Ignore)

The screenshot displays the AnyLogic software interface. The main workspace shows a statechart for a 'Person' agent. The statechart starts with a 'statechart' entry point leading to a yellow 'Susceptible' state. From 'Susceptible', an 'Infection' event leads to an orange 'Infectious' state. From 'Infectious', a 'Contact' event leads to another 'Infectious' state, and a 'Recovery' event leads to a 'Recovered' state. A 'color' variable is shown next to the 'Susceptible' state.

A 'Save model' dialog box is overlaid on the workspace, displaying an error message: 'Model SIR Agent Based save failed: C:\Program Files\AnyLogic 6_7 University\plugins\com.xj.anylogic.examples_6.7.1.201201271258\models\SIR Agent Based\SIR Agent Based.autosave.alp (Access is denied)'. The dialog has an 'OK' button.

The bottom of the interface shows the 'Properties' and 'Console' tabs. The 'Properties' tab is active, showing settings for a 'SlowRecovery - Simulation Experiment'. The 'General' section includes: Name: SlowRecovery, Main active object class (root): Main, and an 'Ignore' checkbox. The 'Random number generation' section has three options: 'Random seed (unique simulation runs)', 'Fixed seed (reproducible simulation runs)' (selected), and 'Custom generator (subclass of Random)'. The 'Fixed seed' option has a 'Seed value' of 1. The 'Custom generator' option is set to 'new Random()'. The 'Average illness duration' is set to 15.

The 'Problems' tab at the bottom left shows 'No problems'.



