# Collecting & Outputting Data from AnyLogic

Nathaniel Osgood

March 15, 2011

# Recording of Results

- A frequent modeler need is to record some components of model state over time
  - State variables (e.g. stocks)
  - States of agents
  - Summaries of model state
  - We informally term this a “trajectory file”
- *Trajectory recording is only supported by AnyLogic Professional*
- AnyLogic does allow for
  - Definition of *DataSets* that record recent values of parameters
  - Statistics summarizing model state
  - Reporting on values of data sets as a graph or table

# Techniques for Outputting Data

- Ad-Hoc Exports from variables
- Manual copies from visible datasets
- Export to files
- Writing to console
- Export to databases
- [AnyLogic Professional] Dataset archiving
- Capturing images of graphs

# Cross-Method Output Tips

- A convenient mechanism is to periodically output data using events (e.g. every time unit)
- Beyond output, be sure to save information on context of run
  - Model version (Use unique id that increment whenever change model)
  - Parameter assumptions
  - Intention
- Think carefully about whether want to save away intermediate data

⚠

Hands on Model Use Ahead

Load Sample Model:

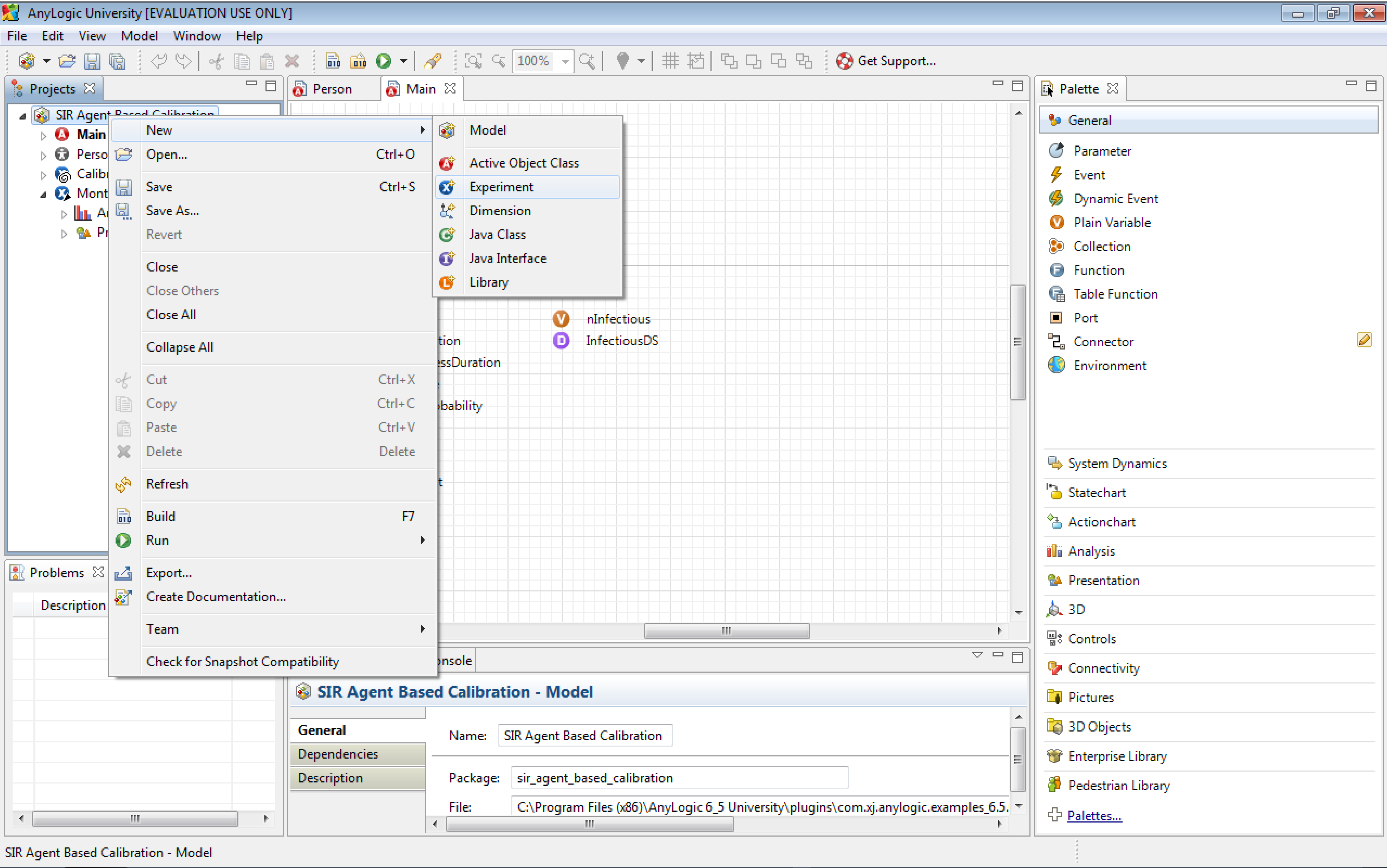
**SIR Agent Based Calibration**

(Via “Sample Models” under “Help” Menu)

# Techniques for Collecting & Outputting Data

- Ad-Hoc Exports from variables
- Pre-Prepared methods
  - Statistics
  - Charts
  - Manual copies from visible datasets
  - Export to files
  - Writing to console
  - Export to databases
  - [AnyLogic Professional] Dataset archiving
  - Capturing images of graphs

# Add an Experiment



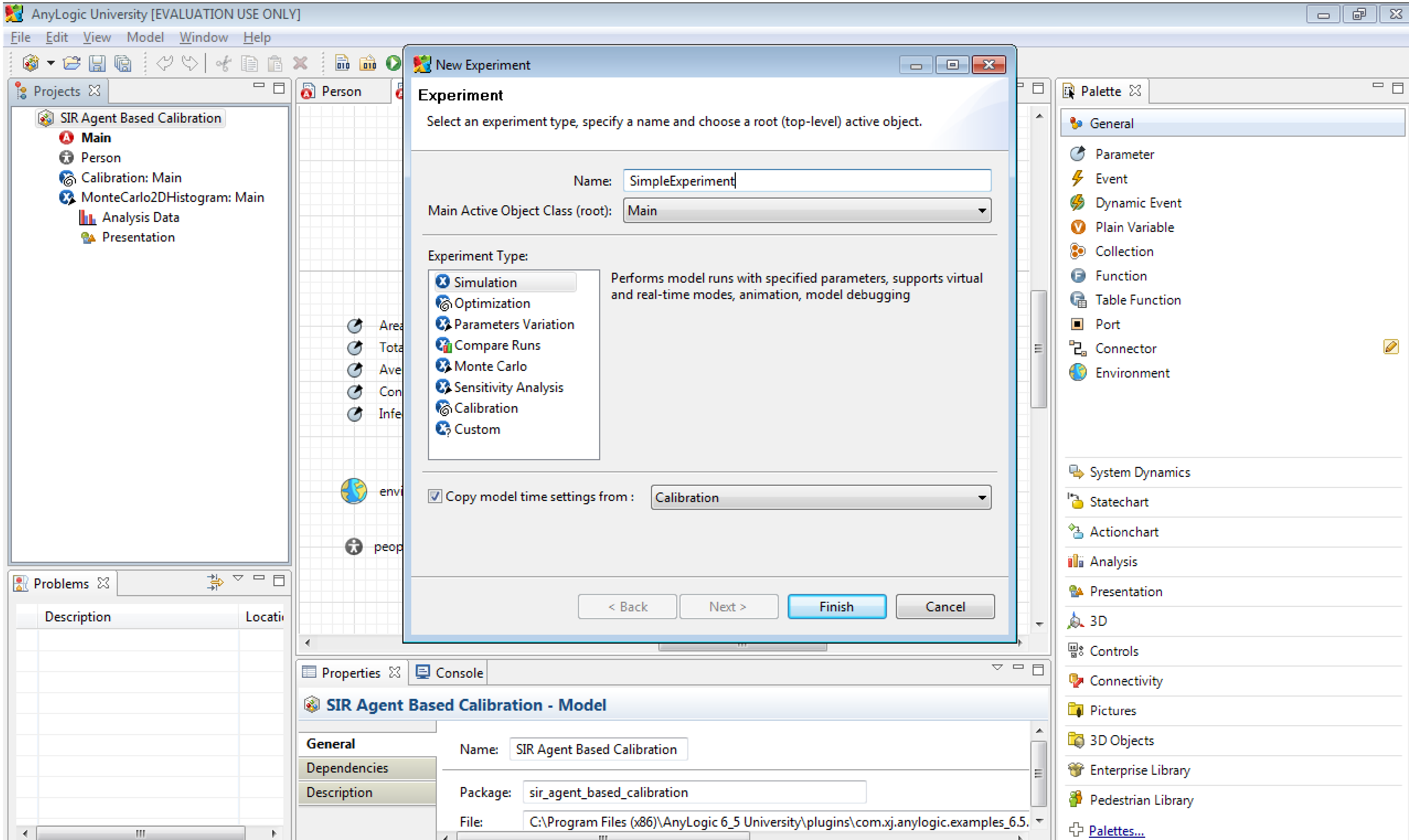
SIR Agent Based Calibration - Model

Name: SIR Agent Based Calibration

Package: sir_agent_based_calibration

File: C:\Program Files (x86)\AnyLogic 6_5 University\plugins\com.xj.anylogic.examples_6.5.

# Add an Experiment



New Experiment

**Experiment**

Select an experiment type, specify a name and choose a root (top-level) active object.

Name: SimpleExperiment

Main Active Object Class (root): Main

Experiment Type:

- Simulation
- Optimization
- Parameters Variation
- Compare Runs
- Monte Carlo
- Sensitivity Analysis
- Calibration
- Custom

Performs model runs with specified parameters, supports virtual and real-time modes, animation, model debugging

☑ Copy model time settings from : Calibration

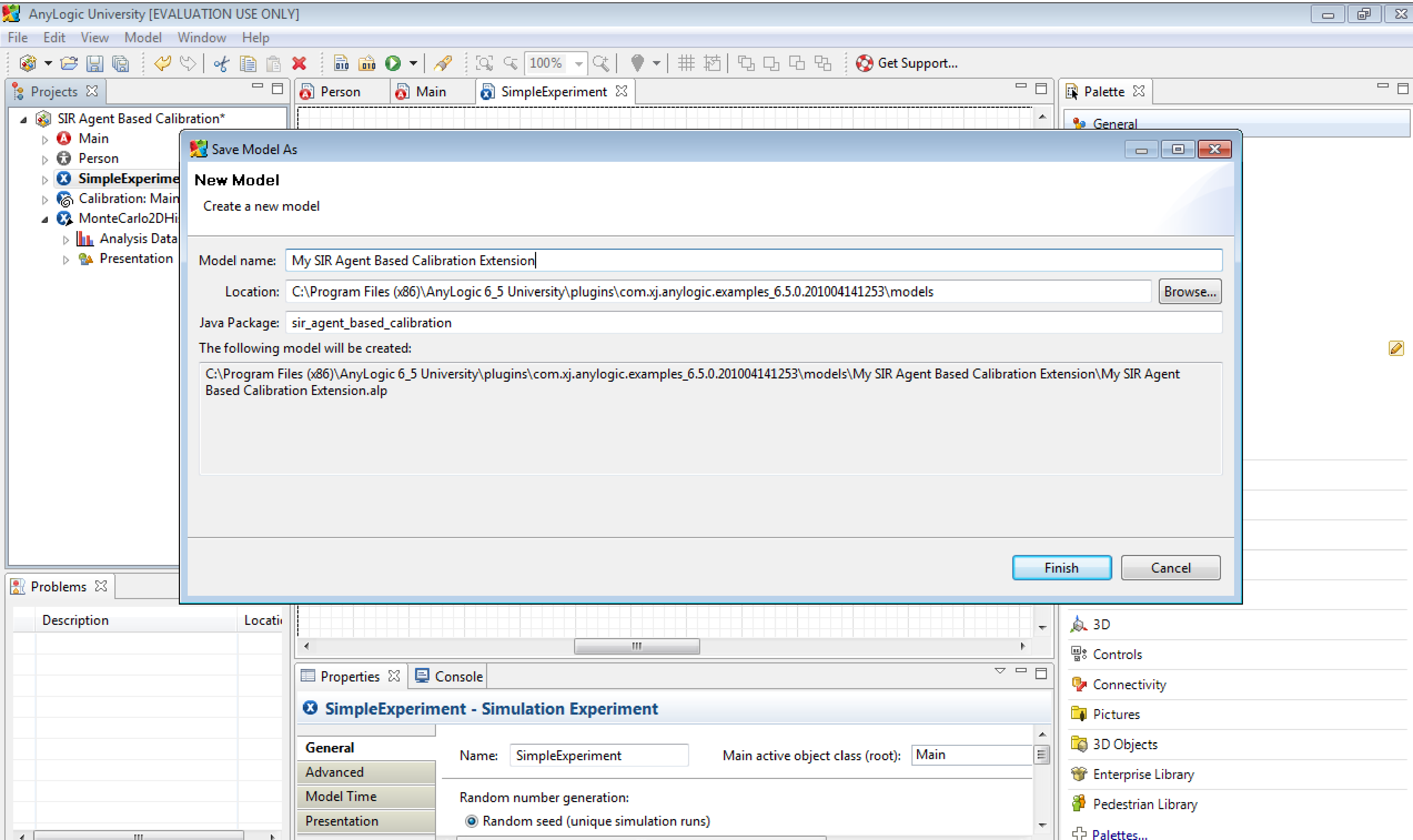
**SIR Agent Based Calibration - Model**

Name: SIR Agent Based Calibration

Package: sir_agent_based_calibration

File: C:\Program Files (x86)\AnyLogic 6_5 University\plugins\com.xj.anylogic.examples_6.5.

# Save the Resulting Model (To Avoid Overwriting the Other Model)



Save Model As

**New Model**

Create a new model

Model name: My SIR Agent Based Calibration Extension

Location: C:\Program Files (x86)\AnyLogic 6_5 University\plugins\com.xj.anylogic.examples_6.5.0.201004141253\models

Java Package: sir_agent_based_calibration

The following model will be created:

C:\Program Files (x86)\AnyLogic 6_5 University\plugins\com.xj.anylogic.examples_6.5.0.201004141253\models\My SIR Agent Based Calibration Extension\My SIR Agent Based Calibration Extension.alp

SimpleExperiment - Simulation Experiment

General | Advanced | Model Time | Presentation

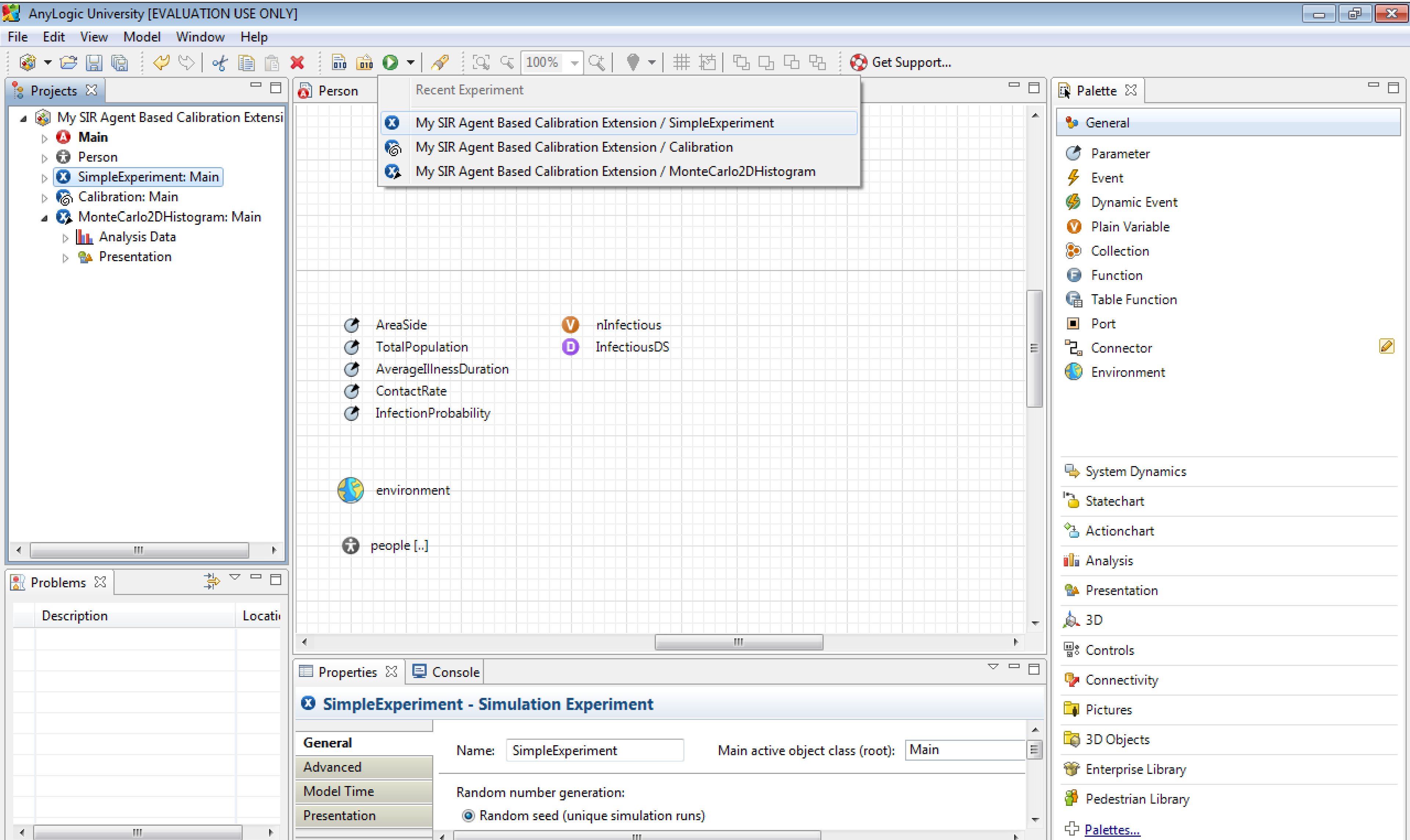
Name: SimpleExperiment

Main active object class (root): Main

Random number generation:

● Random seed (unique simulation runs)

# Run the Experiment (To Verify Functionality)



AnyLogic University [EVALUATION USE ONLY]

File Edit View Model Window Help

Recent Experiment

- My SIR Agent Based Calibration Extension / SimpleExperiment
- My SIR Agent Based Calibration Extension / Calibration
- My SIR Agent Based Calibration Extension / MonteCarlo2DHistogram

SimpleExperiment - Simulation Experiment

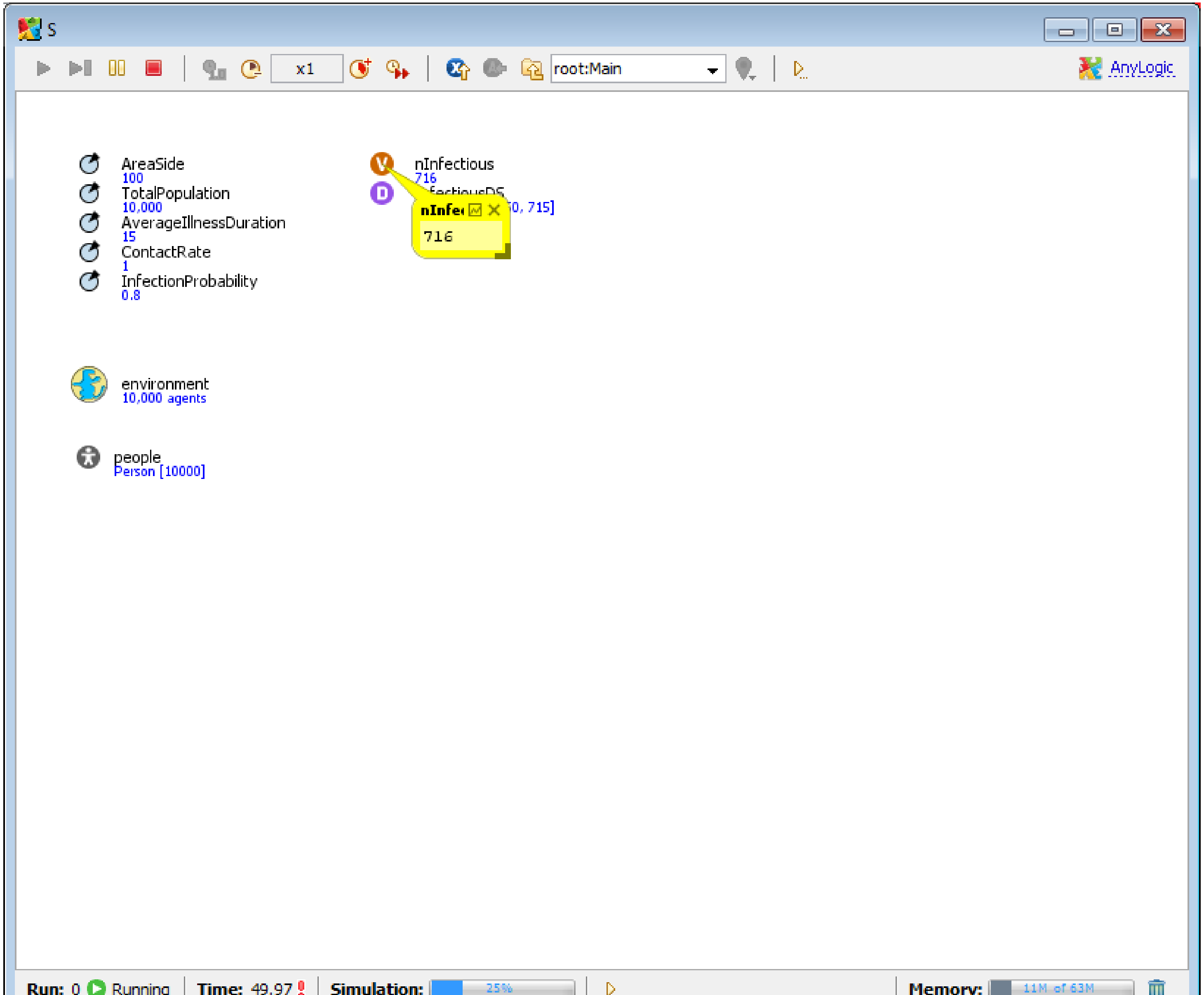
Name: SimpleExperiment

Main active object class (root): Main

Random number generation:

● Random seed (unique simulation runs)

# Click on Variable “nInfectious”



AreaSide
100

TotalPopulation
10,000

AverageIllnessDuration
15

ContactRate
1

InfectionProbability
0.8

nInfectious
716

fectiousDS
0, 715]

