Dynamic Populations and Networks

Nathaniel Osgood
MIT 15.879

March 16, 2012

Critical Role of Network & Population Dynamics

- We have introduced the basic mechanisms for
  - Creating populations of pre-specified sized
  - Creating network from a pre-specified set of network categories
- However,
  - Open populations (e.g. with immigration, death, birth) are the norm
  - Research suggests that many types of networks dynamics (serial partnerships, differing contact durations) are important to infection dynamics

AnyLogic's Support of Network & Population Dynamics

- Fortunately, AnyLogic provides strong support for
  - Adding & removing population members
  - Adding & removing connections
- However, this support does not yet have direct graphical interface support or specification
  - using this support does require that you call "methods" to accomplish this

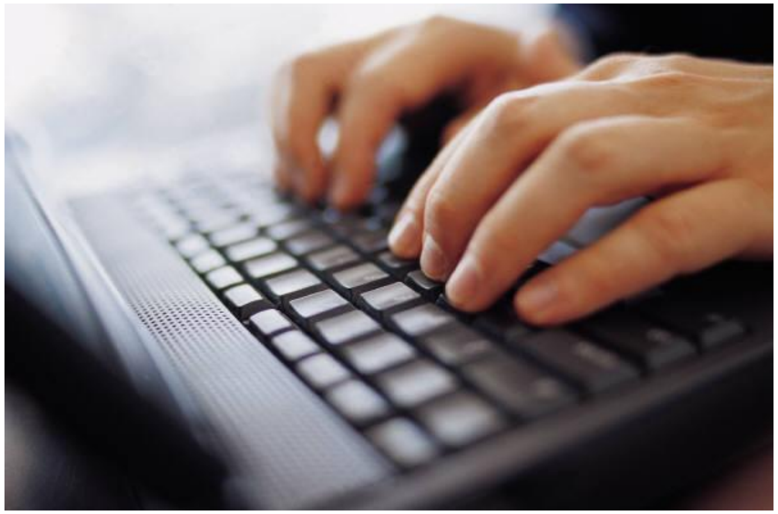
AnyLogic Support for Changing Populations

- Adding to population
  - add_*populationname(parameters)*
    - *Allow explicit specification of agent parameter values*
  - add_*populationname()*
    - *Uses* population *specification of agent parameter values*
- Deleting from population
  - remove_populationname(agentToBeRemoved)

AnyLogic methods for Adding & Deleting Connections

- *agentA*.connectTo(*agentB*)
  - Connects *agentA* to *agentB*
  - NB: Connections are assumed to be undirected and symmetric (i.e. if *agentA* is considered to be connected to *agentB*, then *agentB* is considered to be connected to *agentA*)
- *agentA*.disconnectFrom(*agentB*)
  - Disconnects *agentA* and *agentB* from each other
- For more details and additional methods, see the slides for the *Networks* lecture