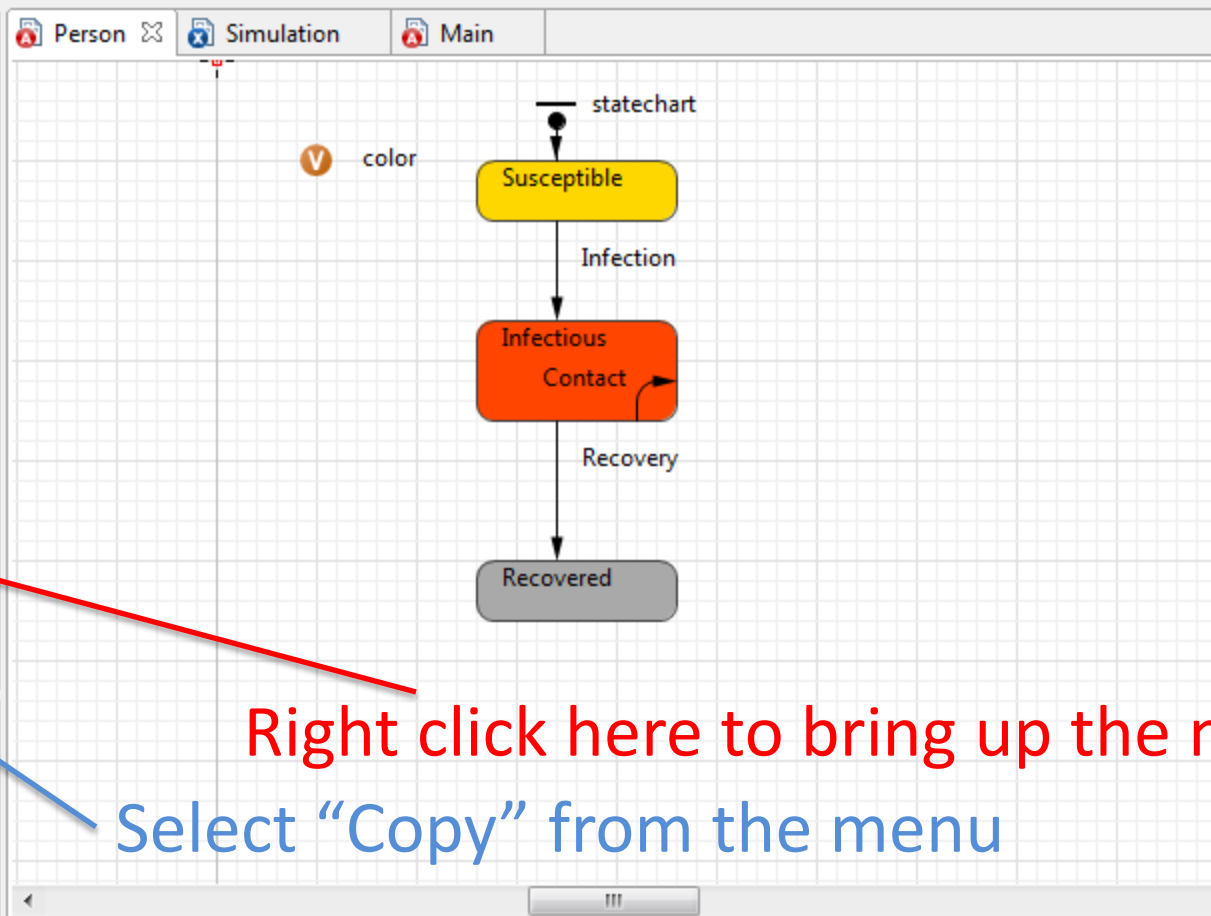
Project Search

- SIR Agent Based*
 - Main
 - Parameters
 - AverageIllnessDuration: 15
 - ContactRate: 1
 - InfectionProbability: 0.5
 - TotalPopulation: 250000
 - Plain Variables
 - Environments
 - Embedded Objects
 - Presentation
 - Person
 - Simulation: Main

Problems

Description

- New
- Open with
- Open... Ctrl+O
- Cut Ctrl+X
- Copy Ctrl+C
- Paste
- Delete Delete
- Refresh
- Refactor
- Run

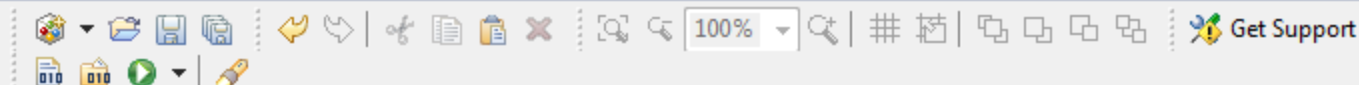


Right click here to bring up the menu
Select "Copy" from the menu

Properties Console

Simulation - Simulation Experiment

General	Advanced	Model Time	Presentation	Window
AverageIllnessDuration	ContactRate	InfectionProbability	TotalPopulation	
15	1	0.5	250000	



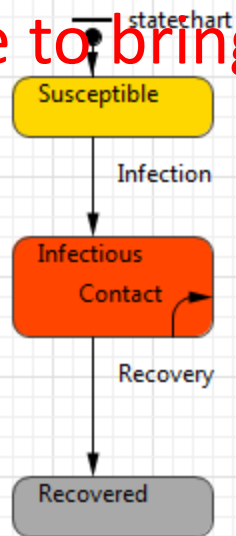
Project Search

- SIR Age
- Ma
- Sim

File menu:

- New
- Open... Ctrl+O
- Save Ctrl+S
- Save As...
- Revert
- Close
- Close Others
- Close All
- Cut Ctrl+X
- Copy Ctrl+C
- Paste
- Delete Delete
- Refresh
- Build F7
- Export...
- Team

Right click here to bring up the menu.



Select "Paste" from the menu to pas
new experiment (a copy of the existi

Properties Console

SIR Agent Based - Model

General

Name: SIR Agent Based

Dependencies

Description

Package: sir_agent_based

File: C:\Program Files (x86)\AnyLogic 6\plugins\com.xj.anylogic.examples_6.2.2.200806

Your Screen Should Look as Follows

The screenshot displays the AnyLogic University software interface, which is used for building agent-based models. The main workspace shows a statechart for a 'Person' agent. The statechart starts with a 'Susceptible' state (yellow rounded rectangle), which transitions to an 'Infectious' state (orange rounded rectangle) via an 'Infection' event. The 'Infectious' state has a self-loop labeled 'Contact' and transitions to a 'Recovered' state (grey rounded rectangle) via a 'Recovery' event. A 'color' variable is associated with the 'Susceptible' state. The interface includes a 'Projects' panel on the left, a 'Palette' on the right, and a 'Problems' panel at the bottom left showing one error: 'Root Active Object Class is... SIR A...'. The status bar at the bottom indicates '0 items selected', 'Selection', and coordinates 'X=643, Y=-14'.