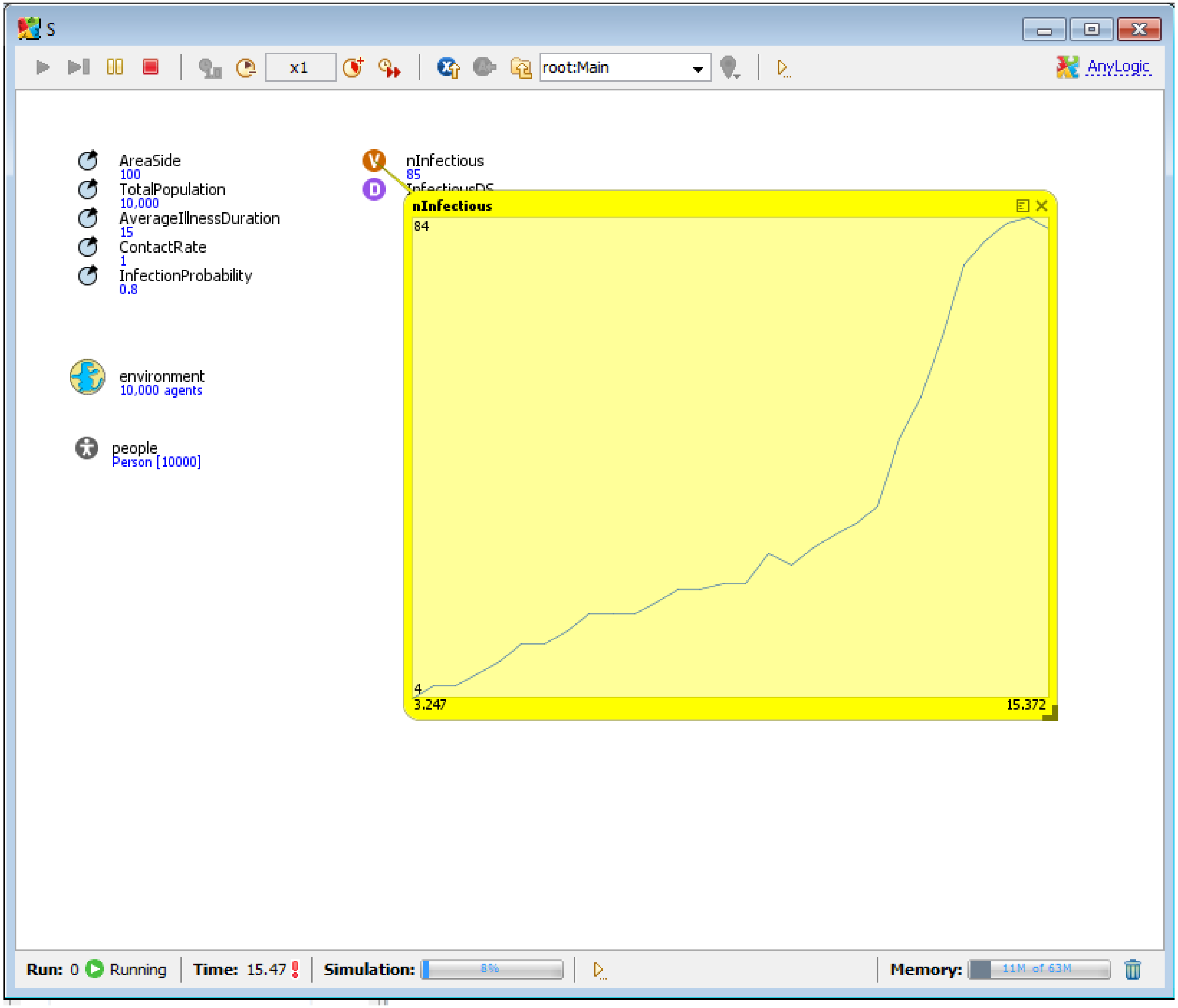
nInfe
716

environment
10,000 agents

people
Person [10000]

Run: 0 Running | Time: 49.97 | Simulation: 25% | Memory: 11M of 63M

# Graph of Variable



AreaSide
100
TotalPopulation
10,000
AverageIllnessDuration
15
ContactRate
1
InfectionProbability
0.8

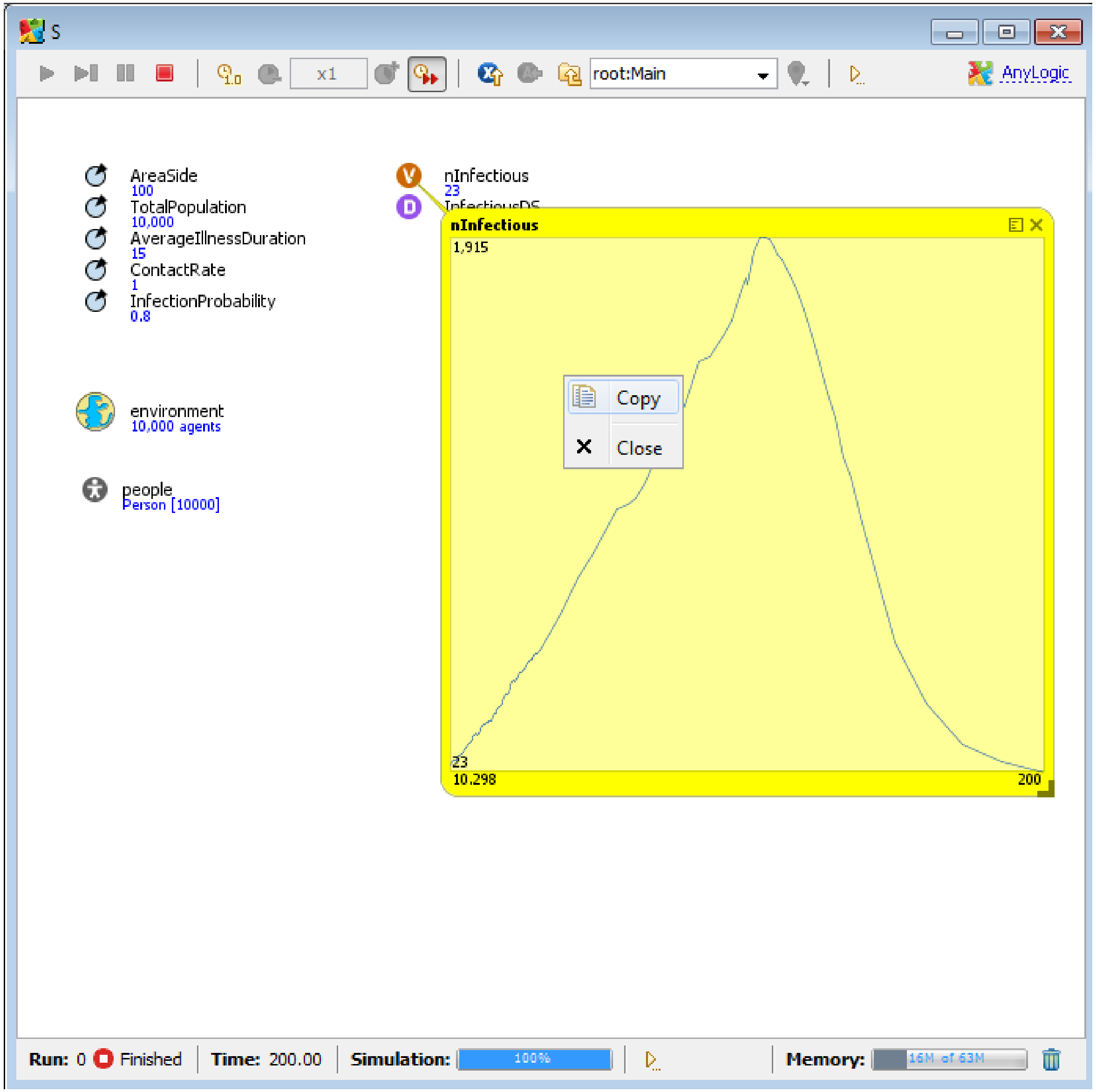
nInfectious
85
InfectiousDS

environment
10,000 agents

people
Person [10000]

Run: 0 Running | Time: 15.47 | Simulation: 8% | Memory: 11M of 63M

# Right-Click to Copy the Numeric Data



AreaSide
100

TotalPopulation
10,000

AverageIllnessDuration
15

ContactRate
1

InfectionProbability
0.8

environment
10,000 agents

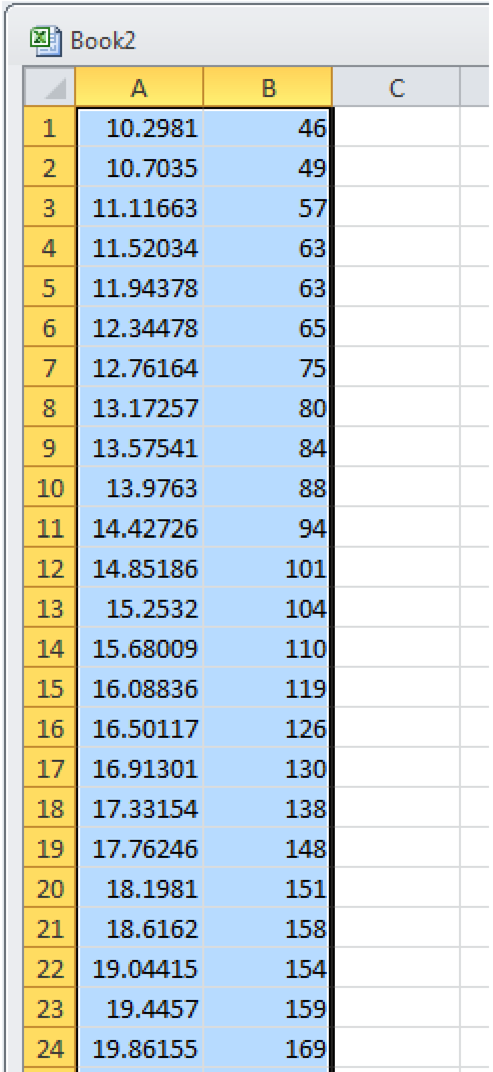
people
Person [10000]

nInfectious
23

InfectiousDS

Run: 0 Finished | Time: 200.00 | Simulation: 100% | Memory: 16M of 63M

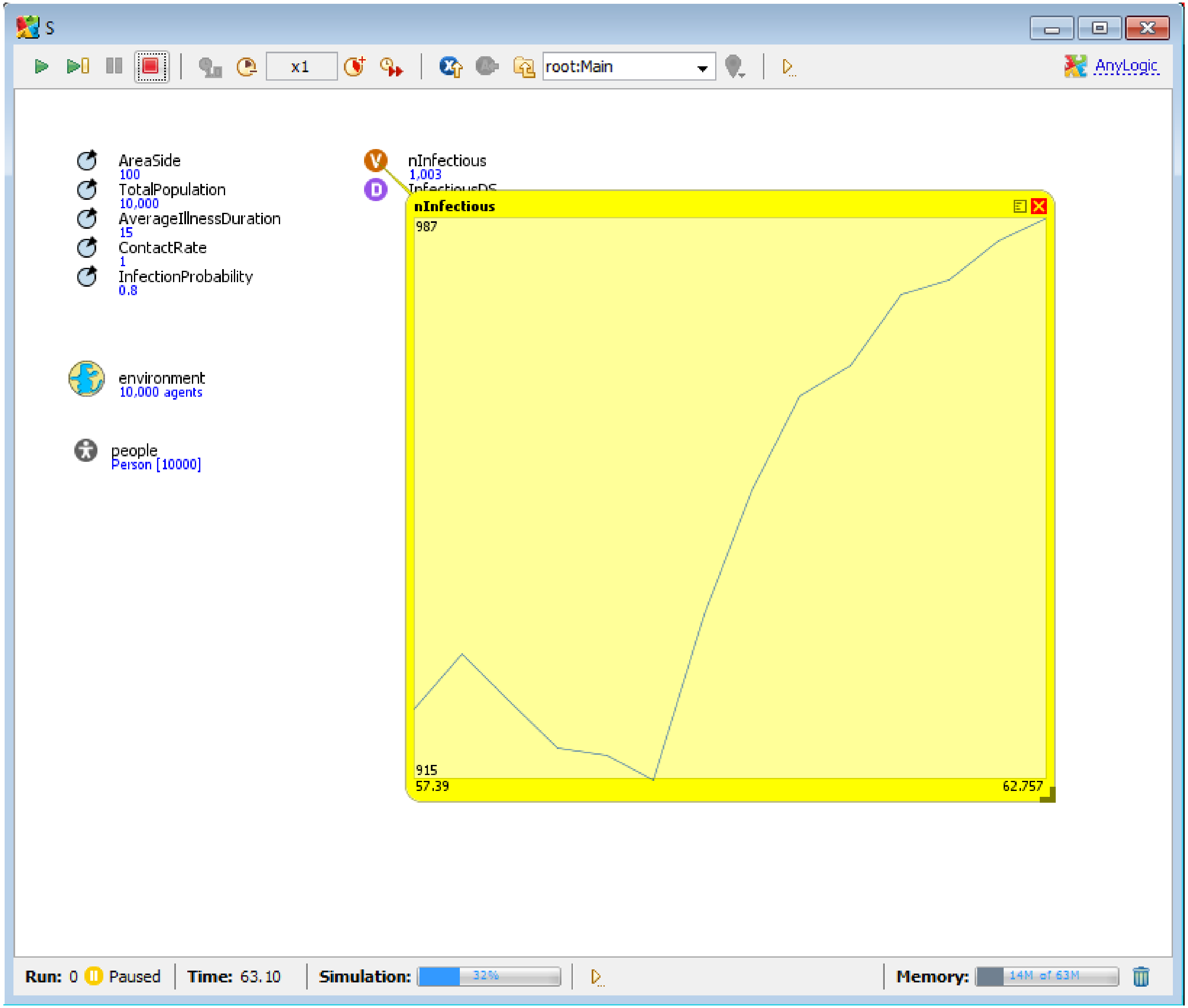
# Pasting Into Excel



Book2

| | A | B | C |
|---|---|---|---|
| 1 | 10.2981 | 46 | |
| 2 | 10.7035 | 49 | |
| 3 | 11.11663 | 57 | |
| 4 | 11.52034 | 63 | |
| 5 | 11.94378 | 63 | |
| 6 | 12.34478 | 65 | |
| 7 | 12.76164 | 75 | |
| 8 | 13.17257 | 80 | |
| 9 | 13.57541 | 84 | |
| 10 | 13.9763 | 88 | |
| 11 | 14.42726 | 94 | |
| 12 | 14.85186 | 101 | |
| 13 | 15.2532 | 104 | |
| 14 | 15.68009 | 110 | |
| 15 | 16.08836 | 119 | |
| 16 | 16.50117 | 126 | |
| 17 | 16.91301 | 130 | |
| 18 | 17.33154 | 138 | |
| 19 | 17.76246 | 148 | |
| 20 | 18.1981 | 151 | |
| 21 | 18.6162 | 158 | |
| 22 | 19.04415 | 154 | |
| 23 | 19.4457 | 159 | |
| 24 | 19.86155 | 169 | |

# Press Red “Stop” Button to Terminate Execution



AreaSide
100

TotalPopulation
10,000

AverageIllnessDuration
15

ContactRate
1

InfectionProbability
0.8

environment
10,000 agents

people
Person [10000]

nInfectious
1,003

InfectiousDS

Run: 0 Paused | Time: 63.10 | Simulation: 32% | Memory: 14M of 63M

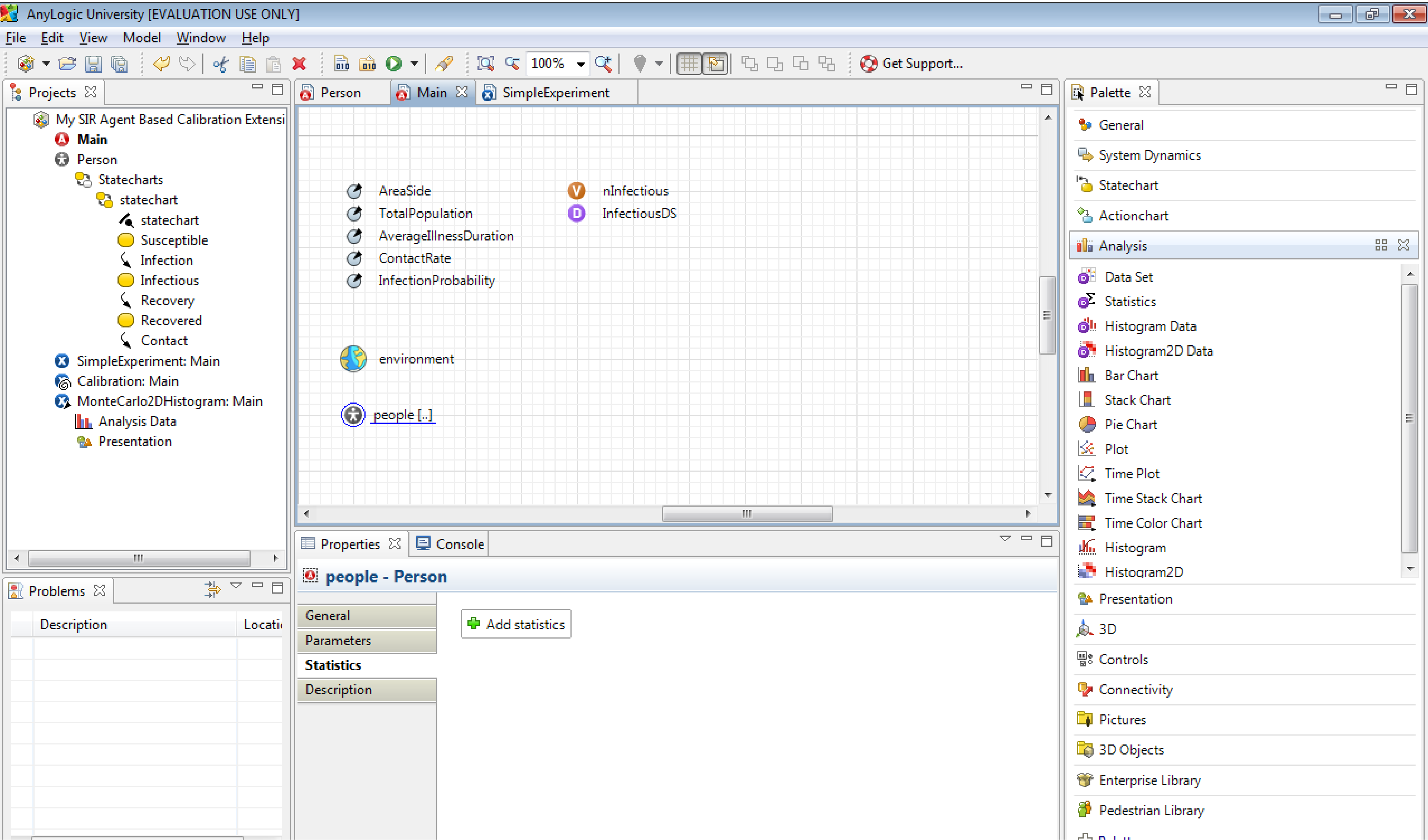
# Techniques for Collecting & Outputting Data

- Ad-Hoc Exports from variables
- Pre-Prepared methods
  - Statistics
  - Charts
  - Manual copies from visible datasets
  - Export to files
  - Writing to console
  - Export to databases
  - [AnyLogic Professional] Dataset archiving
  - Capturing images of graphs

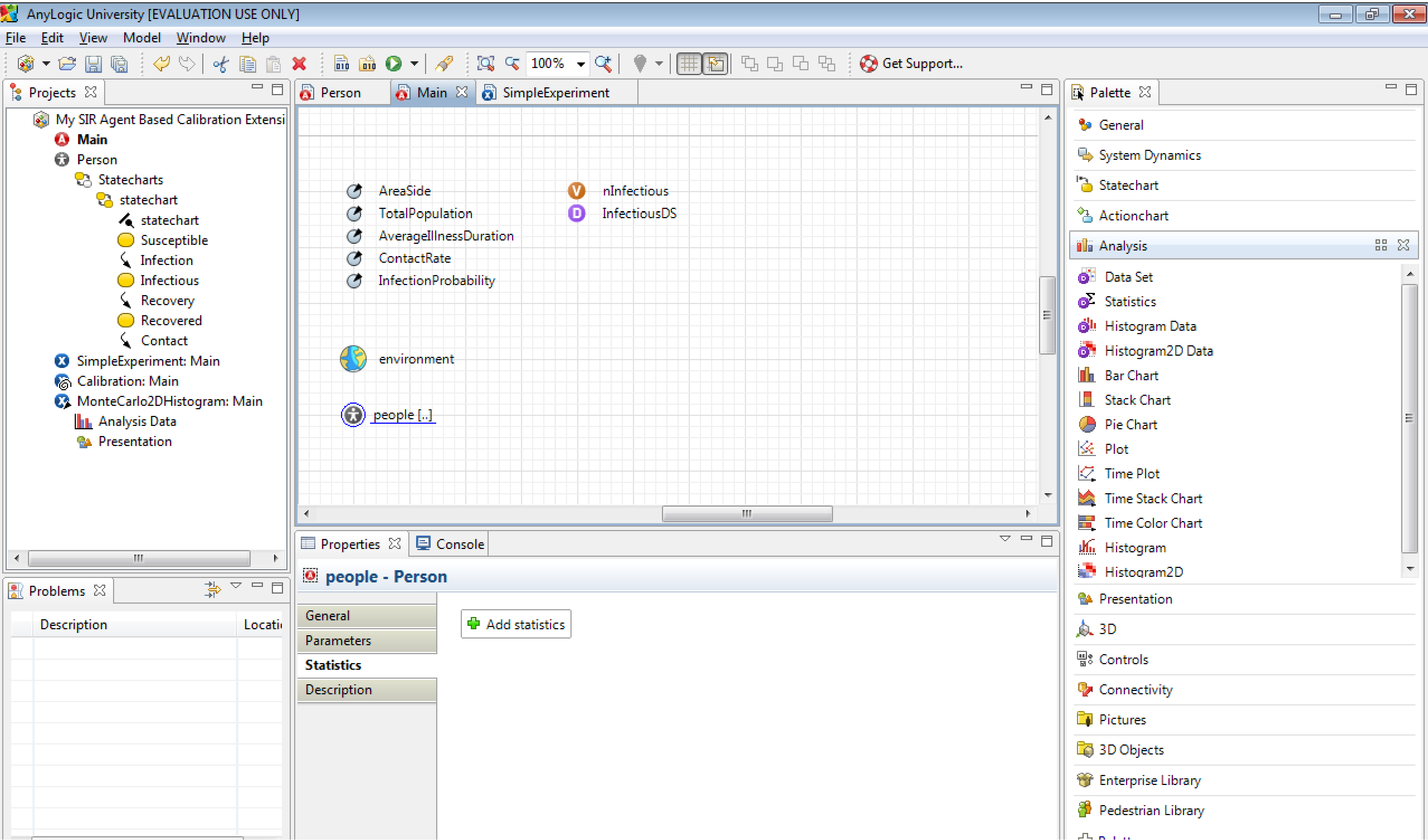
# Statistics & Charts

- A population of agents can have associated statistics that calculate values
- Examples of things that can be computed with using AnyLogic’s statistics
  - – Count of agents in the population for which certain condition (“predicate”) evaluates to true
  - – Function of the values of some expression over the population
    - Maximum value
    - Minimum value
    - Average value
    - Sum  (total) over population
  - – Statistics can be defined as properties of the population