Hands on Model Use Ahead

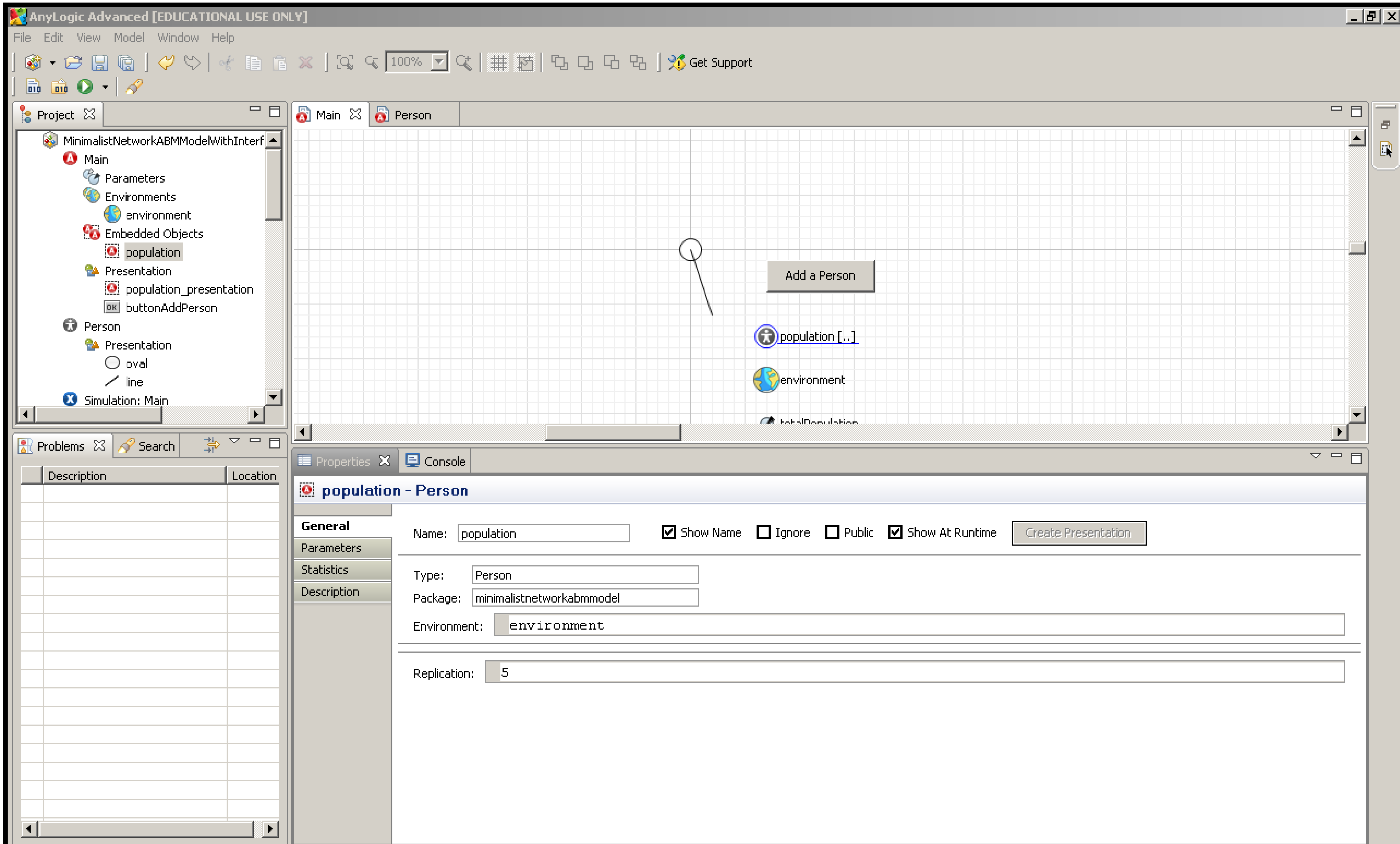


Load Previously Built Model:

MinimalistNetworkABMModel

Suggest Saving as “**MinimalistNetworkABMModelWithInterfaceDrivenPopulationDynamics**”

Set Small Population Size (5)