AnyLogic University [EDUCATIONAL USE ONLY]

File Edit View Draw Model Tools Help

Projects

- SIR Agent Based*
- Main
- Person
- Simulation: Main
- Simulation1

Person

statechart

color

Susceptible

Infection

Infectious

Contact

Recovery

Recovered

Palette

General

- Parameter
- Event
- Dynamic Event
- Variable
- Collection
- Function
- Table Function
- Schedule
- Port
- Connector
- Environment
- Agent Population

System Dynamics

- Statechart
- Actionchart
- Analysis
- Presentation
- 3D
- Controls
- Connectivity
- Enterprise Library
- Pedestrian Library
- Rail Library
- Road Traffic Library - Preview
- Pictures
- 3D Objects
- Palettes...

Problems

1 errors

Description	Locat...
Root Active Object Class is...	SIR A...

Properties Console

0 items selected

Selection

X=643, Y=-14

Changing the Name of the Experiment

The screenshot displays the AnyLogic University interface. On the left, the 'Projects' pane shows a tree view with 'SIR Agent Based*' expanded to 'Person', which contains 'Simulation: Main' and 'SlowRecovery'. A red oval highlights 'SlowRecovery'. In the center, a statechart for the 'Person' class shows states: 'Susceptible' (yellow), 'Infectious' (orange), and 'Recovered' (grey). Transitions are labeled 'infectious' and 'Recovery'. A red arrow points from the text '1) Select here' to the 'SlowRecovery' project in the Projects pane. A blue arrow points from the text '2) Select the "General" tab' to the 'General' tab in the Properties window. A green arrow points from the text '3) Type the name "SlowRecovery"' to the 'Name' field in the Properties window, which contains 'SlowRecovery'. The Properties window also shows the 'Main active object class (root)' dropdown and 'Random number generation' options.

1) Select here (the new experiment) so we can edit its properties (characteristics)

2) Select the "General" tab

3) Type the name "SlowRecovery" for the new experiment

Selecting the Model Used for this Experiment

The screenshot displays the AnyLogic software interface. On the left, the 'Projects' pane shows a tree structure for 'SIR Agent Based*' with sub-items: 'Main', 'Person', 'Simulation: Main', and 'SlowRecovery'. A red arrow points from the text 'Select "Main" here' to the 'Main' item in this tree.

The central workspace shows a statechart for the 'Person' agent. It starts at a 'statechart' node, leading to a yellow 'Susceptible' state. A transition labeled 'Infection' leads to an orange 'Infectious' state, which has a self-loop labeled 'Contact'. A transition labeled 'Recovery' leads to a grey 'Recovered' state. A variable 'color' is shown as a yellow circle.

At the bottom, the 'Properties' pane is open for the 'SlowRecovery - Simulation Experiment'. The 'General' tab is active, showing the 'Name' as 'SlowRecovery' and the 'Main active object class (root)' dropdown menu. The dropdown menu is open, showing 'Main' and 'Person' as options. A red oval highlights this dropdown menu, and a red arrow points from the 'Main' item in the 'Projects' pane to this oval.

Other panes include 'Problems' (showing 1 error), 'Console', and a 'Palette' on the right with various modeling elements like 'Parameter', 'Event', 'Dynamic Event', etc.

Altering Assumptions Regarding Infectiousness Duration (via Parameters)

The screenshot displays the AnyLogic Advanced software interface. On the left, a project tree shows the 'Parameters' tab selected. The main workspace contains a statechart with states: Susceptible (yellow), Infectious Contact (orange), and Recovered (grey). Transitions are labeled 'Infection' and 'Recovery'. A 'color' variable is also shown. The bottom right pane shows the 'Properties' window for the 'SlowRecovery - Simulation Experiment' with the following parameters:

Parameter	Value
AveragellnessDuration	50
ContactRate	1
InfectionProbability	0.5
TotalPopulation	250000

Annotations include a red arrow pointing to the 'Parameters' tab in the project tree and a blue arrow pointing to the 'AveragellnessDuration' value of 50 in the properties window.

1) Select the "Parameters" tab

2) Make the illness duration 50

Run the Model (Right Click the Experiment "SlowRecovery" & select "Run")

The screenshot displays the AnyLogic University software interface. The main workspace shows a statechart for a 'Person' agent. The statechart starts at a 'Susceptible' state (yellow), which transitions to an 'Infectious' state (orange) upon 'Infection'. The 'Infectious' state has a 'Contact' action and transitions to a 'Recovered' state (grey) upon 'Recovery'. A 'color' variable is associated with the 'Susceptible' state.