# Select “People”, and Choose “Statistics”

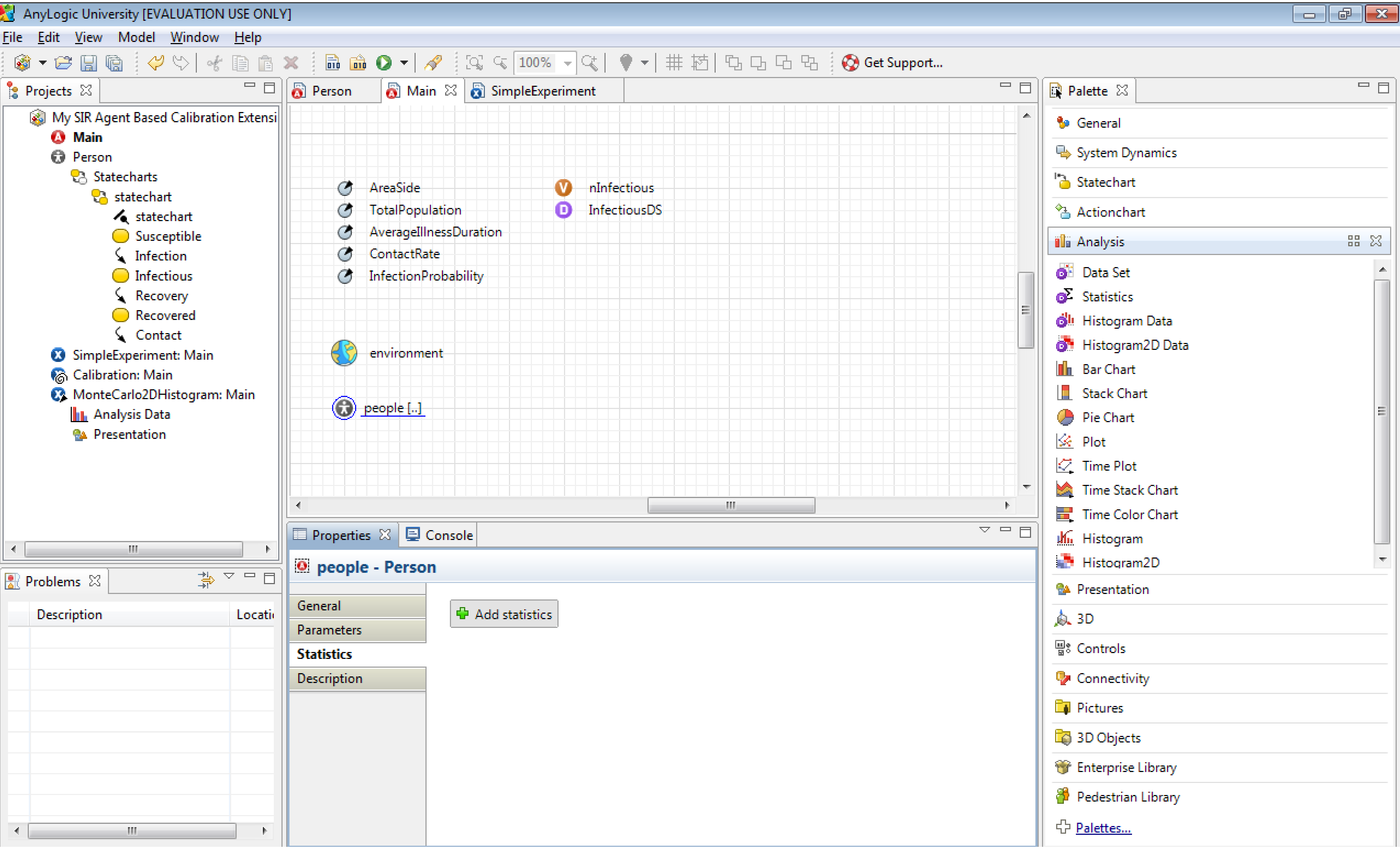


# Also, Expand “Statechart” Under “Person”

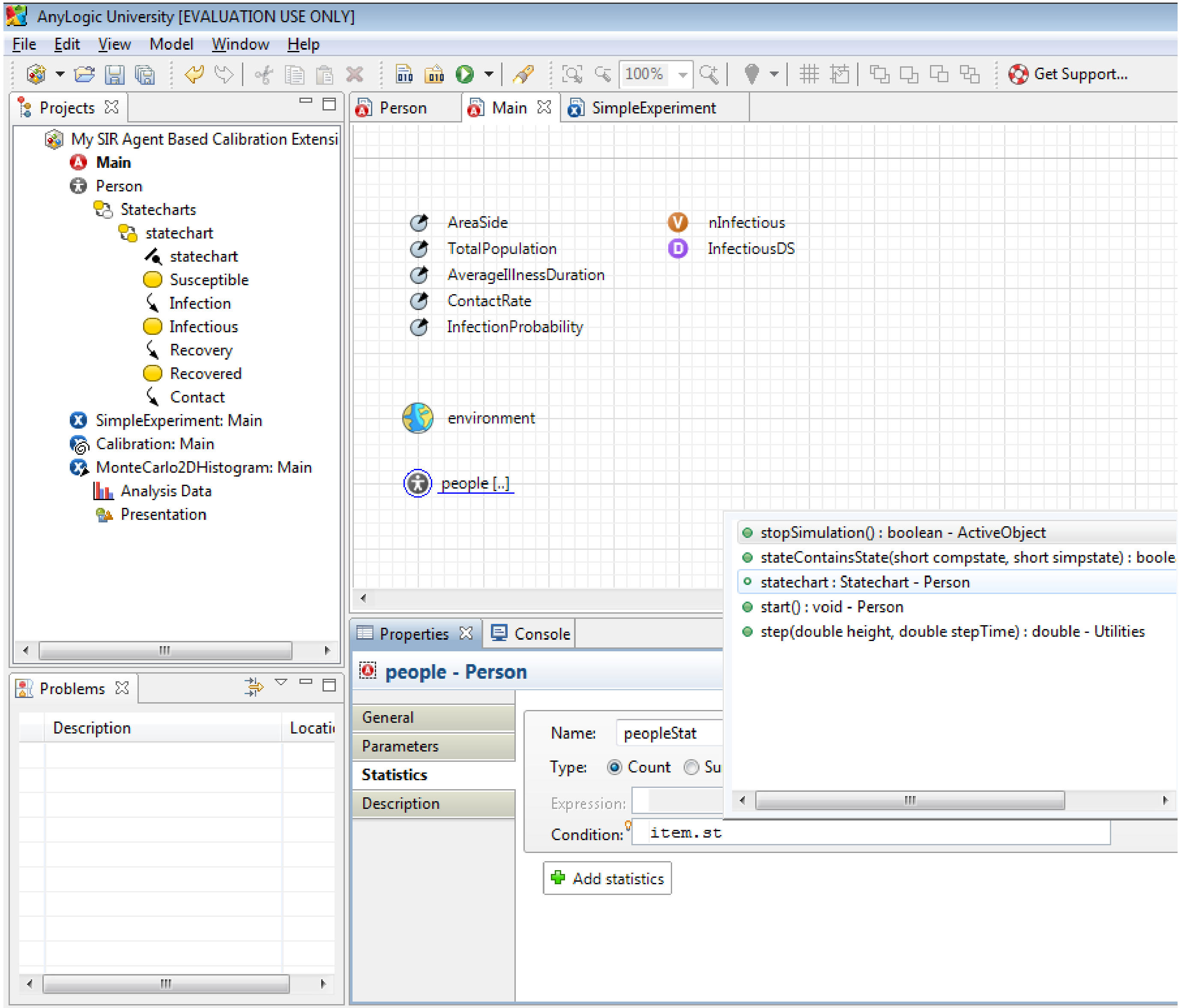


AnyLogic University [EVALUATION USE ONLY]

# Click "Add Statistics"



# Fill in the “Condition” (Predicate) on Person



AnyLogic University [EVALUATION USE ONLY]

- My SIR Agent Based Calibration Extensi
  - Main
  - Person
    - Statecharts
      - statechart
        - statechart
        - Susceptible
        - Infection
        - Infectious
        - Recovery
        - Recovered
        - Contact
  - SimpleExperiment: Main
  - Calibration: Main
  - MonteCarlo2DHistogram: Main
    - Analysis Data
    - Presentation

- stopSimulation() : boolean - ActiveObject
- stateContainsState(short compstate, short simpstate) : boole
- statechart : Statechart - Person
- start() : void - Person
- step(double height, double stepTime) : double - Utilities

people - Person

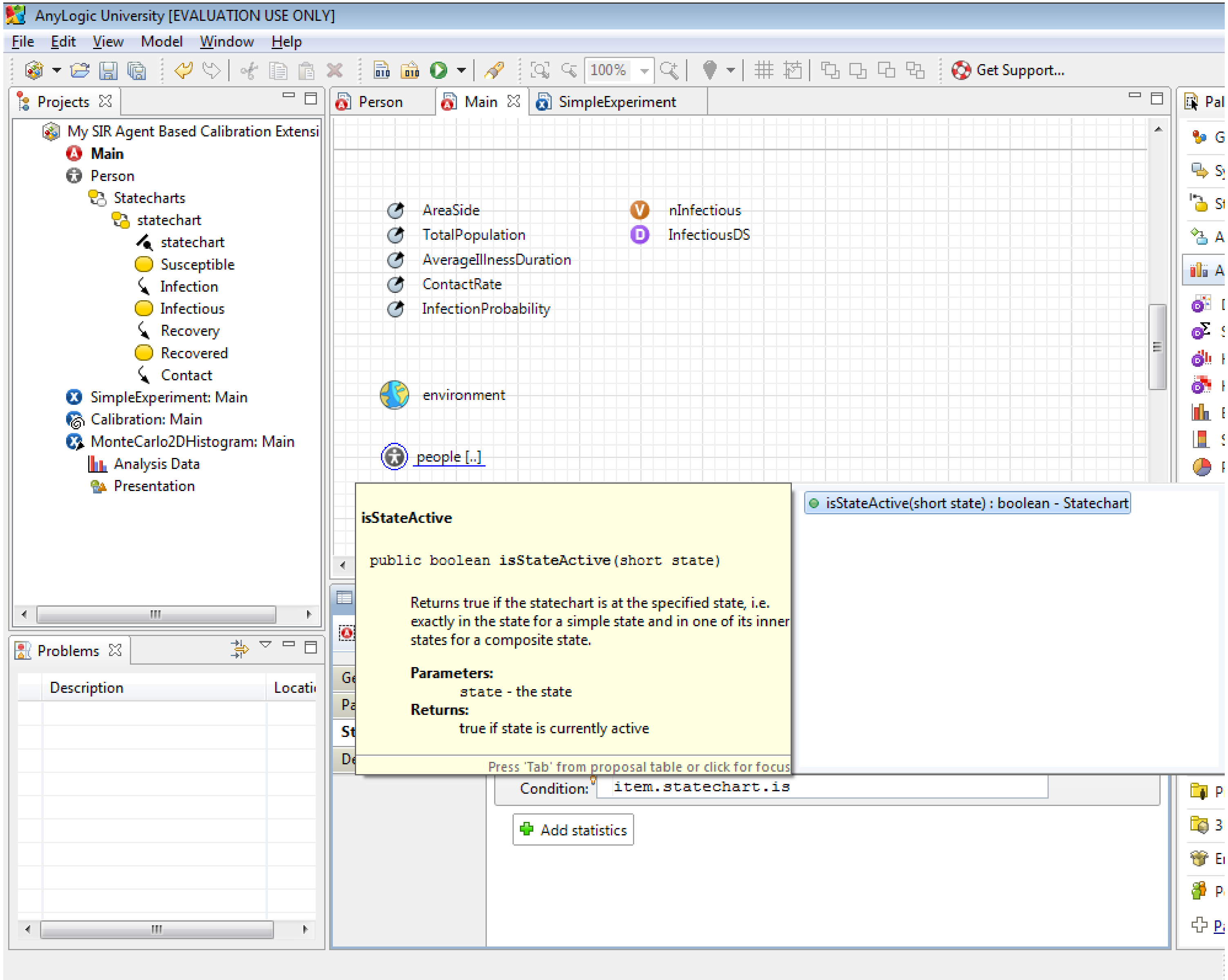
Name: peopleStat

Type: ● Count ○ Su

Expression:

Condition: item.st

# Continue Typing



isStateActive(short state) : boolean - Statechart

**isStateActive**

```
public boolean isStateActive(short state)
```

Returns true if the statechart is at the specified state, i.e. exactly in the state for a simple state and in one of its inner states for a composite state.

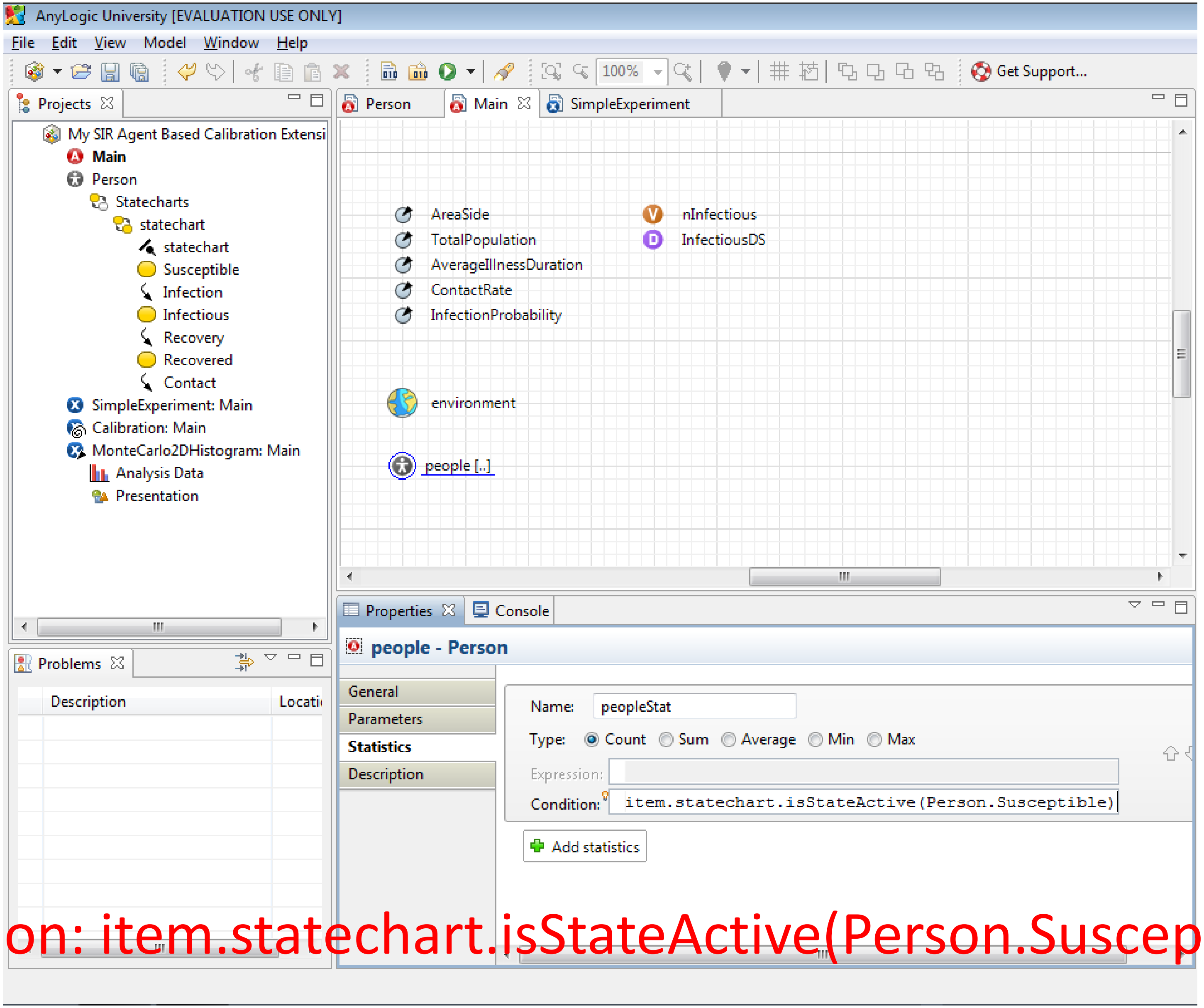
**Parameters:**
`state` - the state

**Returns:**
true if state is currently active

Press 'Tab' from proposal table or click for focus

Condition: `item.statechart.is`

# Full Expression



people - Person

Statistics

Name: peopleStat

Type: ● Count ○ Sum ○ Average ○ Min ○ Max

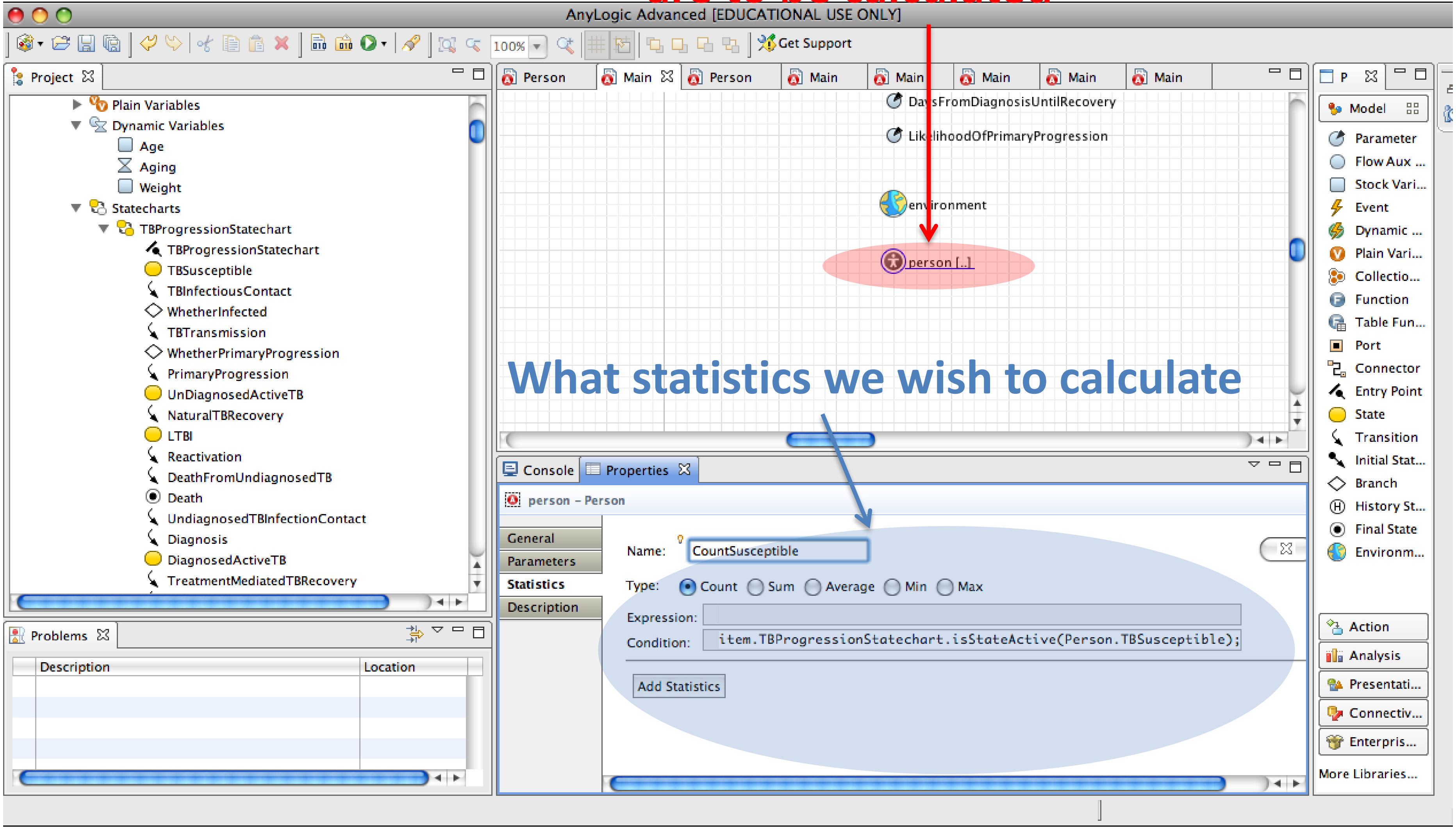
Expression:

Condition: `item.statechart.isStateActive(Person.Susceptible)`

Expression: item.statechart.isStateActive(Person.Susceptible)

# Example Statistics

The population in which the statistics are to be calculated



What statistics we wish to calculate

person – Person

Statistics

Name: CountSusceptible

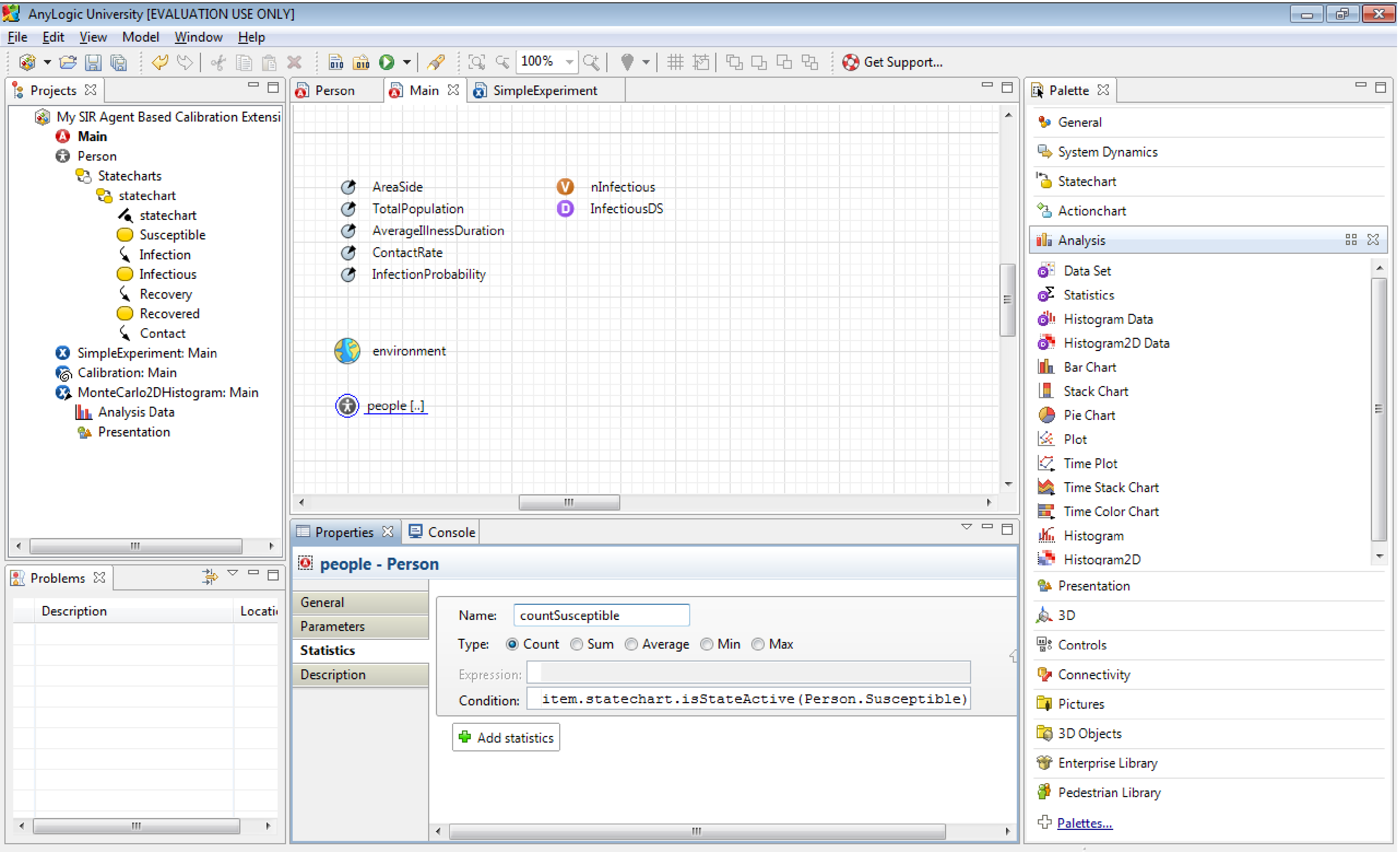
Type: ● Count ○ Sum ○ Average ○ Min ○ Max

Expression:

Condition: `item.TBProgressionStatechart.isStateActive(Person.TBSusceptible);`

Add Statistics

# Name the Statistic "countSusceptible"



**people - Person**

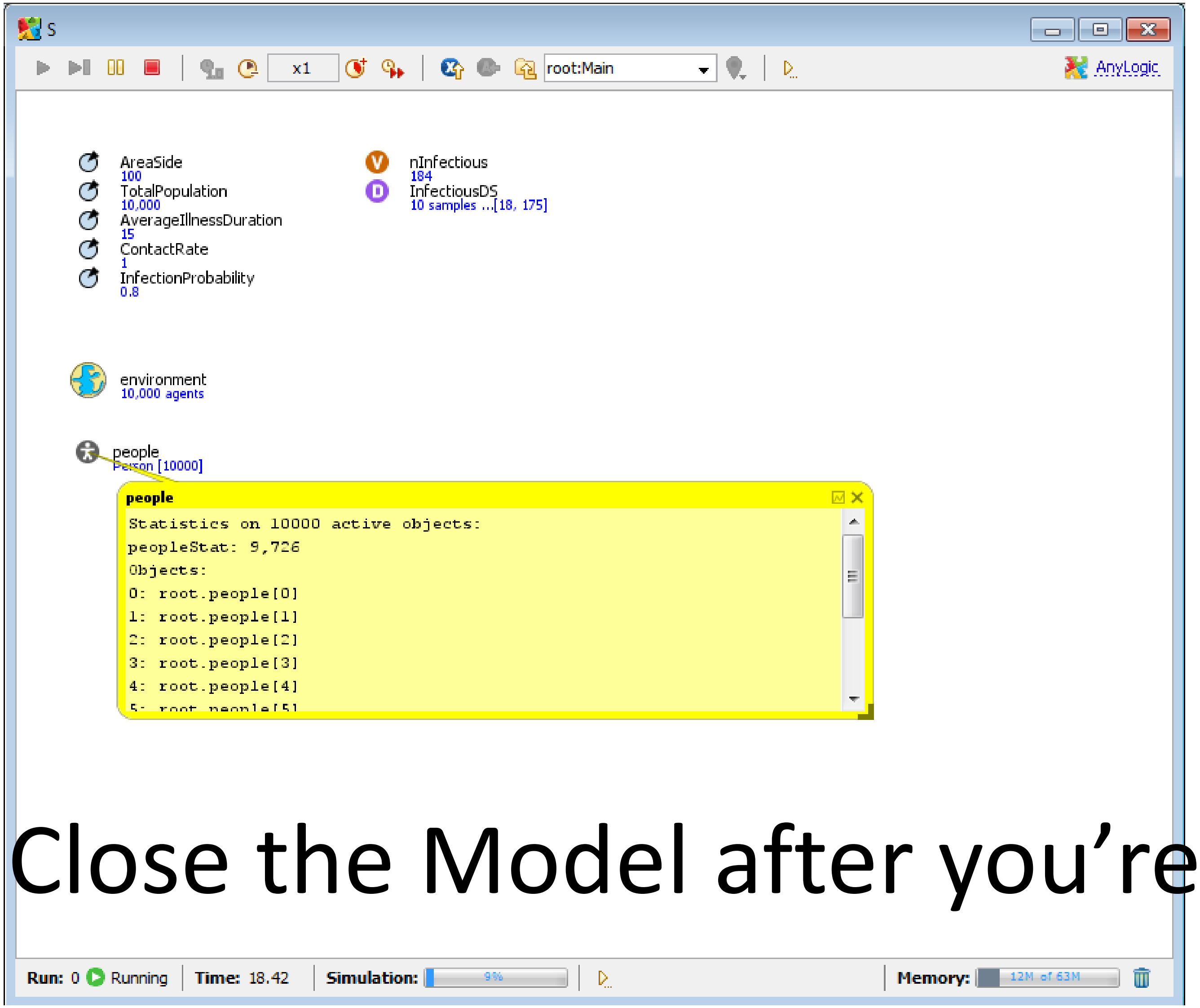
Name: countSusceptible

Type: ● Count ○ Sum ○ Average ○ Min ○ Max

Expression:

Condition: `item.statechart.isStateActive(Person.Susceptible)`

# Run the Model, and Click on “people” The Statistic should be Visible



AreaSide
100
TotalPopulation
10,000
AverageIllnessDuration
15
ContactRate
1
InfectionProbability
0.8

nInfectious
184
InfectiousDS
10 samples ...[18, 175]

environment
10,000 agents

people
Person [10000]

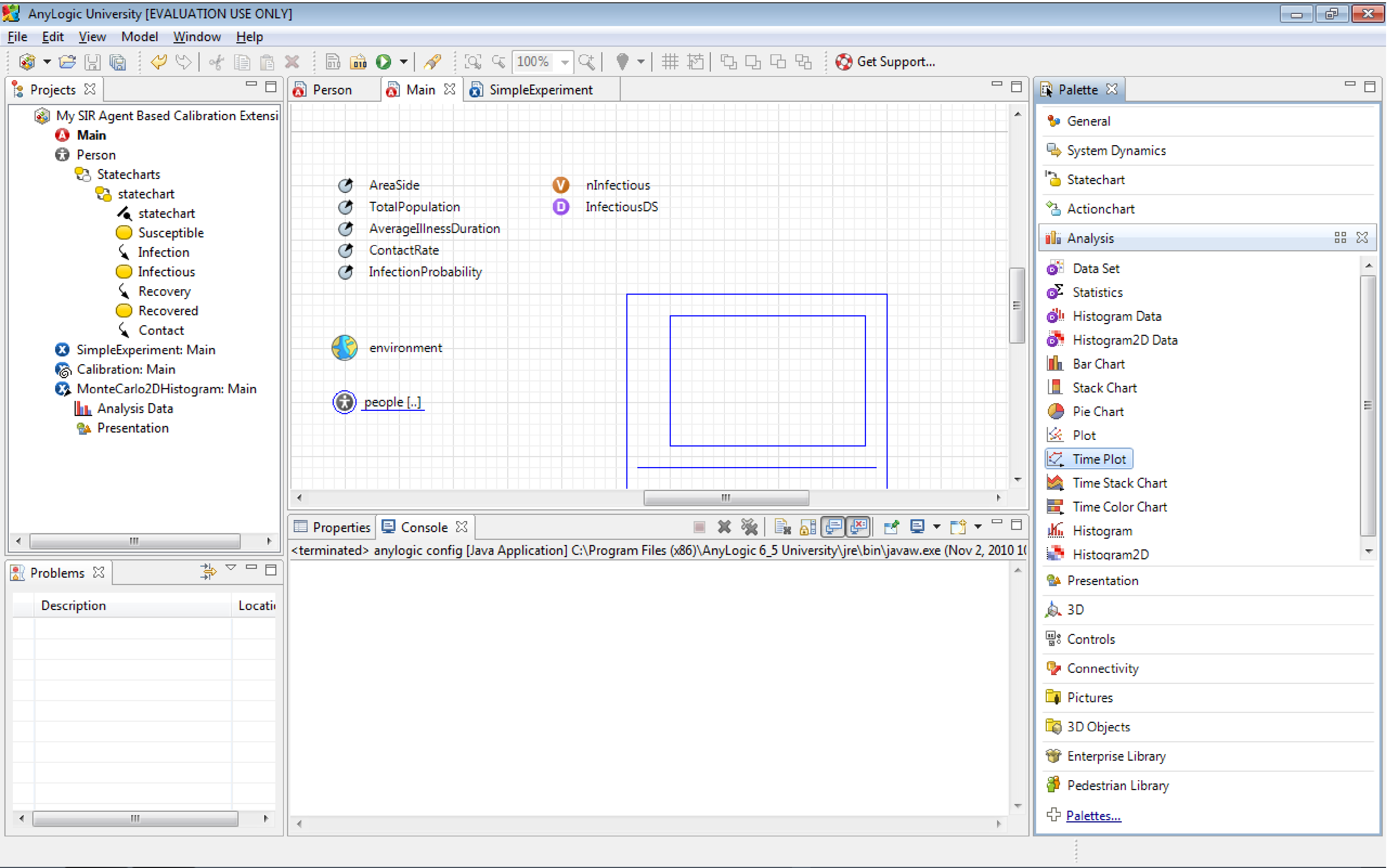
**people**

```
Statistics on 10000 active objects:
peopleStat: 9,726
Objects:
0: root.people[0]
1: root.people[1]
2: root.people[2]
3: root.people[3]
4: root.people[4]
5: root.people[5]
```

**Run:** 0 Running | **Time:** 18.42 | **Simulation:** 9% | **Memory:** 12M of 63M

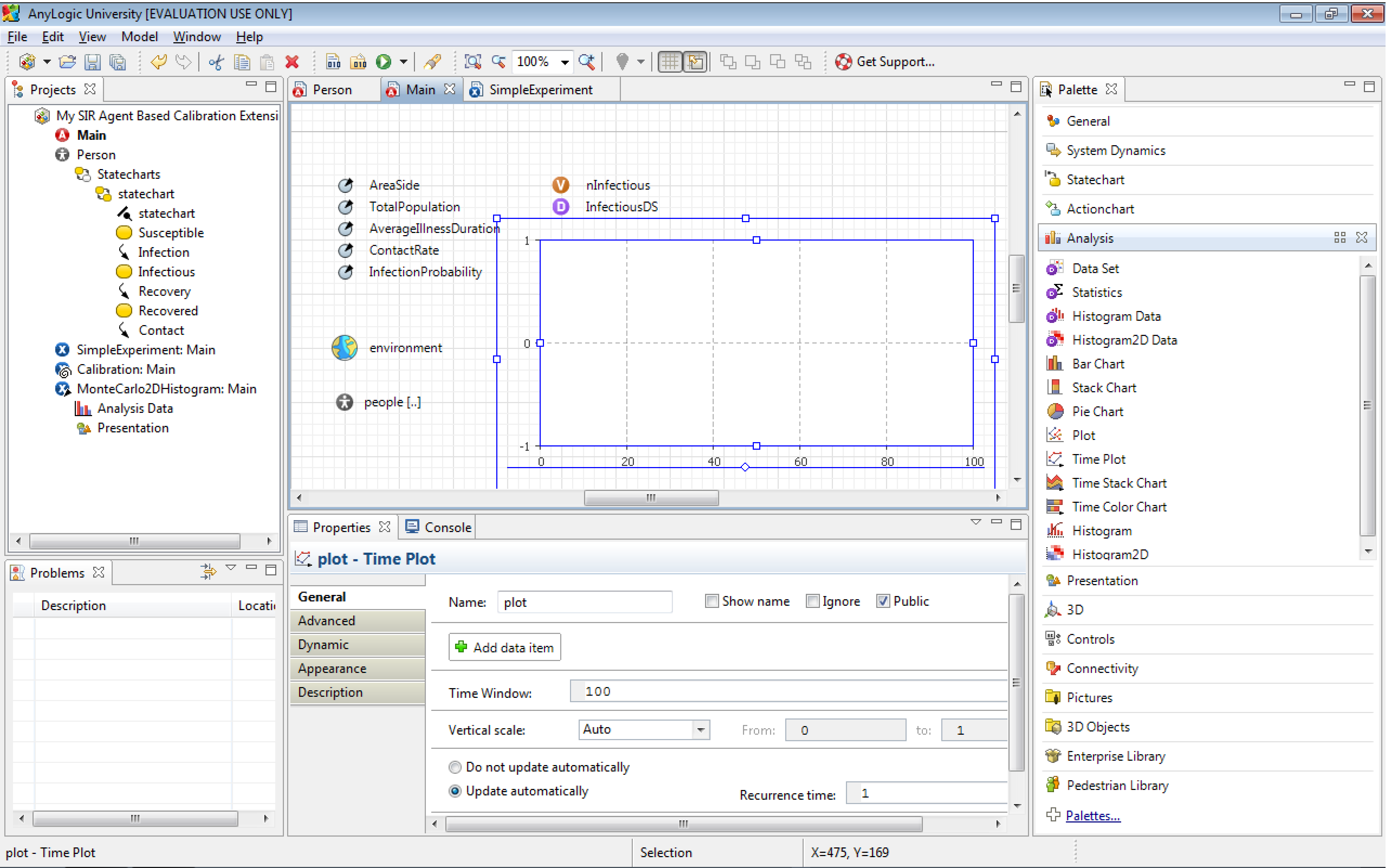
Close the Model after you’re done

# Drag a “Time Plot” from the Palette to the “Main” Canvas



```
<terminated> anylogic config [Java Application] C:\Program Files (x86)\AnyLogic 6_5 University\jre\bin\javaw.exe (Nov 2, 2010 1(
```

# Enlarge the Chart



AnyLogic University [EVALUATION USE ONLY]

plot - Time Plot

Name: plot  □ Show name  □ Ignore  ☑ Public

Time Window: 100

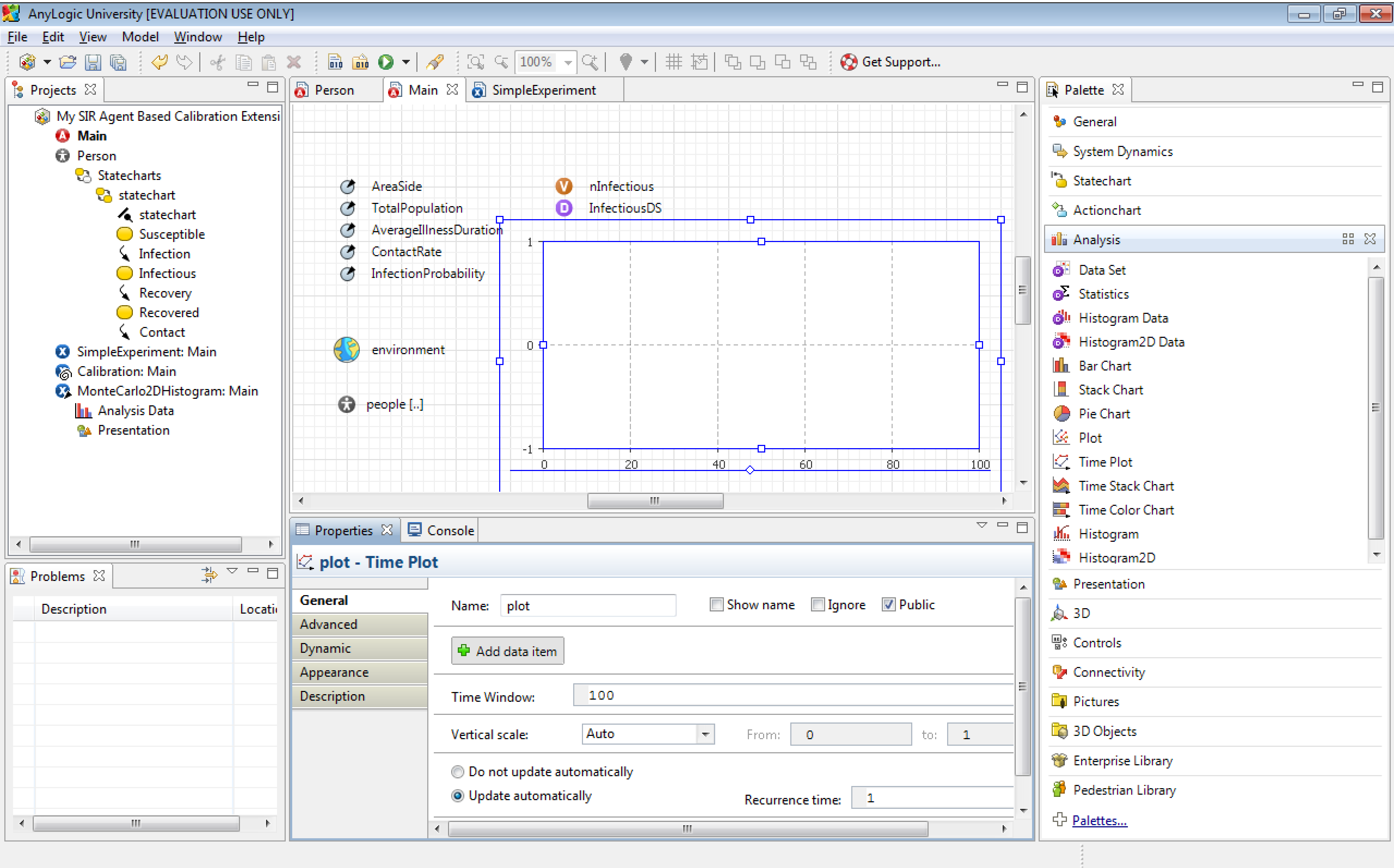
Vertical scale: Auto  From: 0  to: 1

○ Do not update automatically

● Update automatically  Recurrence time: 1

plot - Time Plot | Selection | X=475, Y=169

# Click “Add Data Item”



plot - Time Plot

Name: plot ☐ Show name ☐ Ignore ☑ Public

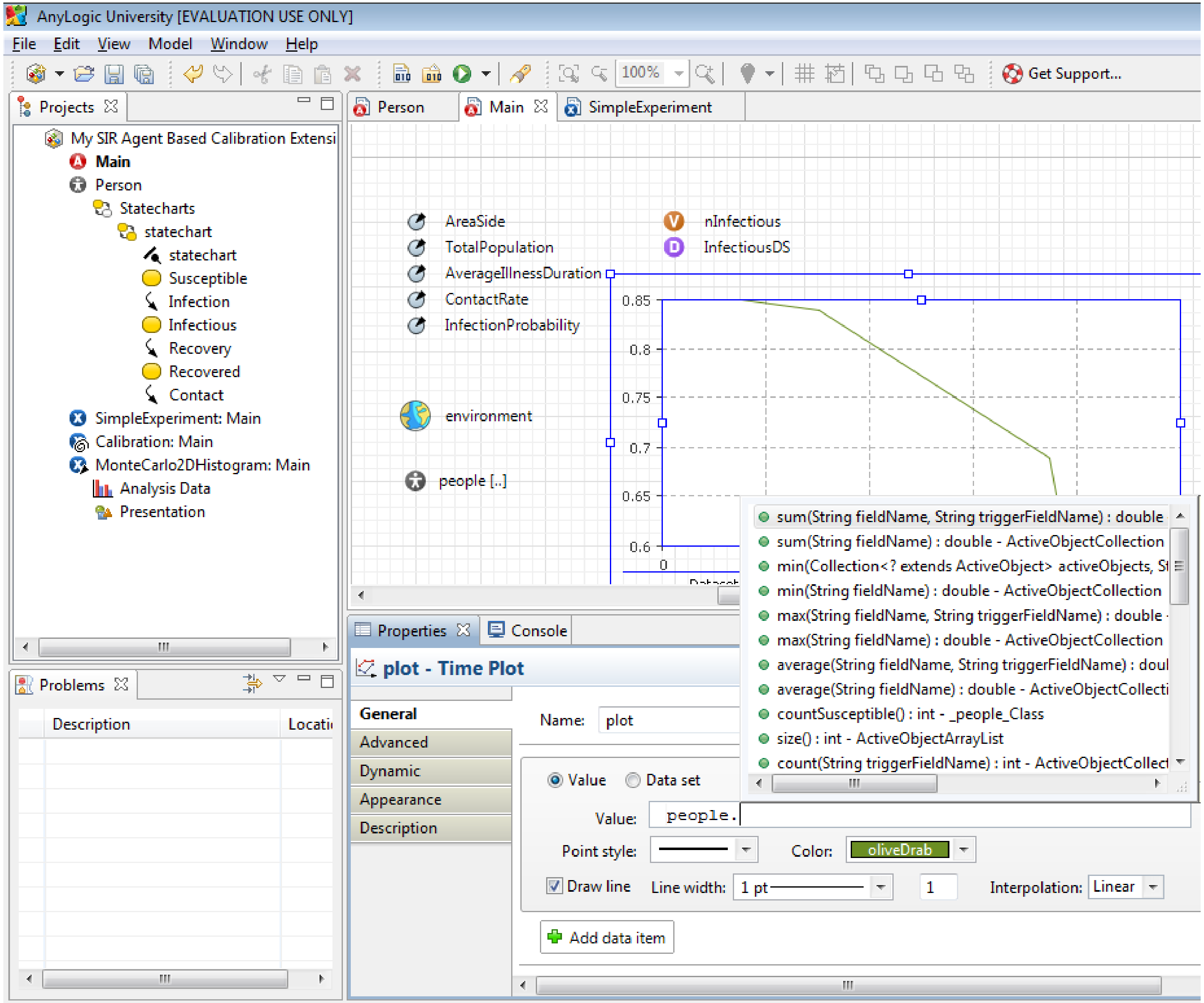
Time Window: 100

Vertical scale: Auto From: 0 to: 1

○ Do not update automatically

● Update automatically Recurrence time: 1

# Put in “people.” and Press Ctrl-Space



**plot - Time Plot**

Name: plot

◉ Value ○ Data set

Value: `people.`

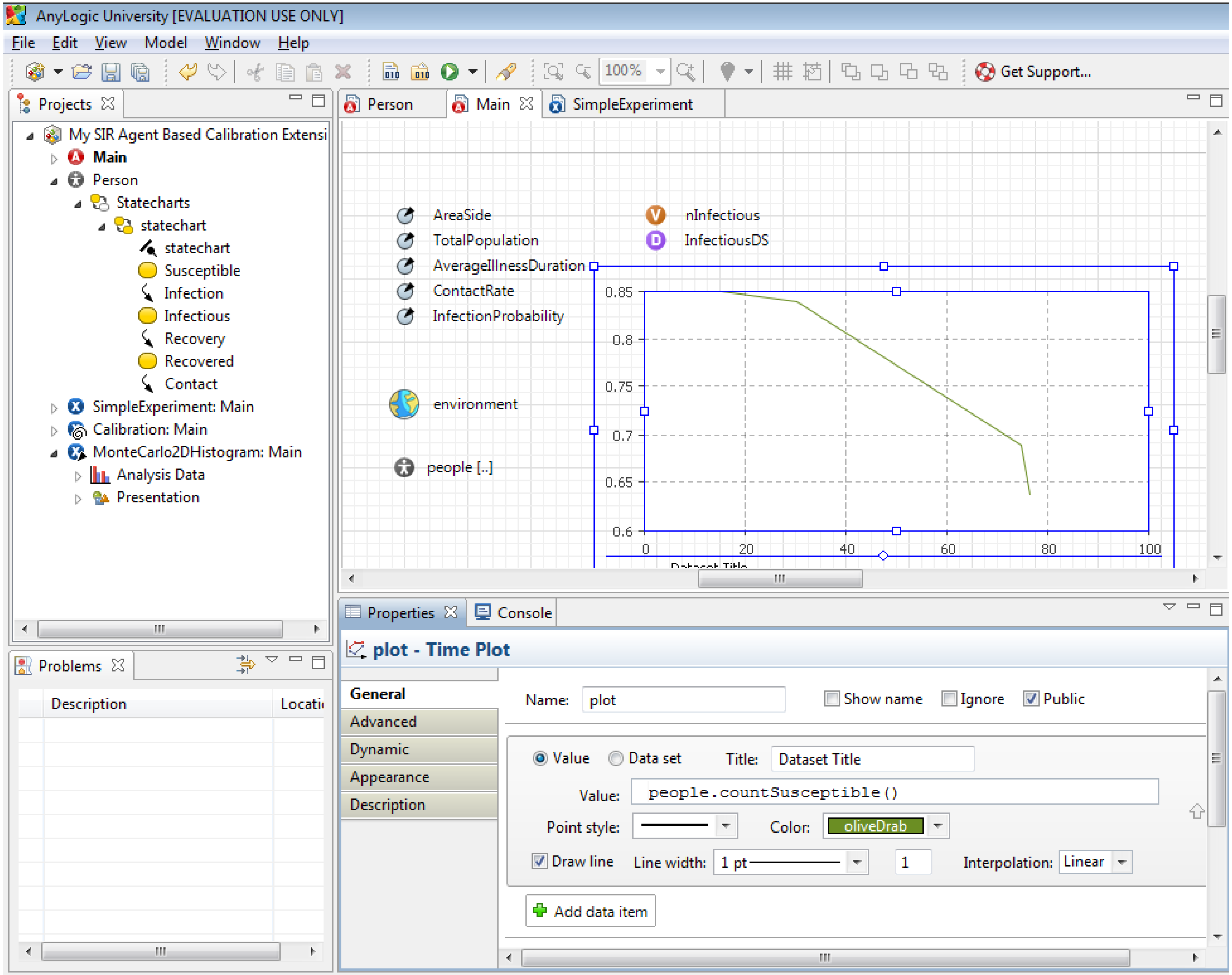
Point style: ———

Color: oliveDrab

☑ Draw line Line width: 1 pt 1 Interpolation: Linear

```
sum(String fieldName, String triggerFieldName) : double
sum(String fieldName) : double - ActiveObjectCollection
min(Collection<? extends ActiveObject> activeObjects, St
min(String fieldName) : double - ActiveObjectCollection
max(String fieldName, String triggerFieldName) : double -
max(String fieldName) : double - ActiveObjectCollection
average(String fieldName, String triggerFieldName) : doul
average(String fieldName) : double - ActiveObjectCollecti
countSusceptible() : int - _people_Class
size() : int - ActiveObjectArrayList
count(String triggerFieldName) : int - ActiveObjectCollect
```