population - Person

Name: population

[✓] Show Name [ ] Ignore [ ] Public [✓] Show At Runtime

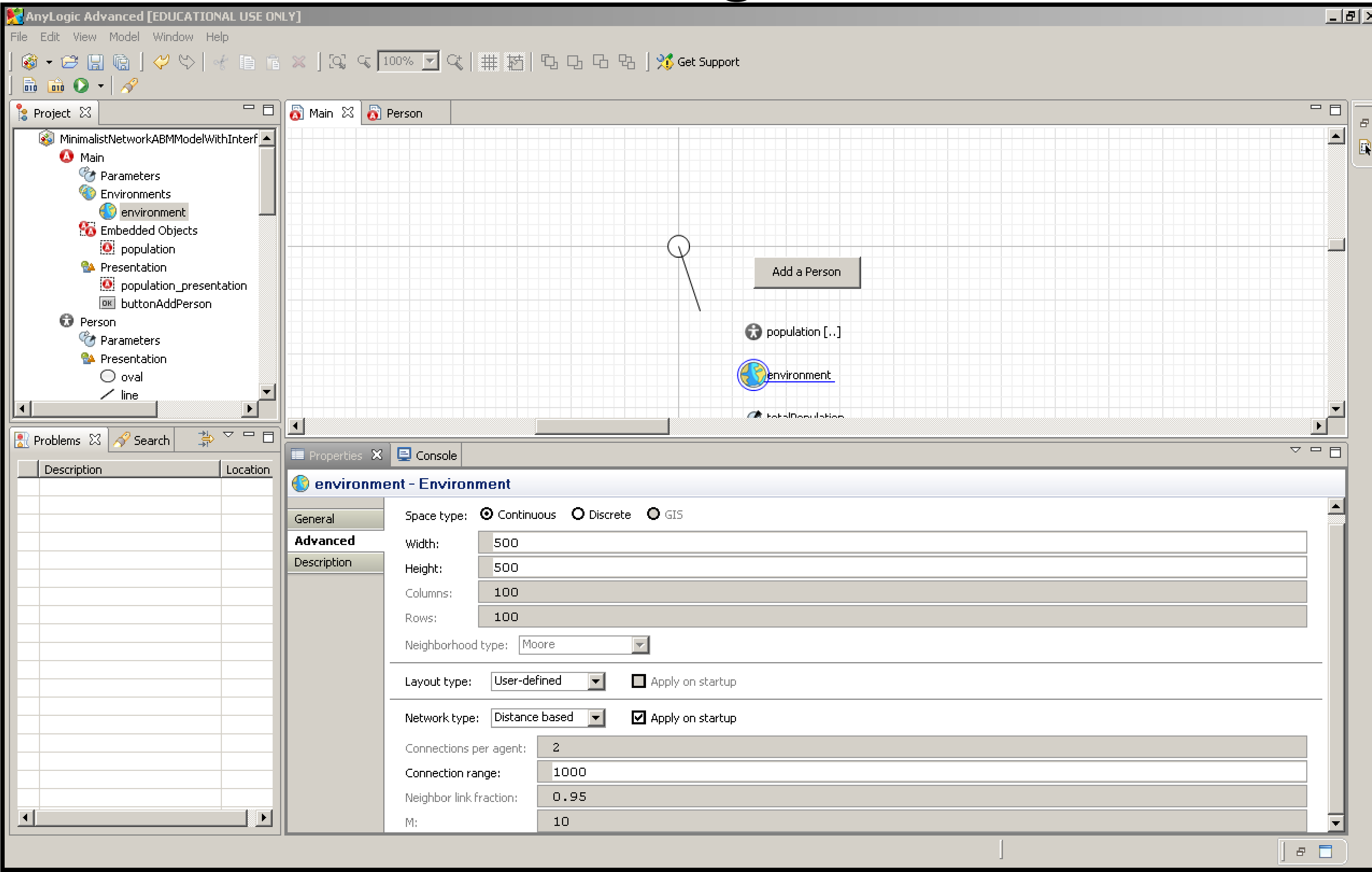
Type: Person

Package: minimalistnetworkabmmodel

Environment: `environment`

Replication: 5

Set Distance Based Network with High Connection Range Threshold



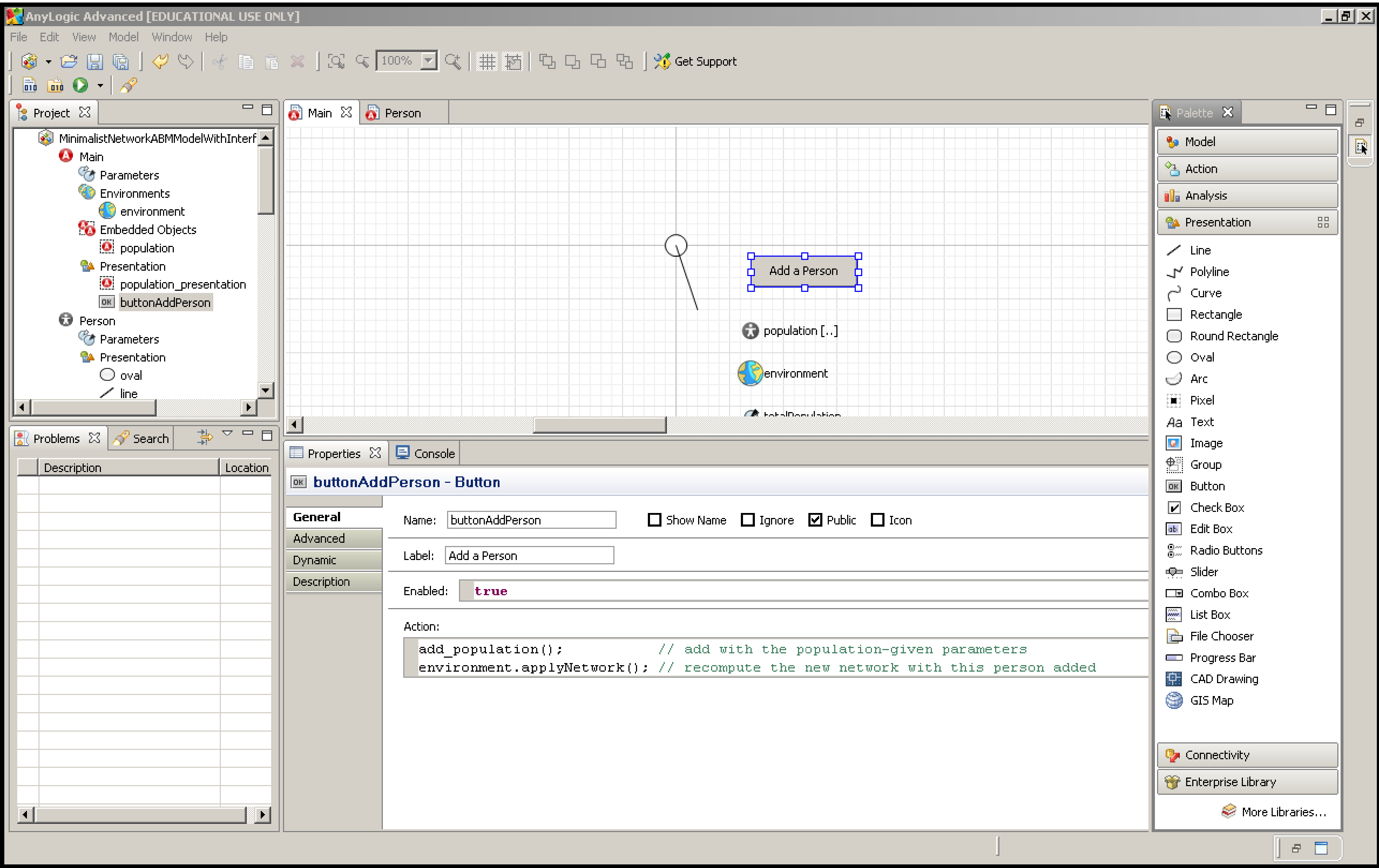
environment - Environment

General | Advanced | Description

Space type: (●) Continuous ( ) Discrete ( ) GIS

Width:	500
Height:	500
Columns:	100
Rows:	100
Neighborhood type:	Moore
Layout type:	User-defined ☐ Apply on startup
Network type:	Distance based ☑ Apply on startup
Connections per agent:	2
Connection range:	1000
Neighbor link fraction:	0.95
M:	10