The 'Projects' panel on the left shows a tree view with 'SIR Agent Based*' expanded to 'Main', 'Person', 'Simulation: Main', and 'SlowRecovery: Main'. A red arrow points to 'SlowRecovery: Main', and a context menu is open over it with the 'Run' option selected. A blue arrow points from the 'Run' option to the statechart.

The 'Properties' panel at the bottom shows the configuration for the 'SlowRecovery - Simulation Experiment'. The 'General' tab is active, showing the experiment name 'SlowRecovery' and the main active object class 'Main'. The 'Random number generation' section is configured with 'Fixed seed (reproducible simulation runs)' selected, a 'Seed value' of 1, and a custom generator 'new Random()'. The 'Average illness duration' is set to 15.

The 'Palette' on the right lists various modeling elements such as Parameter, Event, Dynamic Event, Variable, Collection, Function, Table Function, Schedule, Port, Connector, Environment, and Agent Population.

You Should See Something Like This

SIR Agent Based : Simulation - AnyLogic Advanced [EDUCATIONAL USE ONLY]

Get Support

statechart

Susceptible

Infection

Infectious
Contact

Recovery

Recovered

Model

- Parameter
- Flow Aux Variable
- Stock Variable
- Event
- Dynamic Event
- Plain Variable
- Collection Variable
- Function
- Table Function
- Port
- Connector
- Entry Point
- State
- Transition
- Initial State Pointer
- Branch
- History State
- Final State
- Environment

es (x86)\AnyLogic 6\jre\bin\javaw.exe (Jul 11, 2010 6:47:45 PM)

Run: 0 Running Time: 201.45 Step: 31,562,999 [175,702] EPS: 143,862 FPS: 1.0 Men

How quickly does the wave of infection take to reach the top border? How does this compare to the situation where we assumed a shorter period of infectiousness? Why?

Adding a Transition

The screenshot displays the AnyLogic software interface. The main workspace shows a statechart for a 'Person' agent. The statechart starts at a 'statechart' node, leading to a 'Susceptible' state (yellow rounded rectangle). A transition labeled 'Infection' leads to an 'Infectious' state (orange rounded rectangle). From 'Infectious', a transition labeled 'Recovery' leads to a 'Recovered' state (grey rounded rectangle). A 'color' variable is also shown. The right-hand 'Palette' window is open, and the 'Statechart' icon is highlighted with a red oval and a red arrow pointing to it. The 'Statechart' icon is located under the 'System Dynamics' category in the palette.

Click on "Statechart" to view
The statechart-related palette

Adding a Transition

The screenshot displays the AnyLogic software interface. The main workspace shows a statechart for a 'Person' agent. The statechart consists of three states: 'Susceptible' (yellow), 'Infectious' (orange), and 'Recovered' (grey). Transitions are labeled 'Infection' (from Susceptible to Infectious) and 'Recovery' (from Infectious to Recovered). A red arrow points from the 'Transition' icon in the 'Statechart' palette to the 'Recovered' state. A red text box at the bottom contains instructions: 'To add a transition to the statechart Drag from "Transition" on the Palette to the "Recovered" state'.

**To add a transition to the statechart
Drag from "Transition" on the Palette to
the "Recovered" state**

Connecting the Two States

While holding down the mouse button, drag the mouse to **here** and only then Release the mouse button

1) Dragging the transition should have led to a connection here

2) Click on the other end of the transition,

transition - Transition

Description	Locat...
Hanging transition	SIR A..

1 errors

Properties Console

transition - Transition

General

Name: transition Show name Ignore Show at runtime

Triggered by: Timeout

Timeout: 1

Action:

Guard:

Selection X=477, Y=177

Give the Transition a Name (Make sure it is selected by clicking on it)

The screenshot displays the AnyLogic Advanced interface. The central workspace shows a statechart with three states: Susceptible (yellow), Infectious (orange), and Recovered (grey). Transitions are labeled 'Infection' (Susceptible to Infectious), 'Recovery' (Infectious to Recovered), and 'Contact' (Infectious to Infectious). A red arrow points from the text 'Type the name ("waningImmunity") here' to the 'Name' field in the 'transition - Transition' properties panel, which contains the text 'waningImmunity'. The left sidebar shows a project tree with 'SIR Agent Based*' and 'Main' selected. The right sidebar shows a palette of modeling elements. The bottom panel shows the 'Properties' window for the selected transition.