# Choose “Count Susceptible”



Name: plot  □ Show name  □ Ignore  ☑ Public

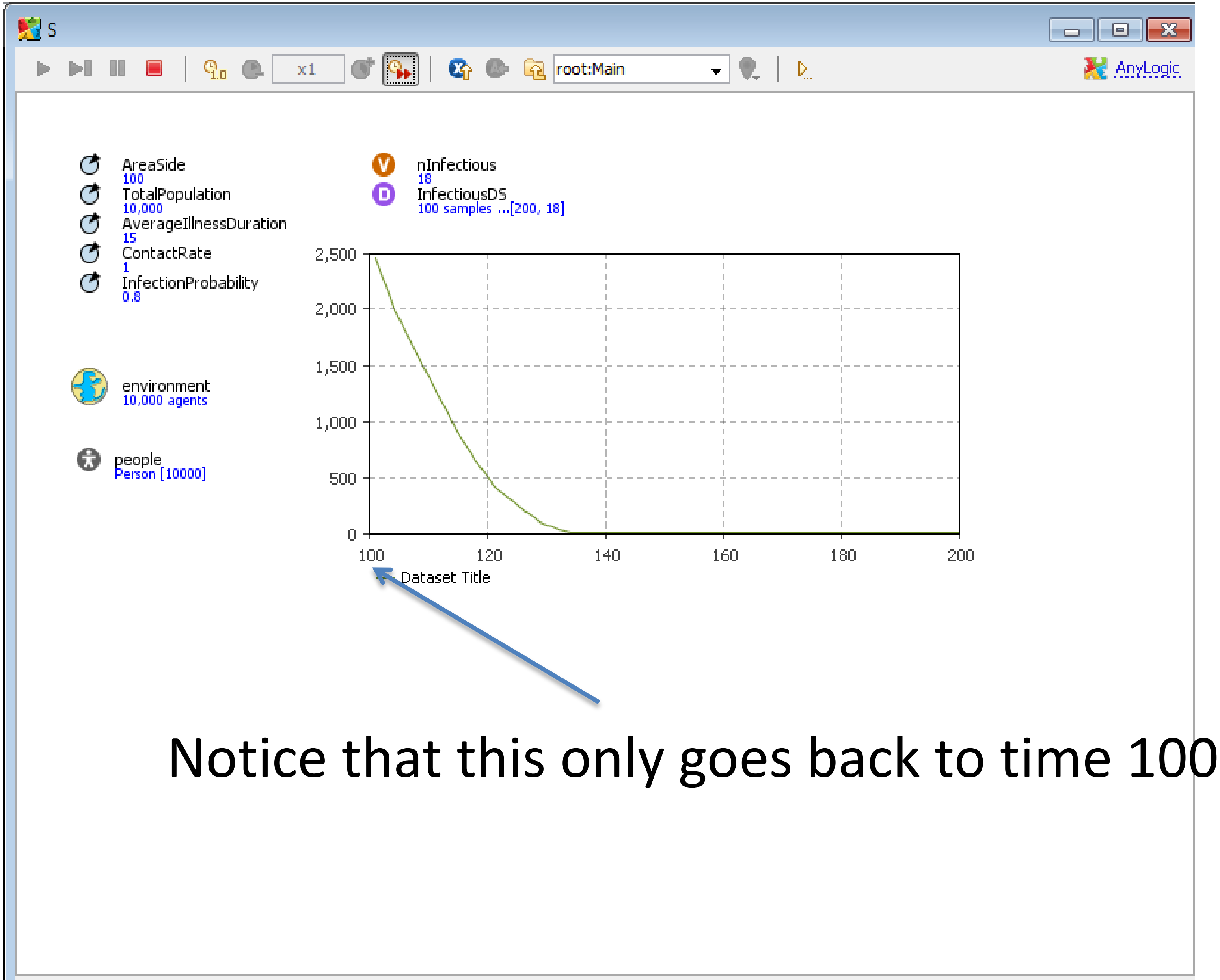
● Value  ○ Data set  Title: Dataset Title

Value: `people.countSusceptible()`

Point style: ——  Color: oliveDrab

☑ Draw line  Line width: 1 pt  1  Interpolation: Linear

# Now Run the Model



AreaSide
100

TotalPopulation
10,000

AverageIllnessDuration
15

ContactRate
1

InfectionProbability
0.8

nInfectious
18

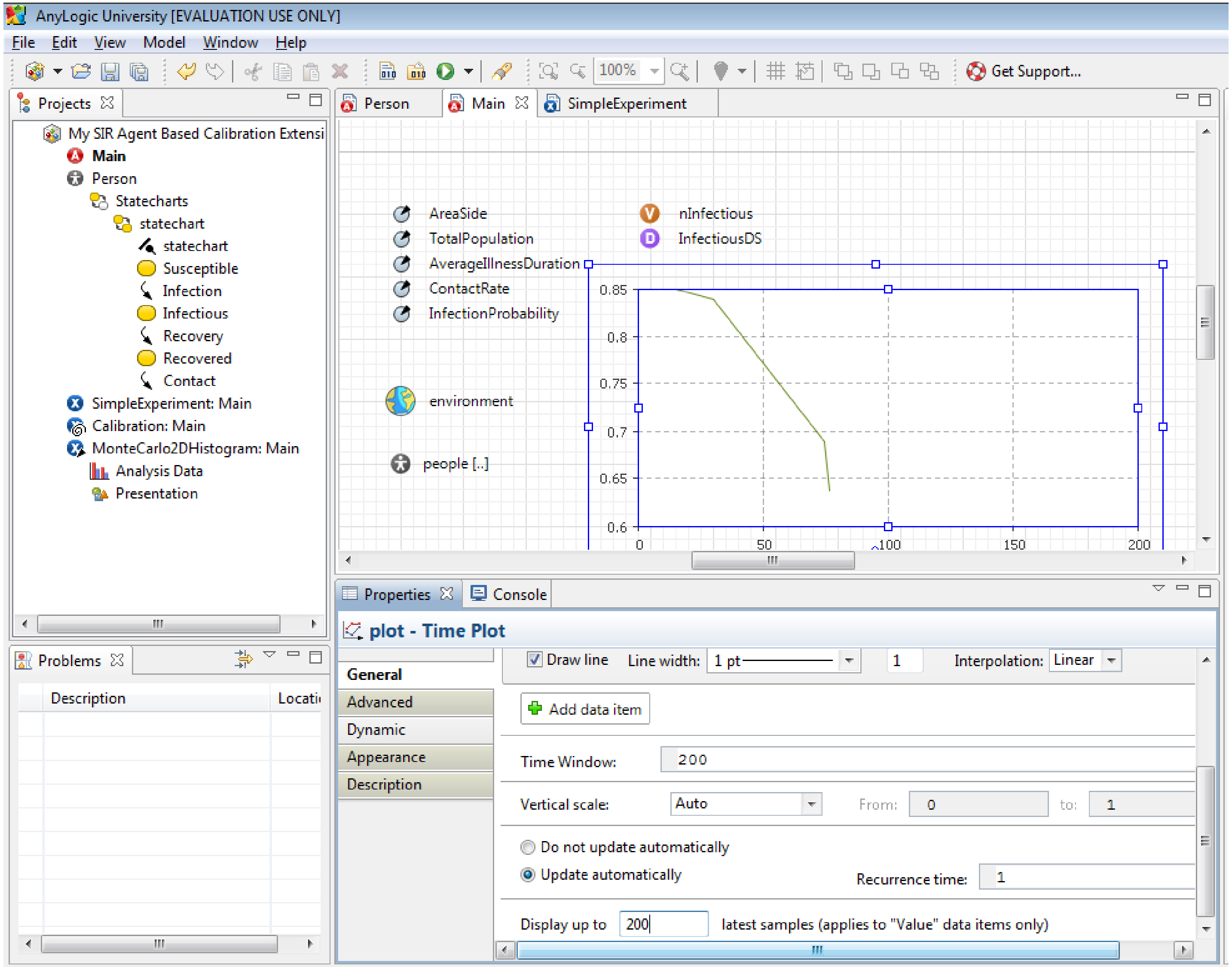
InfectiousDS
100 samples ...[200, 18]

environment
10,000 agents

people
Person [10000]

Notice that this only goes back to time 100

# Stop the Simulation, and Click on the Plot. Change Time Window & Display Size to 200



☑ Draw line   Line width: 1 pt   1   Interpolation: Linear

Time Window: 200

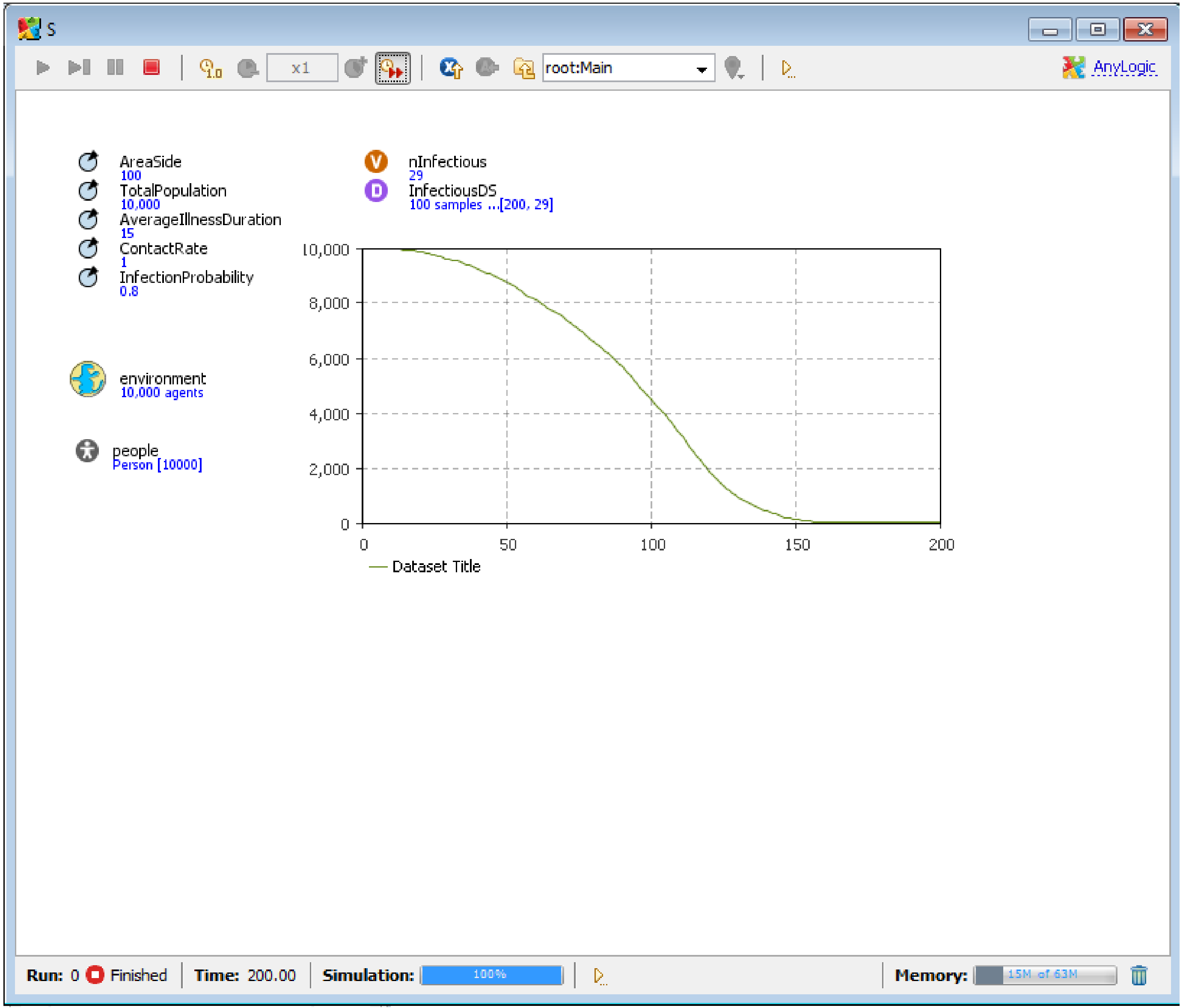
Vertical scale: Auto   From: 0   to: 1

○ Do not update automatically

● Update automatically   Recurrence time: 1

Display up to 200 latest samples (applies to "Value" data items only)

# This Captures the Full Time Range



AreaSide
100

TotalPopulation
10,000

AverageIllnessDuration
15

ContactRate
1

InfectionProbability
0.8

environment
10,000 agents

people
Person [10000]

nInfectious
29

InfectiousDS
100 samples ...[200, 29]

Run: 0 Finished | Time: 200.00 | Simulation: 100% | Memory: 15M of 63M

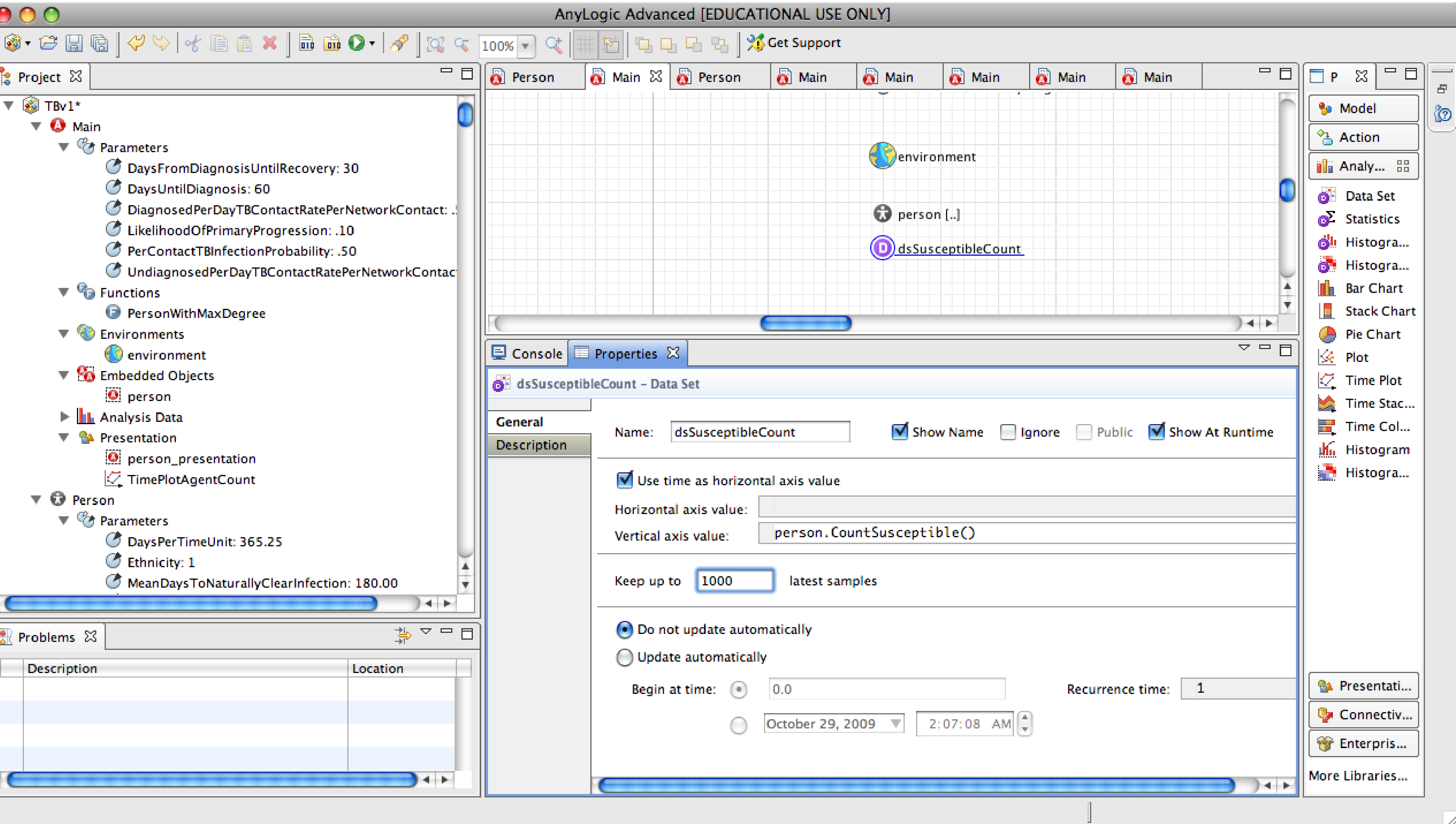
# Techniques for Collecting & Outputting Data

- Ad-Hoc Exports from variables
- Pre-Prepared methods
  - Statistics
  - Charts
  - Manual copies from visible datasets
  - Export to files
  - Writing to console
  - Export to databases
  - [AnyLogic Professional] Dataset archiving
  - Capturing images of graphs

# Datasets

- Datasets store recent values of some quantities from the model
- Datasets can be exported easily using custom code
  - This can simply call the dataset’s to string method

# Output: Datasets



AnyLogic Advanced [EDUCATIONAL USE ONLY]

Project

- TBv1*
  - Main
    - Parameters
      - DaysFromDiagnosisUntilRecovery: 30
      - DaysUntilDiagnosis: 60
      - DiagnosedPerDayTBContactRatePerNetworkContact: .
      - LikelihoodOfPrimaryProgression: .10
      - PerContactTBInfectionProbability: .50
      - UndiagnosedPerDayTBContactRatePerNetworkContac
    - Functions
      - PersonWithMaxDegree
    - Environments
      - environment
    - Embedded Objects
      - person
    - Analysis Data
    - Presentation
      - person_presentation
      - TimePlotAgentCount
  - Person
    - Parameters
      - DaysPerTimeUnit: 365.25
      - Ethnicity: 1
      - MeanDaysToNaturallyClearInfection: 180.00

dsSusceptibleCount – Data Set

Name: dsSusceptibleCount ☑ Show Name □ Ignore □ Public ☑ Show At Runtime

☑ Use time as horizontal axis value

Horizontal axis value:

Vertical axis value: person.CountSusceptible()

Keep up to 1000 latest samples

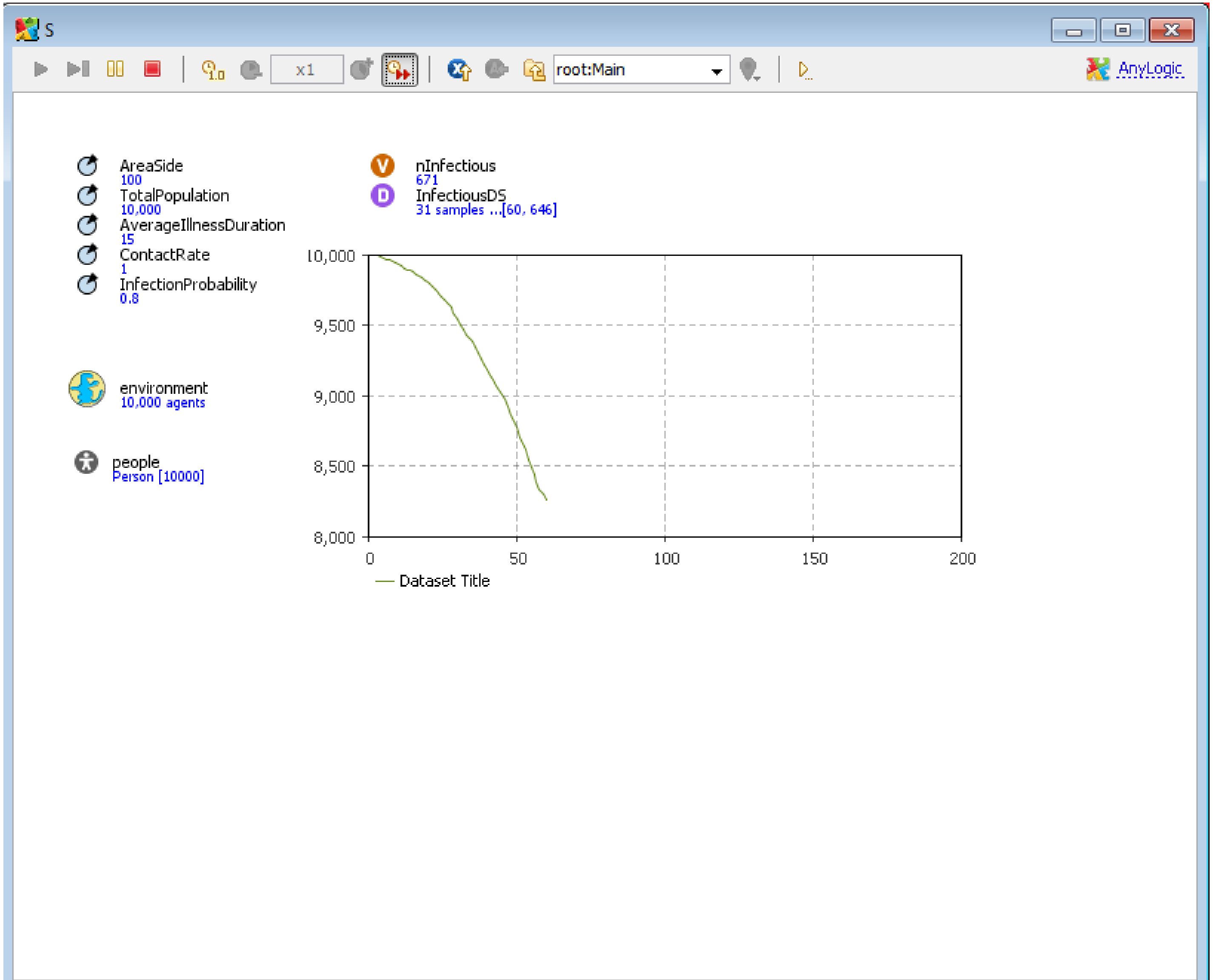
● Do not update automatically

○ Update automatically

Begin at time: ● 0.0 Recurrence time: 1

○ October 29, 2009 2:07:08 AM

# Run the Experiment & Click on "Infectious DS"



AreaSide
100

TotalPopulation
10,000

AverageIllnessDuration
15

ContactRate
1

InfectionProbability
0.8

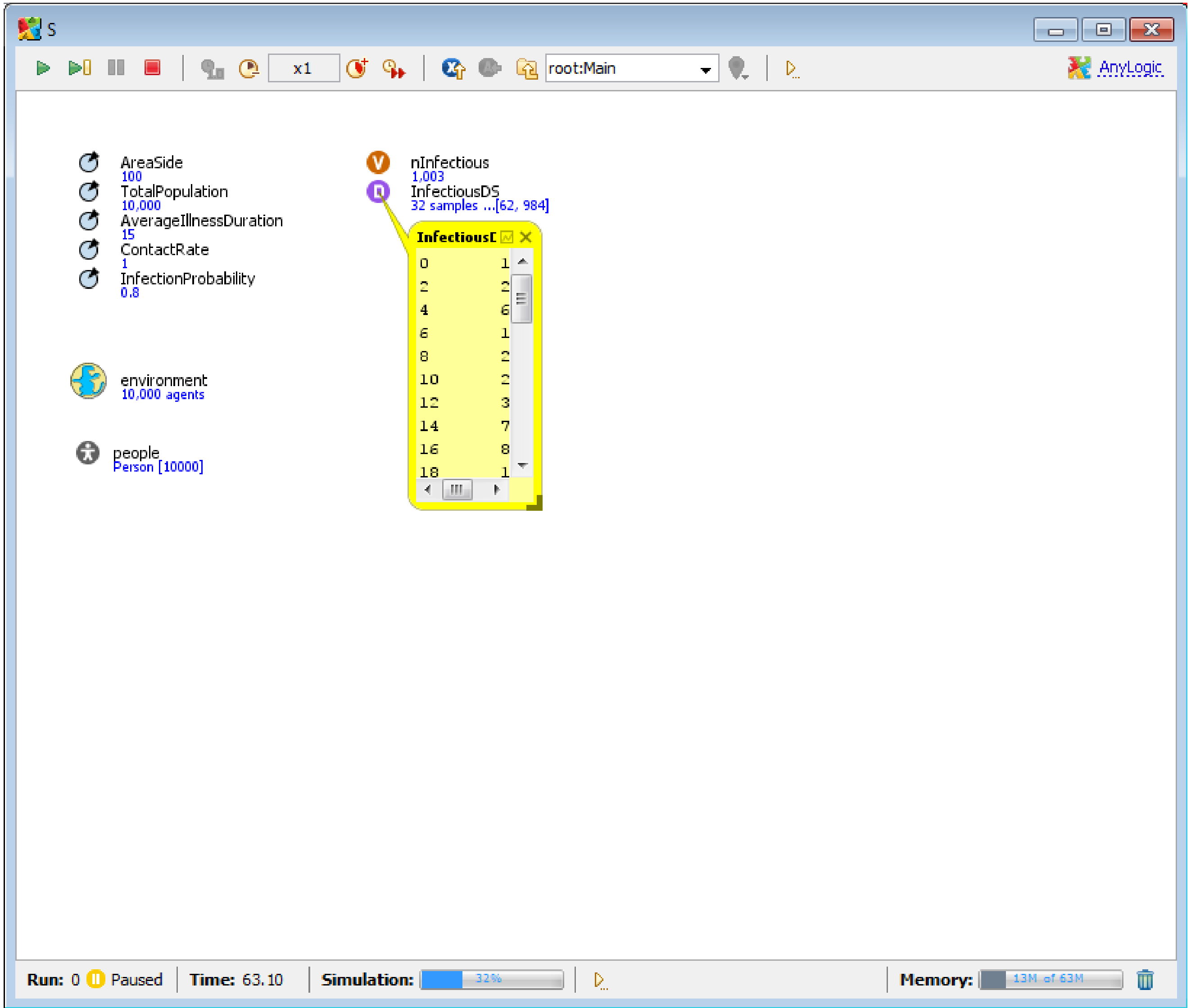
nInfectious
671

InfectiousDS
31 samples ...[60, 646]