To Main: Add Button to Request Adding Population Member



buttonAddPerson - Button

General | Advanced | Dynamic | Description

Name: buttonAddPerson   ☐ Show Name ☐ Ignore ☑ Public ☐ Icon

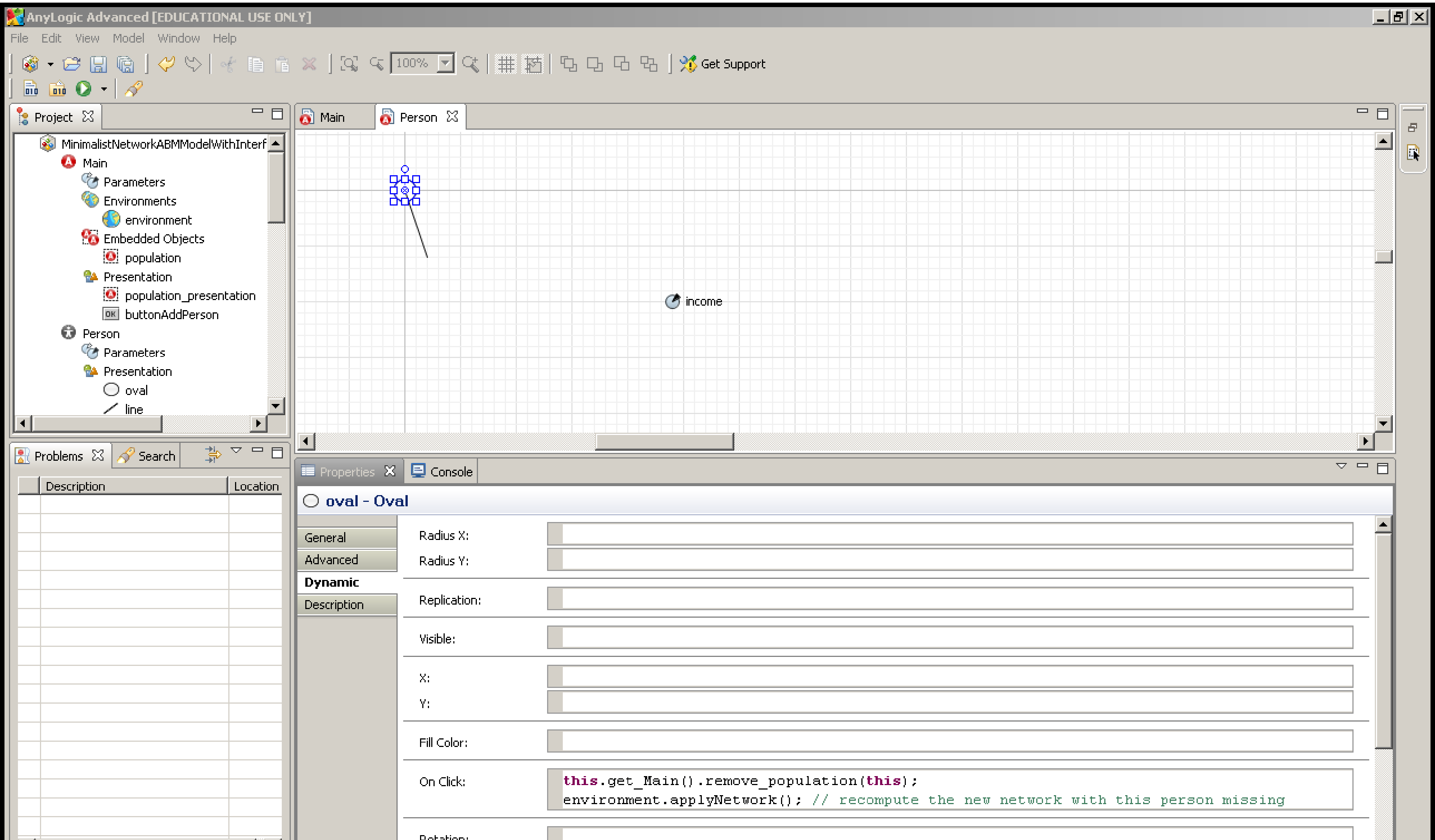
Label: Add a Person

Enabled: true

Action:

```
add_population();           // add with the population-given parameters
environment.applyNetwork(); // recompute the new network with this person added
```

To Person's "Oval", Add a "Handler" to Delete a Person if their Node is Clicked



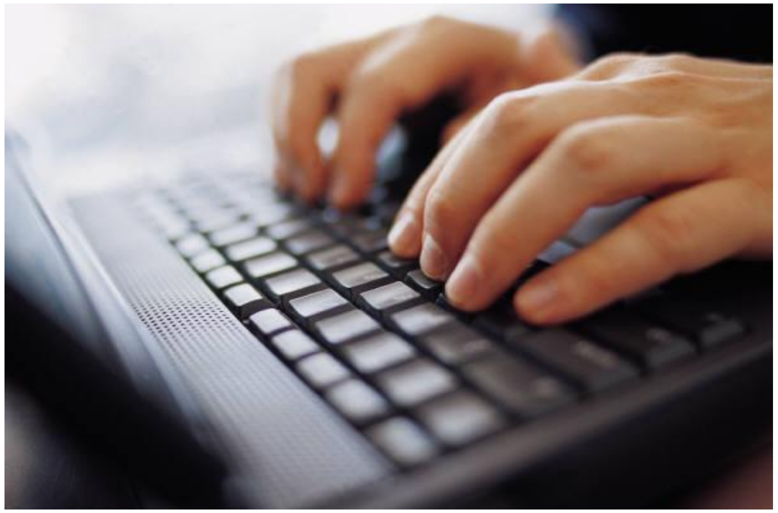
Radius X:	
Radius Y:	
Replication:	
Visible:	
X:	
Y:	
Fill Color:	