Project: SIR Agent Based*
Main
Parameters
AveragellnessDuration: 15
ContactRate: 1
InfectionProbability: 0.5
TotalPopulation: 250000
Plain Variables
Environments
Embedded Objects
Presentation
Person
Simulation: Main
SlowRecovery: Main

statechart
color
Susceptible
Infectious
Contact
Recovery
Recovered

Type the name ("waningImmunity") here

transition - Transition
Press Ctrl+Enter to perform refactoring (replace name occurrences)
Name: waningImmunity
Show Name Ignore Public Show At Runtime
Triggered by: Timeout
Timeout: 1
Action:

Model
Parameter
Flow Aux Variable
Stock Variable
Event
Dynamic Event
Plain Variable
Collection Variable
Function
Table Function
Port
Connector
Entry Point
State
Transition
Initial State Pointer
Branch
History State
Final State
Environment
Action
Analysis
Presentation
Connectivity
Enterprise Library
More Libraries...

Setting the Duration Until Immunity Wanes

The screenshot displays the AnyLogic Advanced interface. On the left, a project tree shows the 'SIR Agent Based*' model with a 'Main' statechart. The central workspace shows a statechart with three states: 'Susceptible' (yellow), 'Infectious' (orange), and 'Recovered' (grey). Transitions are labeled 'Infection' (Susceptible to Infectious), 'Contact' (Infectious to Susceptible), and 'Recovery' (Infectious to Recovered). A 'color' variable is also shown. On the right, a palette lists various model elements. At the bottom, the 'Properties' window for the 'waningImmunity - Transition' is open. The 'Triggered by' dropdown is set to 'Timeout', and the 'Timeout' value is set to '100'. A red arrow points from the text '1) Make sure this is set to "Timeout"' to the dropdown menu. A blue arrow points from the text '2) Set the waning time To 100' to the '100' value in the 'Timeout' field.

1) Make sure this is set to "Timeout"

2) Set the waning time To 100

Run the original experiment (“Simulation”) with the newly changed model by right clicking on “Simulation” & selecting