environment
10,000 agents

people
Person [10000]

# Click on “InfectiousDS” to See Data in Dataset



AreaSide
100

TotalPopulation
10,000

AverageIllnessDuration
15

ContactRate
1

InfectionProbability
0.8

environment
10,000 agents

people
Person [10000]

nInfectious
1,003

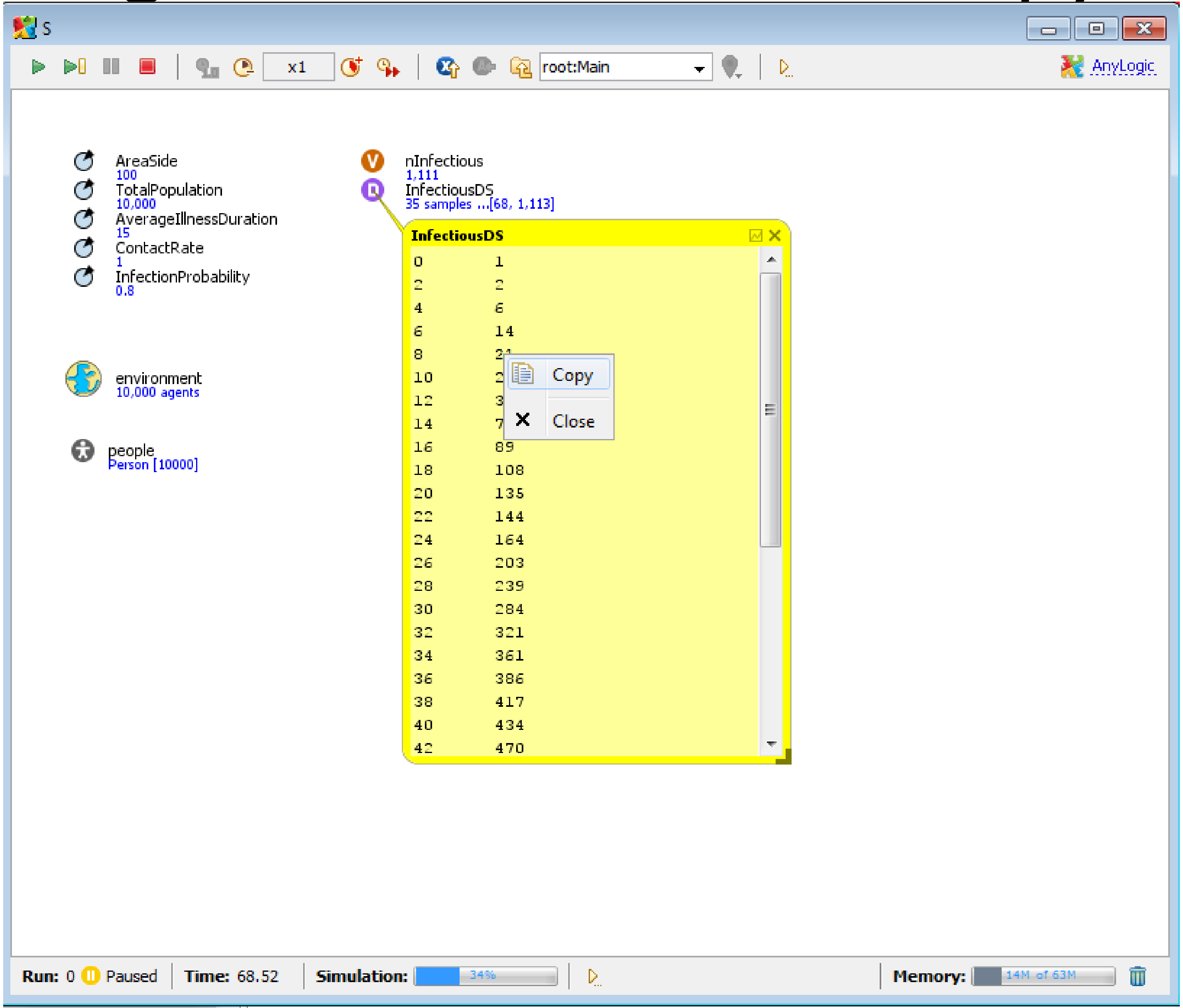
InfectiousDS
32 samples ...[62, 984]

InfectiousD

| | |
|---|---|
| 0 | 1 |
| 2 | 2 |
| 4 | 6 |
| 6 | 1 |
| 8 | 2 |
| 10 | 2 |
| 12 | 3 |
| 14 | 7 |
| 16 | 8 |
| 18 | 1 |

Run: 0 Paused | Time: 63.10 | Simulation: 32% | Memory: 13M of 63M

# Right Click and Select “Copy”



AreaSide
100
TotalPopulation
10,000
AverageIllnessDuration
15
ContactRate
1
InfectionProbability
0.8

nInfectious
1,111
InfectiousDS
35 samples ...[68, 1,113]

environment
10,000 agents

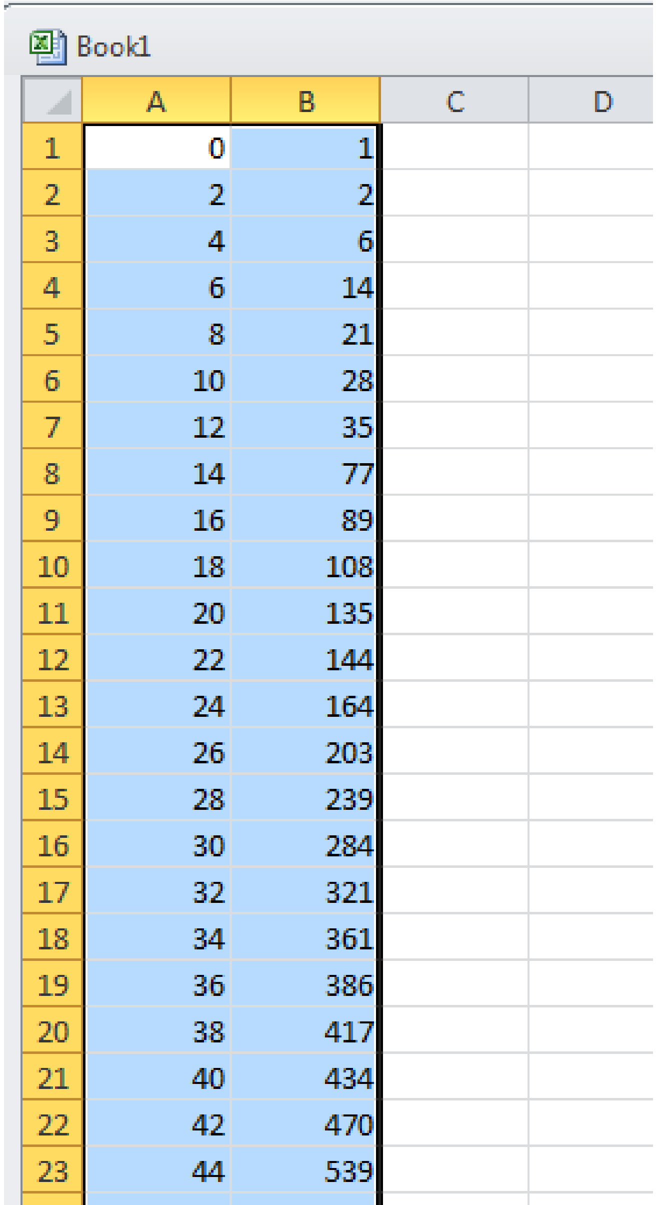
people
Person [10000]

**InfectiousDS**

```
0         1
2         2
4         6
6         14
8         21
10        2
12        3
14        7
16        89
18        108
20        135
22        144
24        164
26        203
28        239
30        284
32        321
34        361
36        386
38        417
40        434
42        470
```

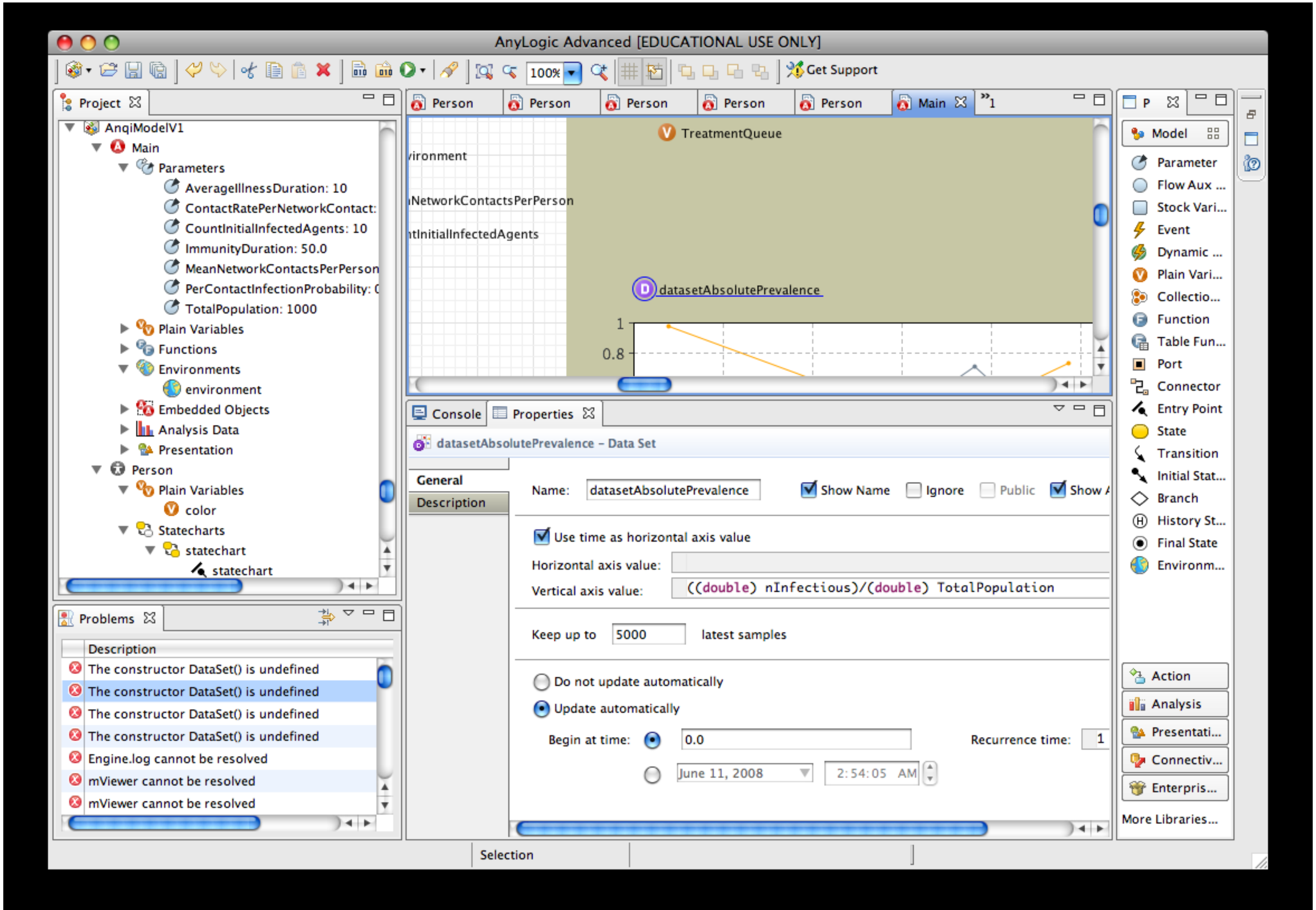
Run: 0 Paused | Time: 68.52 | Simulation: 34% | Memory: 14M of 63M

# Call Up Excel and Paste into It



| | A | B | C | D |
|---|---|---|---|---|
| 1 | 0 | 1 | | |
| 2 | 2 | 2 | | |
| 3 | 4 | 6 | | |
| 4 | 6 | 14 | | |
| 5 | 8 | 21 | | |
| 6 | 10 | 28 | | |
| 7 | 12 | 35 | | |
| 8 | 14 | 77 | | |
| 9 | 16 | 89 | | |
| 10 | 18 | 108 | | |
| 11 | 20 | 135 | | |
| 12 | 22 | 144 | | |
| 13 | 24 | 164 | | |
| 14 | 26 | 203 | | |
| 15 | 28 | 239 | | |
| 16 | 30 | 284 | | |
| 17 | 32 | 321 | | |
| 18 | 34 | 361 | | |
| 19 | 36 | 386 | | |
| 20 | 38 | 417 | | |
| 21 | 40 | 434 | | |
| 22 | 42 | 470 | | |
| 23 | 44 | 539 | | |

# Dataset Properties



datasetAbsolutePrevalence – Data Set

Name: datasetAbsolutePrevalence ☑ Show Name ☐ Ignore ☐ Public ☑ Show A

☑ Use time as horizontal axis value

Horizontal axis value:

Vertical axis value: ((double) nInfectious)/(double) TotalPopulation

Keep up to 5000 latest samples

○ Do not update automatically

● Update automatically

Begin at time: ● 0.0 Recurrence time: 1

○ June 11, 2008 2:54:05 AM

The constructor DataSet() is undefined

The constructor DataSet() is undefined

The constructor DataSet() is undefined

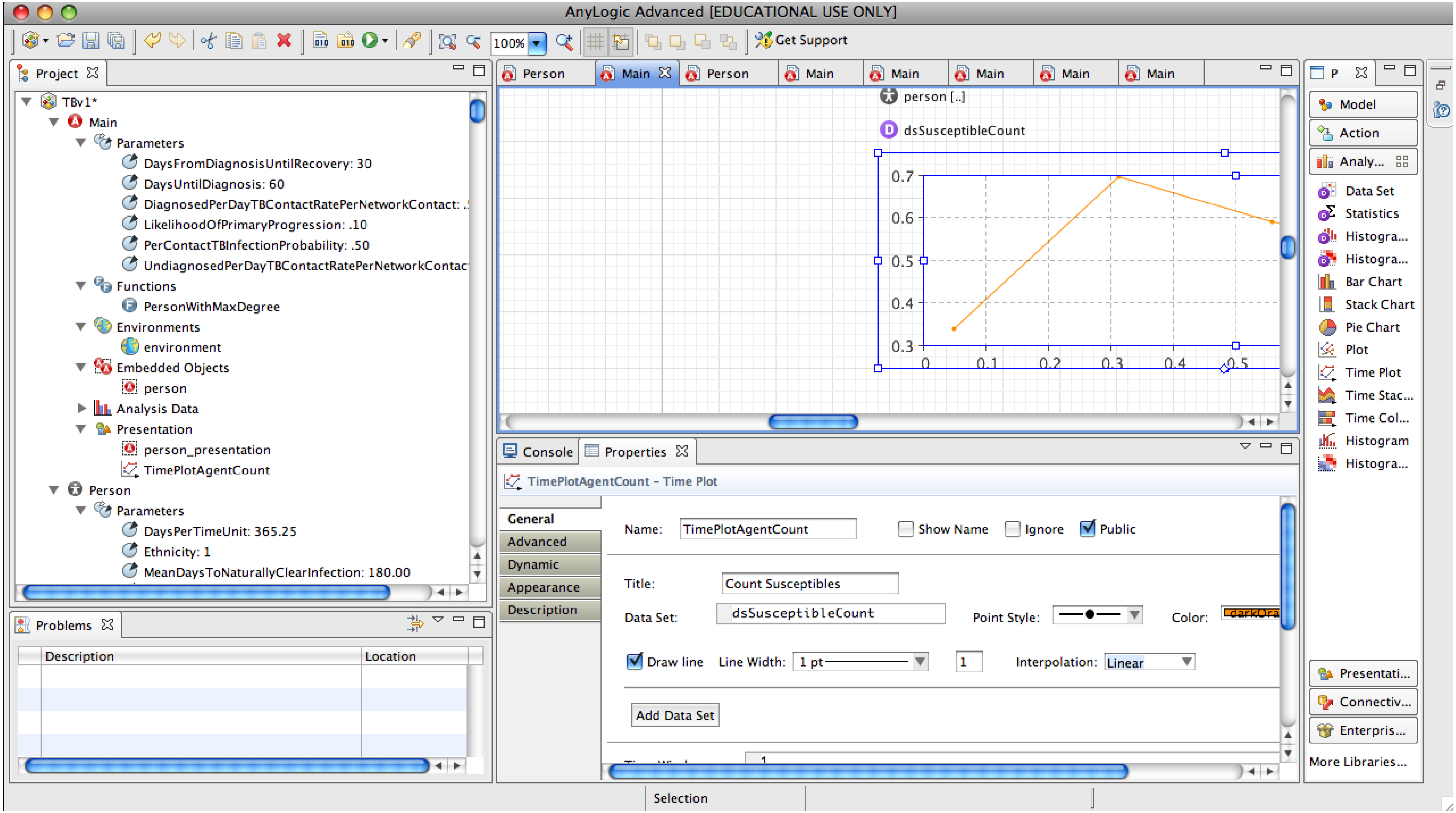
The constructor DataSet() is undefined

Engine.log cannot be resolved

mViewer cannot be resolved

mViewer cannot be resolved

# Chart Use of Datasets



AnyLogic Advanced [EDUCATIONAL USE ONLY]

- TBv1*
  - Main
    - Parameters
      - DaysFromDiagnosisUntilRecovery: 30
      - DaysUntilDiagnosis: 60
      - DiagnosedPerDayTBContactRatePerNetworkContact: .
      - LikelihoodOfPrimaryProgression: .10
      - PerContactTBInfectionProbability: .50
      - UndiagnosedPerDayTBContactRatePerNetworkContac
    - Functions
      - PersonWithMaxDegree
    - Environments
      - environment
    - Embedded Objects
      - person
    - Analysis Data
    - Presentation
      - person_presentation
      - TimePlotAgentCount
  - Person
    - Parameters
      - DaysPerTimeUnit: 365.25
      - Ethnicity: 1
      - MeanDaysToNaturallyClearInfection: 180.00

TimePlotAgentCount - Time Plot

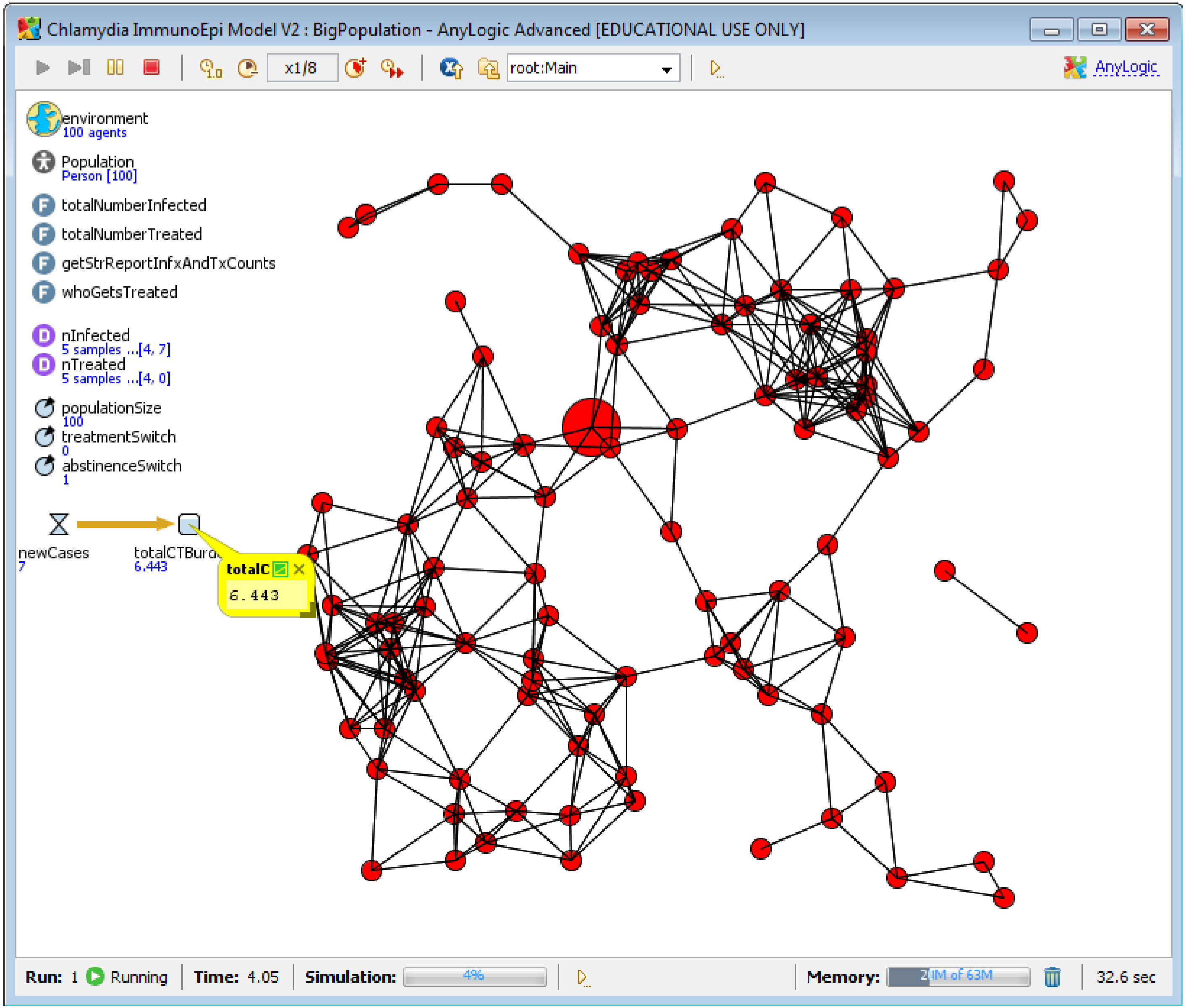
Name: TimePlotAgentCount □ Show Name □ Ignore ☑ Public