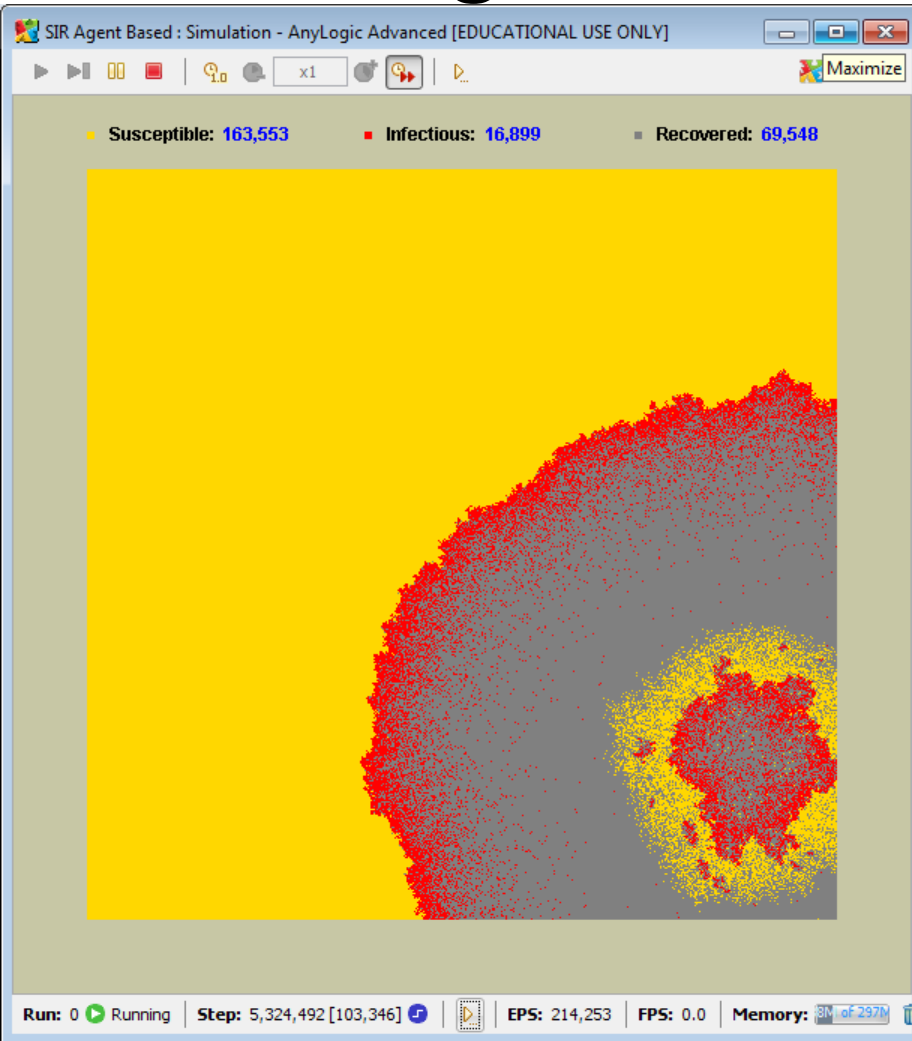
Run

The screenshot shows the AnyLogic University interface. The main workspace displays a statechart for a 'Person' agent with states: Susceptible (yellow), Infectious (orange), and Recovered (grey). Transitions include 'Infection' from Susceptible to Infectious, 'Recovery' from Infectious to Recovered, and 'Contact' from Infectious to Infectious. A 'color' variable is also shown. A context menu is open over the 'Simulation' element in the Projects pane, with the 'Run' option highlighted. The 'Simulation - Simulation Experiment' window at the bottom shows the following parameters:

Parameter	Value
AverageIllnessDuration	15
ContactRate	1
InfectionProbability	0.5
TotalPopulation	250000

Let's Run the Revised Model!

After Starting the Model, You Should See Something Like This. What Happens as Time Progresses?



Let's Run the Revised Model!

Microsoft PowerPoint

Slide 32 of 63 Office Theme English (Canada) 50%

- When it is much simpler to describe behavior at indiv. level
- Seek flexibility in exploring different heterogeneity dimensions
- Needs of stakeholders to engage with individual-based models
- Want to describe behaviour at multiple scales

Click to add notes

What Happens as Time Progresses?

SIR Agent Based : Simulation - AnyLogic Advanced [EDUCATIONAL USE ONLY]

Susceptible: 51,360 Infectious: 29,672 Recovered: 168,968

statechart

```
graph TD
    Start(( )) --> Susceptible
    Susceptible -- Infection --> Infectious
    Infectious -- Recovery --> Recovered
    Recovered --> Susceptible
```

Run: 0 Running Step: 17,514,034 [229,125] EPS: 137,195 FPS: 1.0 Memory: 10M of 297M

es (x86)\AnyLogic 6\jre\bin\javaw.exe (Jul 11, 2010 7:42:29 PM)

Palette

- Model
- Parameter
- Flow Aux Variable
- Stock Variable
- Event
- Dynamic Event
- Plain Variable
- Collection Variable
- Function
- Table Function
- Port
- Connector
- Entry Point
- State
- Transition
- Initial State Pointer
- Branch
- History State
- Final State
- Environment

Action

Analysis

Presentation

Connectivity

Enterprise Library

More Libraries...