Title: Count Susceptibles

Data Set: dsSusceptibleCount Point Style: Color: darkOra

☑ Draw line Line Width: 1 pt 1 Interpolation: Linear

# Ad-hoc Export



Chlamydia ImmunoEpi Model V2 : BigPopulation - AnyLogic Advanced [EDUCATIONAL USE ONLY]
x1/8
root:Main
AnyLogic

environment
100 agents

Population
Person [100]

totalNumberInfected

totalNumberTreated

getStrReportInfxAndTxCounts

whoGetsTreated

nInfected
5 samples ...[4, 7]

nTreated
5 samples ...[4, 0]

populationSize
100

treatmentSwitch
0

abstinenceSwitch
1

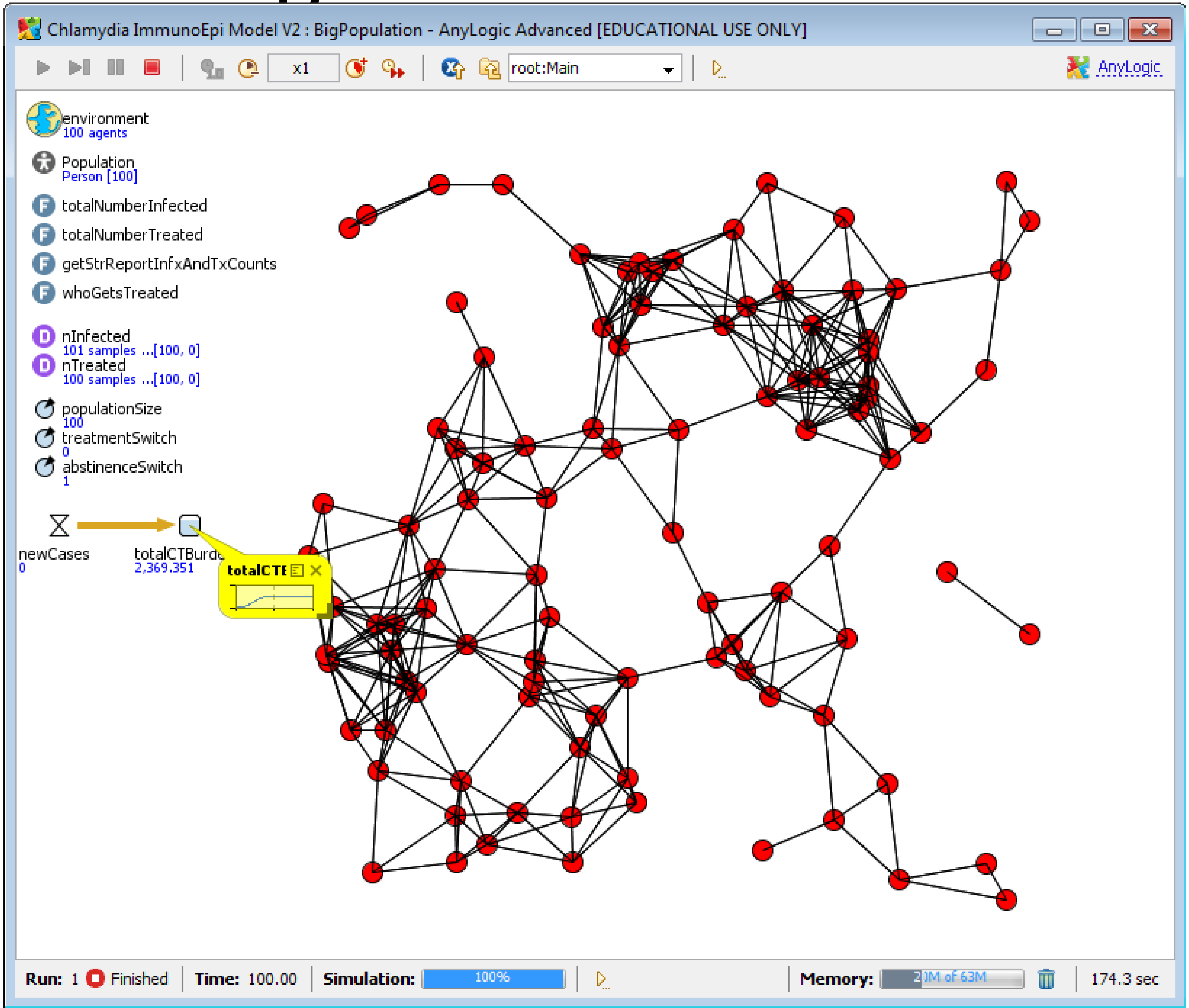
newCases
7

totalCTBurd
6.443

totalC
6.443

Run: 1 Running | Time: 4.05 | Simulation: 4% | Memory: 21M of 63M | 32.6 sec

# Begins as a Small Chart



Chlamydia ImmunoEpi Model V2 : BigPopulation - AnyLogic Advanced [EDUCATIONAL USE ONLY]

environment
100 agents

Population
Person [100]

totalNumberInfected

totalNumberTreated

getStrReportInfxAndTxCounts

whoGetsTreated

nInfected
101 samples ...[100, 0]

nTreated
100 samples ...[100, 0]

populationSize
100

treatmentSwitch
0

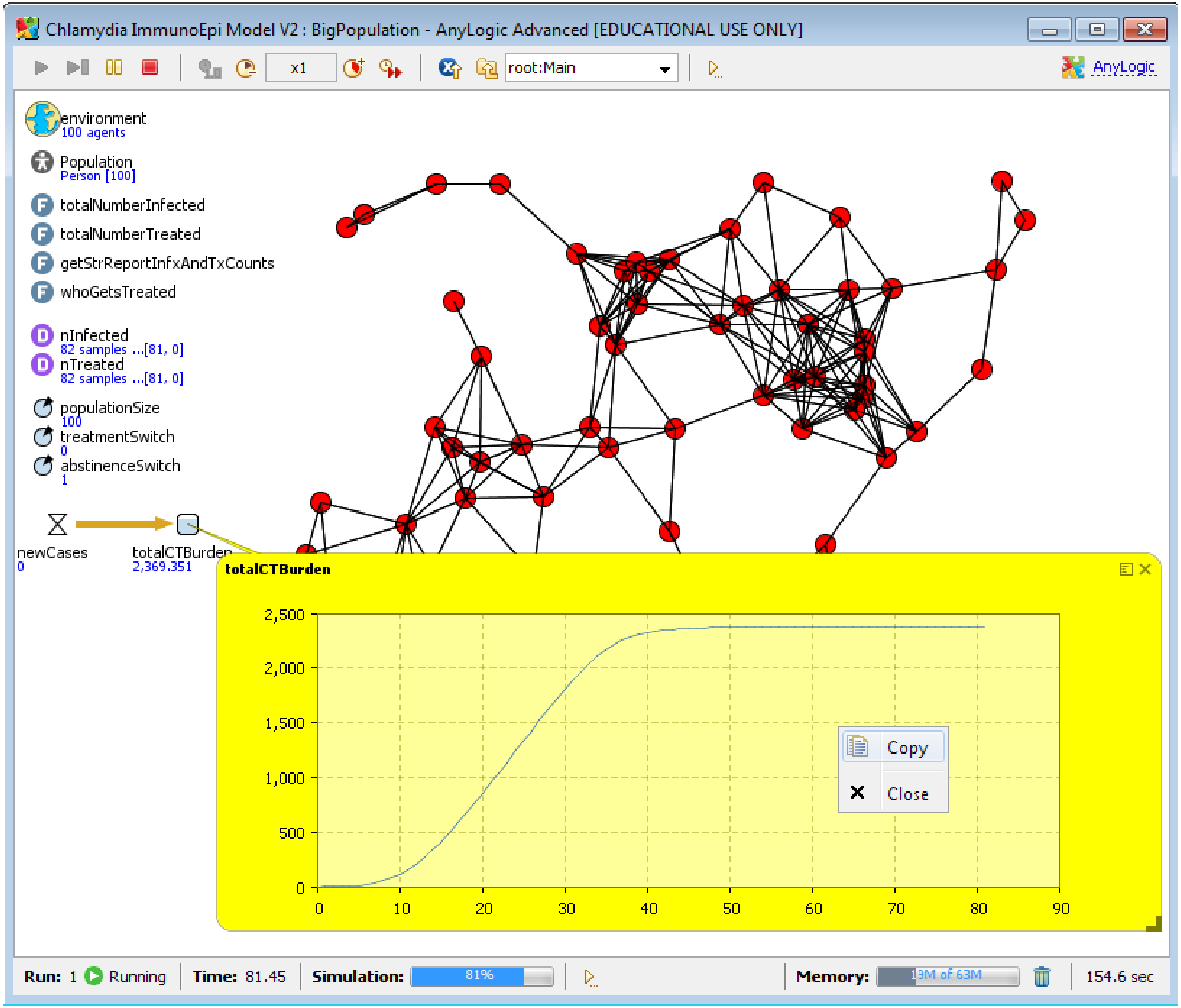
abstinenceSwitch
1

newCases
0

totalCTBurde
2,369.351

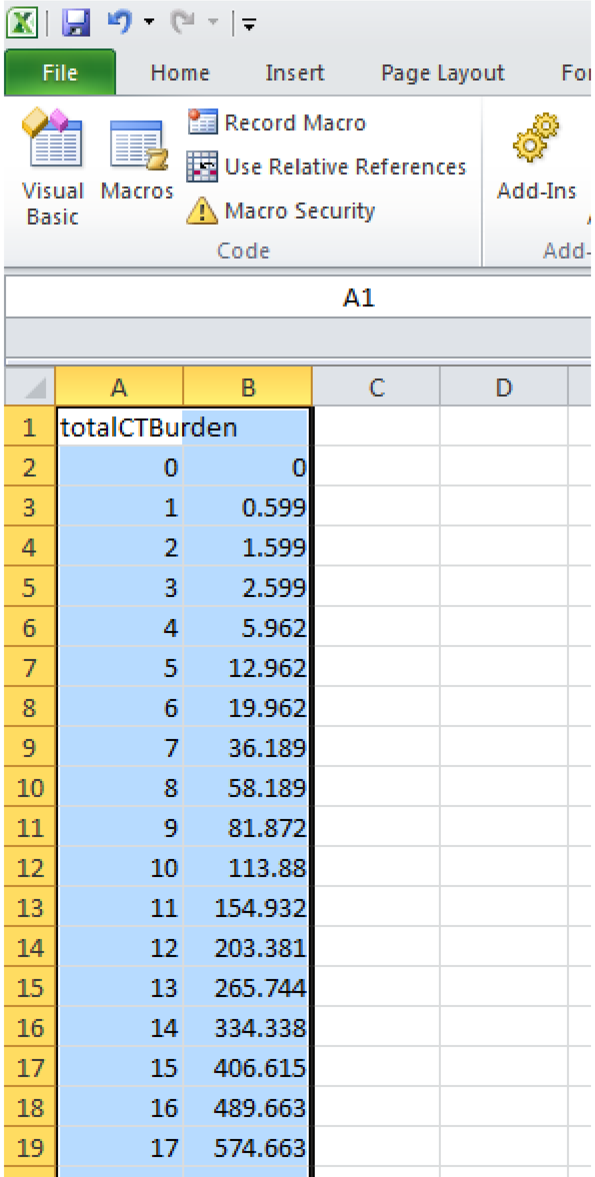
Run: 1 Finished | Time: 100.00 | Simulation: 100% | Memory: 20M of 63M | 174.3 sec

# Copying Data



Chlamydia ImmunoEpi Model V2 : BigPopulation - AnyLogic Advanced [EDUCATIONAL USE ONLY]

# Data Exported from Ad-Hoc Chart



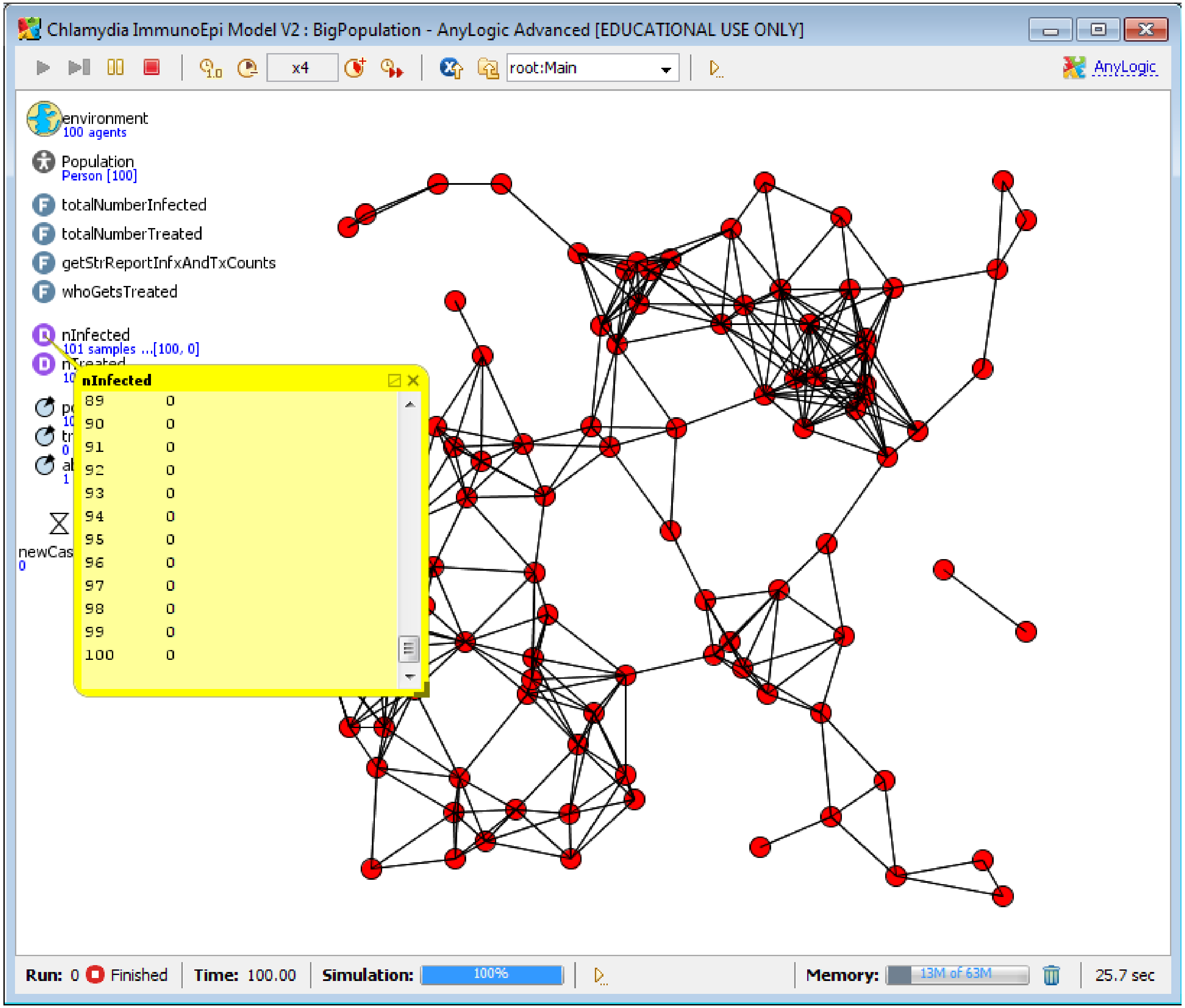
A1

|  | A | B |
|---|---|---|
| 1 | totalCTBurden |  |
| 2 | 0 | 0 |
| 3 | 1 | 0.599 |
| 4 | 2 | 1.599 |
| 5 | 3 | 2.599 |
| 6 | 4 | 5.962 |
| 7 | 5 | 12.962 |
| 8 | 6 | 19.962 |
| 9 | 7 | 36.189 |
| 10 | 8 | 58.189 |
| 11 | 9 | 81.872 |
| 12 | 10 | 113.88 |
| 13 | 11 | 154.932 |
| 14 | 12 | 203.381 |
| 15 | 13 | 265.744 |
| 16 | 14 | 334.338 |
| 17 | 15 | 406.615 |
| 18 | 16 | 489.663 |
| 19 | 17 | 574.663 |

# Techniques for Outputting Data

- Ad-Hoc Exports from variables
- Manual copies from visible datasets
- Capturing images of graphs
- Export to files
- Writing to console
- [AnyLogic Professional] Dataset archiving
- Export to databases

# Manual Output from Datasets



Chlamydia ImmunoEpi Model V2 : BigPopulation - AnyLogic Advanced [EDUCATIONAL USE ONLY]

environment
100 agents

Population
Person [100]

totalNumberInfected

totalNumberTreated

getStrReportInfxAndTxCounts

whoGetsTreated

nInfected
101 samples ...[100, 0]

nTreated

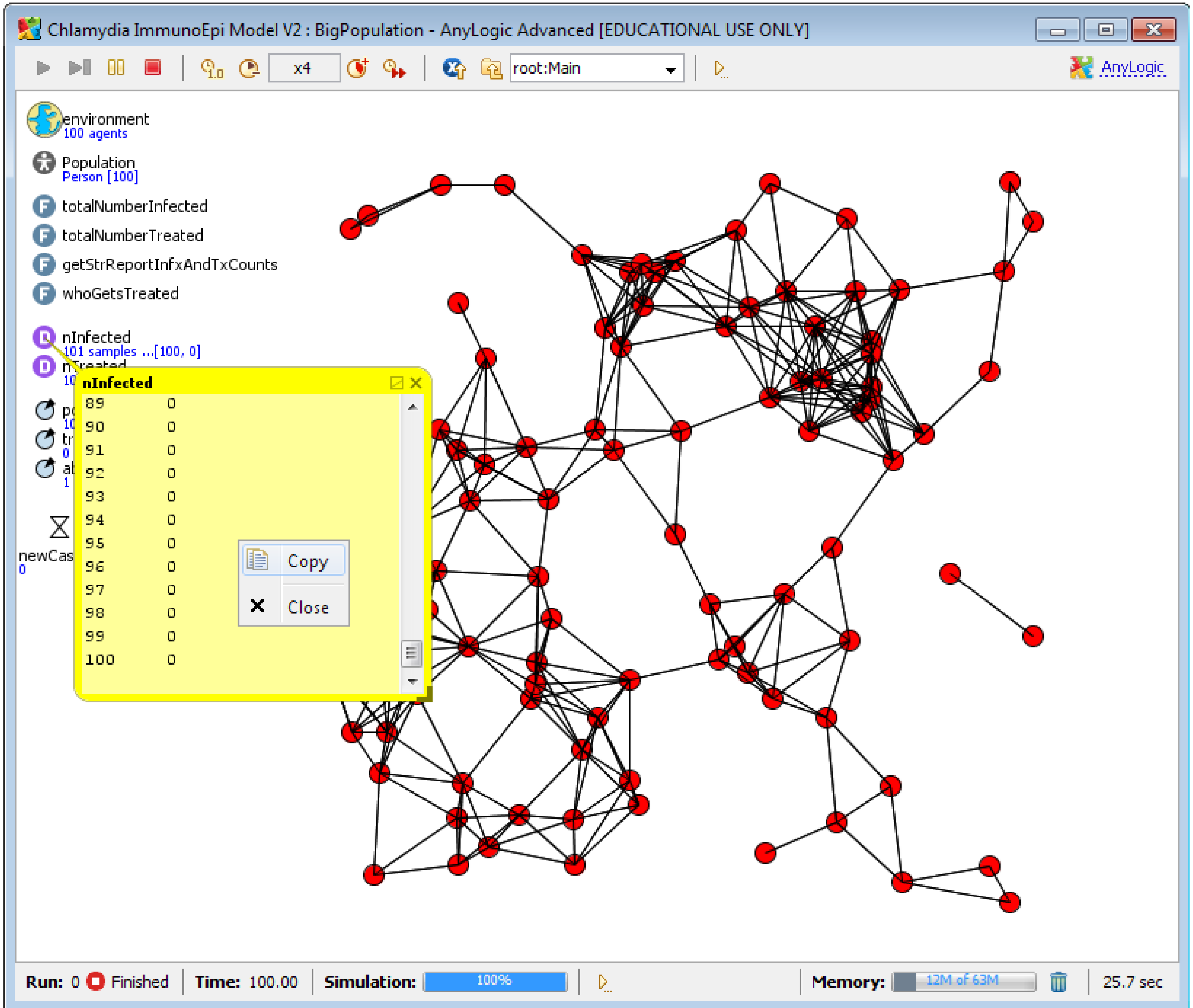
newCas
0

nInfected

```
89      0
90      0
91      0
92      0
93      0
94      0
95      0
96      0
97      0
98      0
99      0
100     0
```

Run: 0 Finished | Time: 100.00 | Simulation: 100% | Memory: 13M of 63M | 25.7 sec

# Right Clicking Gives Context Menu

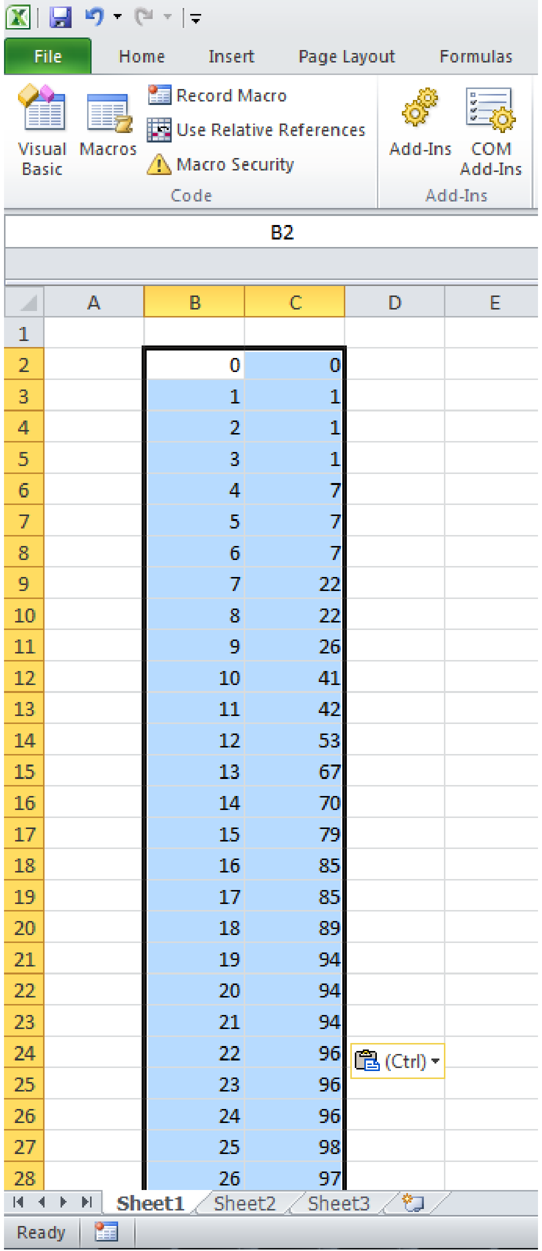


nInfected

```
89    0
90    0
91    0
92    0
93    0
94    0
95    0
96    0
97    0
98    0
99    0
100   0
```