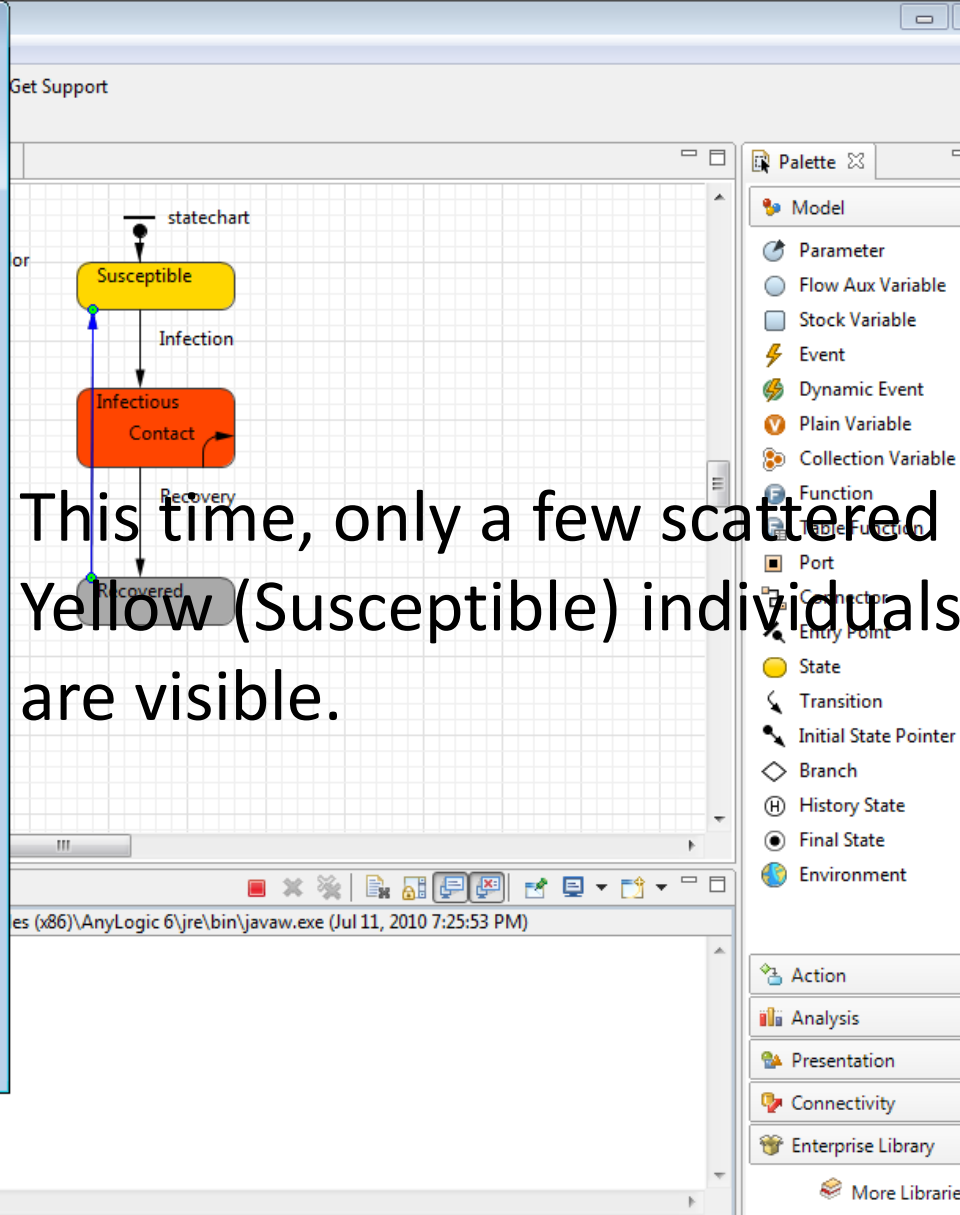
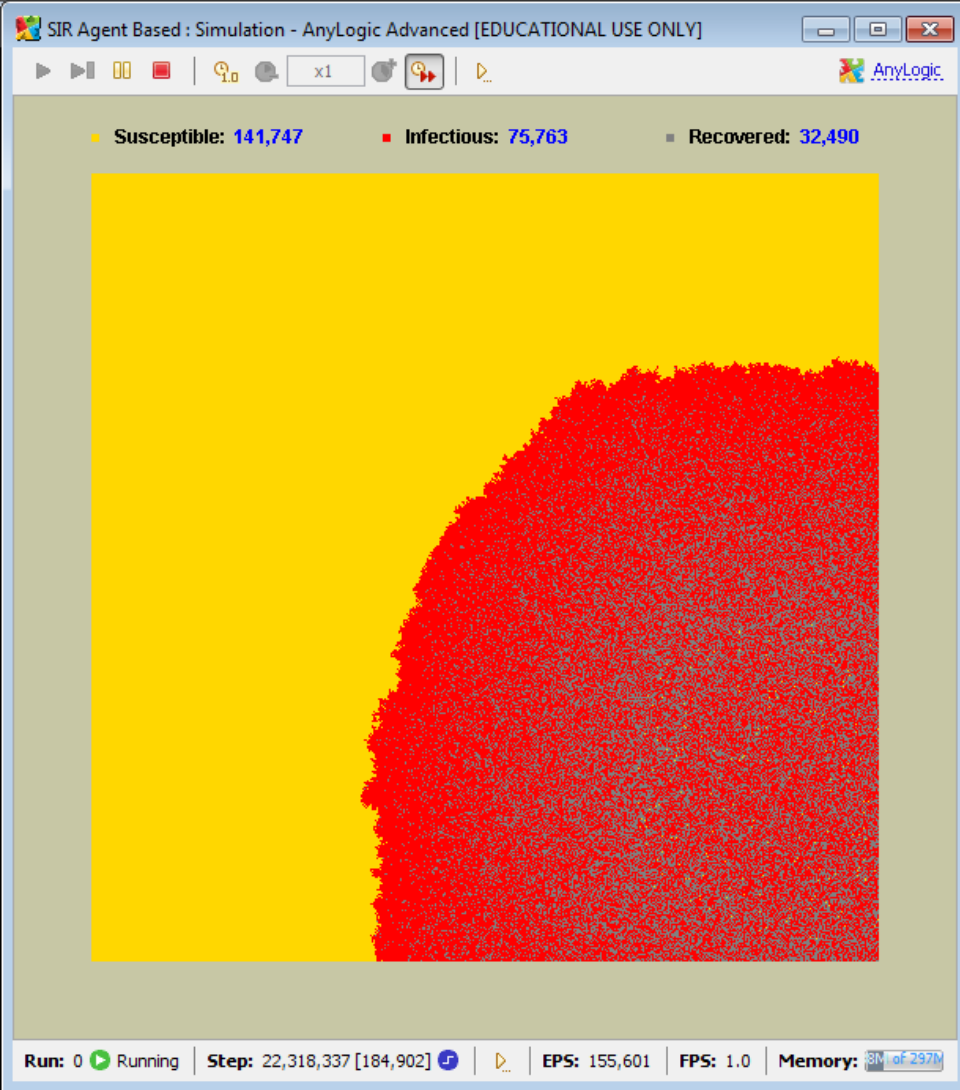
Use the Run Button & run the “SlowRecovery” Experiment

The screenshot displays the AnyLogic University interface. The main workspace shows a statechart for a 'Person' agent. The states are: Susceptible (yellow), Infectious (orange), and Recovered (grey). Transitions include 'Infection' from Susceptible to Infectious, 'Recovery' from Infectious to Recovered, and 'Contact' from Infectious to Infectious. A 'color' variable is also shown. The 'Run' button in the Projects palette is highlighted with a red arrow.

The 'SlowRecovery - Simulation Experiment' configuration panel is visible at the bottom, showing the following parameters:

Category	Parameter	Value
General	AverageIllnessDuration	50
	ContactRate	1
	InfectionProbability	0.5
	TotalPopulation	250000
Parameters		
Description		

Slow Recovery Results



This time, only a few scattered Yellow (Susceptible) individuals are visible.

As Time Progresses, Little Internal Structure – Why?

The screenshot displays the AnyLogic software interface for an SIR agent-based simulation. The main window is titled "SIR Agent Based : Simulation - AnyLogic Advanced [EDUCATIONAL USE ONLY]".

Simulation Status:

- Susceptible: 50,155
- Infectious: 134,360
- Recovered: 65,485

The simulation area shows a spatial distribution of agents. A yellow region represents susceptible agents, and a red region represents infectious agents. The red region is significantly larger and more dense than the yellow region, indicating a high prevalence of the infectious state.

Statechart:

```
statechart
    state Susceptible
    state Infectious
    state Recovered
    Susceptible --> Infectious : Infection
    Infectious --> Recovered : Recovery
    Recovered --> Susceptible
```

The statechart shows three states: Susceptible (yellow), Infectious (orange), and Recovered (grey). Transitions are labeled "Infection" (from Susceptible to Infectious) and "Recovery" (from Infectious to Recovered). There is also a feedback loop from Recovered back to Susceptible.

Palette:

- Model
- Parameter
- Flow Aux Variable
- Stock Variable
- Event
- Dynamic Event
- Plain Variable
- Collection Variable
- Function
- Table Function
- Port
- Connector
- Entry Point
- State
- Transition
- Initial State Pointer
- Branch
- History State
- Final State
- Environment

Bottom Status Bar:

- Run: 0 Running
- Step: 56,894,173 [334,467]
- EPS: 134,267
- FPS: 1.0
- Memory: 18M of 297M

Stylized Measurement 2

- How Long Does it Take for The Infection to Reach the Top or Left Boundaries?
- How does this compare with the earlier experiment with a shorter duration of immunity?
- **Bonus question:** What would an aggregate (random mixing) model have predicted?

Observations

- A brief & informal glimpse of AnyLogic's user interface for building, modifying & running models
- Take-Home Points
 - Much of a model can be described graphically
 - Running a structurally simple model can lead to complex *emergent* patterns over time & space
 - Modifying the model quantitative assumptions (described by parameters) can significantly change results
 - Modifying the model structure can qualitatively change model behavior