Run: 0 Finished | Time: 100.00 | Simulation: 100% | Memory: 12M of 63M | 25.7 sec

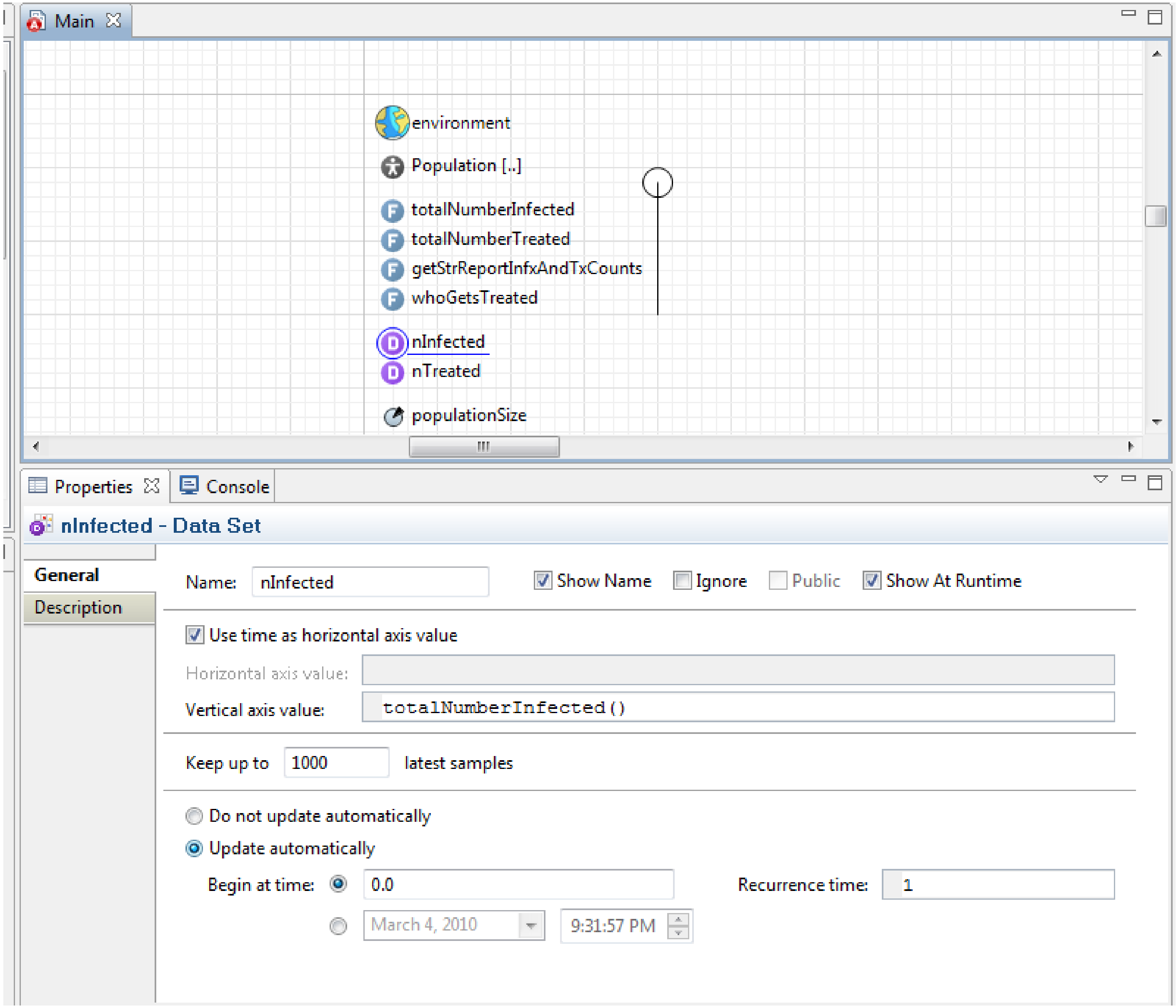
# Copied Data Can be Pasted into Excel



B2

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | | | | | |
| 2 | | 0 | 0 | | |
| 3 | | 1 | 1 | | |
| 4 | | 2 | 1 | | |
| 5 | | 3 | 1 | | |
| 6 | | 4 | 7 | | |
| 7 | | 5 | 7 | | |
| 8 | | 6 | 7 | | |
| 9 | | 7 | 22 | | |
| 10 | | 8 | 22 | | |
| 11 | | 9 | 26 | | |
| 12 | | 10 | 41 | | |
| 13 | | 11 | 42 | | |
| 14 | | 12 | 53 | | |
| 15 | | 13 | 67 | | |
| 16 | | 14 | 70 | | |
| 17 | | 15 | 79 | | |
| 18 | | 16 | 85 | | |
| 19 | | 17 | 85 | | |
| 20 | | 18 | 89 | | |
| 21 | | 19 | 94 | | |
| 22 | | 20 | 94 | | |
| 23 | | 21 | 94 | | |
| 24 | | 22 | 96 | | |
| 25 | | 23 | 96 | | |
| 26 | | 24 | 96 | | |
| 27 | | 25 | 98 | | |
| 28 | | 26 | 97 | | |

# Declaratively Specifying Datasets



nInfected - Data Set

Name: nInfected ☑ Show Name □ Ignore □ Public ☑ Show At Runtime

☑ Use time as horizontal axis value

Horizontal axis value:

Vertical axis value: `totalNumberInfected()`

Keep up to 1000 latest samples

○ Do not update automatically

● Update automatically

Begin at time: ● 0.0 Recurrence time: 1

○ March 4, 2010 9:31:57 PM

# Supported Dataset Types

- Simple
  - holds values only -- no  timestamps
- Timed
  - holds values and timestamps
- Phase
  - holds pairs of values but no timesamps
- Histogram
  - can define bins for data set
  - data set will record # falling in each bin

# Techniques for Outputting Data

- Ad-Hoc Exports from variables
- Manual copies from visible datasets
- Capturing images of graphs
- Export to files
- Writing to console
- [AnyLogic Professional] Dataset archiving
- Export to databases

# Output to Console

- Pros
  - Easy to program
    - ActiveObject.traceln(Stringstr) outputs string to console
    - System.out.println(Stringstr)
  - Readily visible
  - Copy & Paste to another document
- Cons
  - May be mixed with other output (easy to miss other output)
  - Limited length
  - Depends on memory to copy

# Techniques for Outputting Data

- Ad-Hoc Exports from variables
- Manual copies from visible datasets
- Capturing images of graphs
- Export to files
- Writing to console
- [AnyLogic Professional] Dataset archiving
- Export to databases

# Data Output to File

- Pros
  - Simple to perform
  - Relatively easy to import into e.g. Excel, R, etc.
  - Files can be readily archived
- Cons
  - Awkward to draw combine from multiple files
  - Denormalization: Requires either
    - Duplication of scenario-wide information (e.g. parameter values) on each row
    - Separate header section & later section

# Example code to Export Dataset to File

```
FileOutputStream fos = new
   FileOutputStream(“Filename”);
PrintStream p = new PrintStream(fos);
p.println(datasetName.toString());  // outputs
   tab delimited values
```

# Techniques for Outputting Data

- Ad-Hoc Exports from variables
- Manual copies from visible datasets
- Capturing images of graphs
- Export to files
- Writing to console
- [AnyLogic Professional] Dataset archiving
- Export to databases

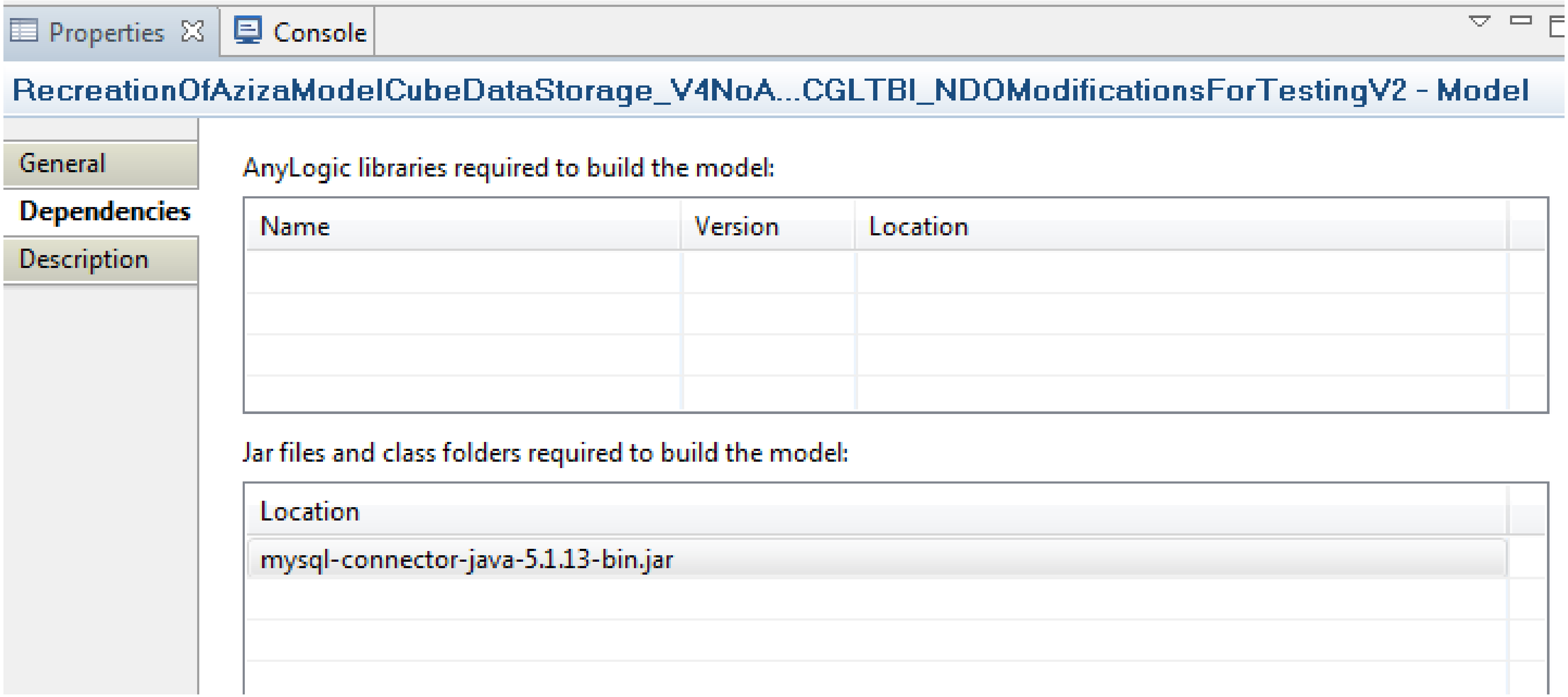
# Output to Databases: Tradeoffs

- Pros
  - More flexible than string output to file
  - Can query from diverse tools (e.g. excel, R, SPSS, SAS, etc.)
  - Can easily clean up
  - For larger databases
    - Transactional (either writes entirely or not at all)
    - Can query from remote machines
- Cons
  - More programming
  - Need to set up a database

# Output to Databases: Steps

- One Time:
  - Install database on computer
  - Add reference to database libraries
- Each time during simulation
  - Open database connection at start of model
  - Optionally, “insert” model version & parameter information into the database
  - Periodically during simulation
    - “insert” values into databases
  - At end of model execution, close database connection

# Database Dependencies (MySQL database)



Properties
Console
RecreationOfAzizaModelCubeDataStorage_V4NoA...CGLTBI_NDOModificationsForTestingV2 - Model
General
Dependencies
Description

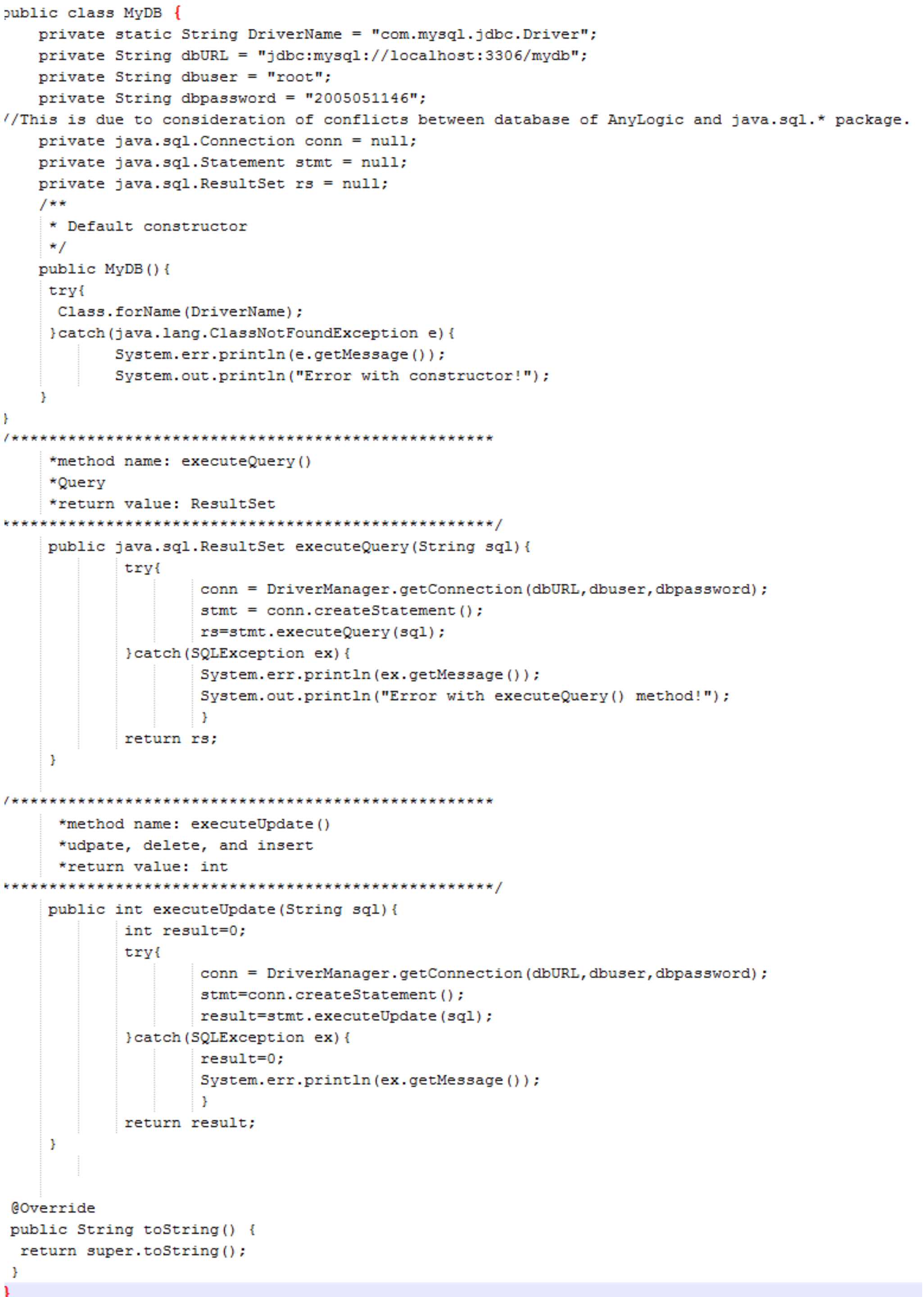
AnyLogic libraries required to build the model:

| Name | Version | Location |
|---|---|---|

Jar files and class folders required to build the model:

| Location |
|---|
| mysql-connector-java-5.1.13-bin.jar |

# Example Simple Database Class

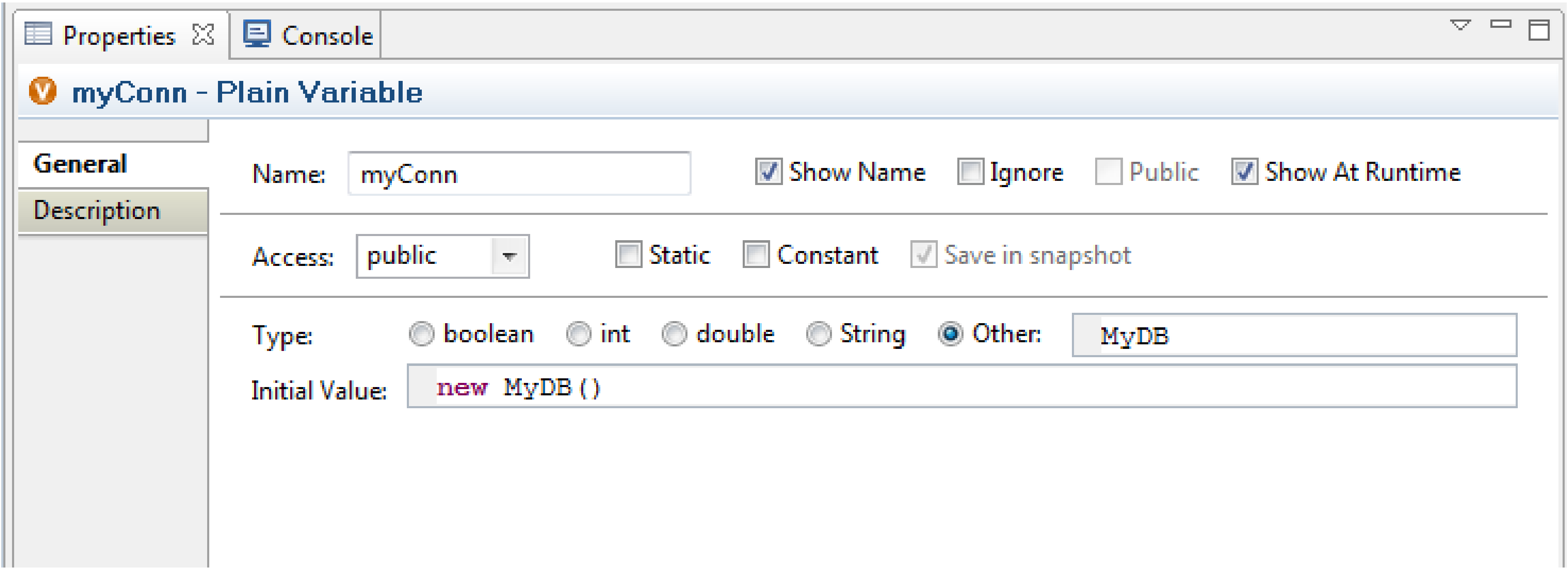


```
public class MyDB {
    private static String DriverName = "com.mysql.jdbc.Driver";
    private String dbURL = "jdbc:mysql://localhost:3306/mydb";
    private String dbuser = "root";
    private String dbpassword = "2005051146";
//This is due to consideration of conflicts between database of AnyLogic and java.sql.* package.
    private java.sql.Connection conn = null;
    private java.sql.Statement stmt = null;
    private java.sql.ResultSet rs = null;
    /**
     * Default constructor
     */
    public MyDB(){
     try{
      Class.forName(DriverName);
     }catch(java.lang.ClassNotFoundException e){
            System.err.println(e.getMessage());
            System.out.println("Error with constructor!");
    }
}
/***************************************************
     *method name: executeQuery()
     *Query
     *return value: ResultSet
***************************************************/
     public java.sql.ResultSet executeQuery(String sql){
             try{
                     conn = DriverManager.getConnection(dbURL,dbuser,dbpassword);
                     stmt = conn.createStatement();
                     rs=stmt.executeQuery(sql);
             }catch(SQLException ex){
                     System.err.println(ex.getMessage());
                     System.out.println("Error with executeQuery() method!");
                     }
             return rs;
     }

/***************************************************
      *method name: executeUpdate()
      *udpate, delete, and insert
      *return value: int
***************************************************/
     public int executeUpdate(String sql){
             int result=0;
             try{
                     conn = DriverManager.getConnection(dbURL,dbuser,dbpassword);
                     stmt=conn.createStatement();
                     result=stmt.executeUpdate(sql);
             }catch(SQLException ex){
                     result=0;
                     System.err.println(ex.getMessage());
                     }
             return result;
     }

 @Override
 public String toString() {
  return super.toString();
 }
}
```

# Setup for Database Class



Properties | Console

myConn - Plain Variable

General

Description

Name: myConn ☑ Show Name □ Ignore □ Public ☑ Show At Runtime

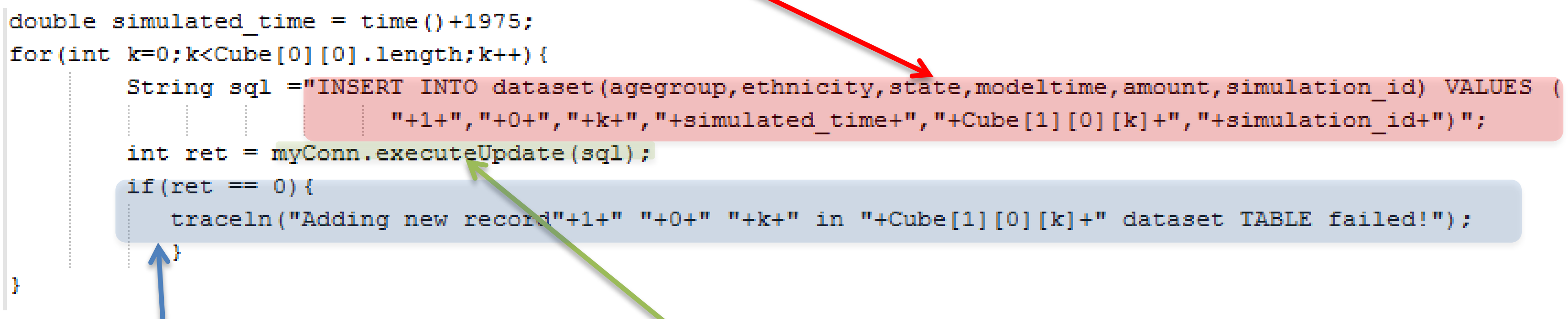
Access: public □ Static □ Constant ☑ Save in snapshot

Type: ○ boolean ○ int ○ double ○ String ● Other: MyDB

Initial Value: `new MyDB()`

# Example Database Output Code

A database query language (SQL) statement



```
double simulated_time = time()+1975;
for(int k=0;k<Cube[0][0].length;k++){
        String sql ="INSERT INTO dataset(agegroup,ethnicity,state,modeltime,amount,simulation_id) VALUES (
                    "+1+","+0+","+k+","+simulated_time+","+Cube[1][0][k]+","+simulation_id+")";
        int ret = myConn.executeUpdate(sql);
        if(ret == 0){
           traceln("Adding new record"+1+" "+0+" "+k+" in "+Cube[1][0][k]+" dataset TABLE failed!");
        }
}
```

Requesting that the database class execute the SQL statemen

Checking to make sure that the insert worked properly