On Click:

```
this.get_Main().remove_population(this);
environment.applyNetwork(); // recompute the new network with this person missing
```

Rotation:

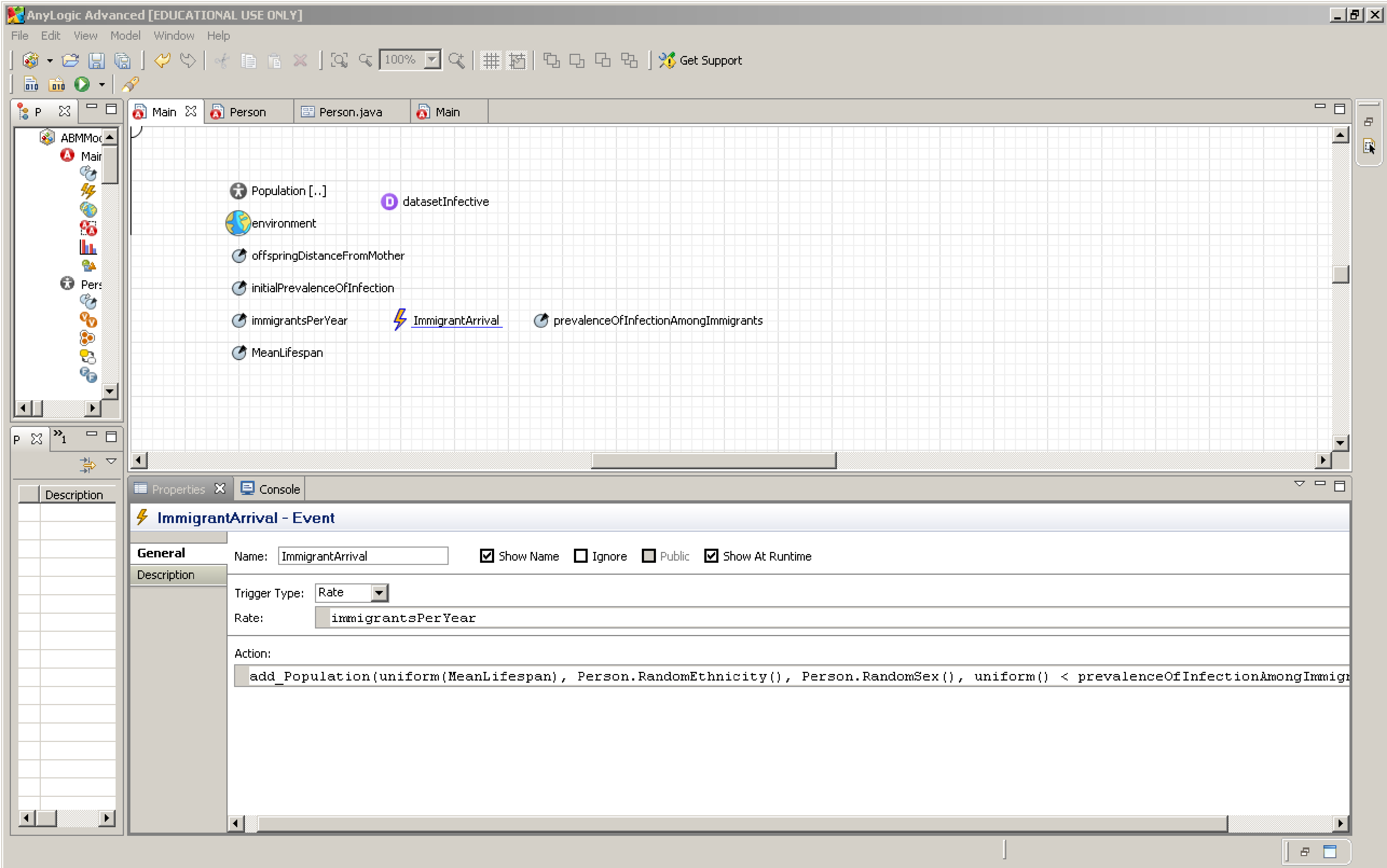
Hands on Model Use Ahead



Load Provided Shared Model:

ABMModelWithBirthDeath

Adding an Immigrant to the Model Population



ImmigrantArrival - Event

General | Description

Name: ImmigrantArrival ☑ Show Name ☐ Ignore ☐ Public ☑ Show At Runtime

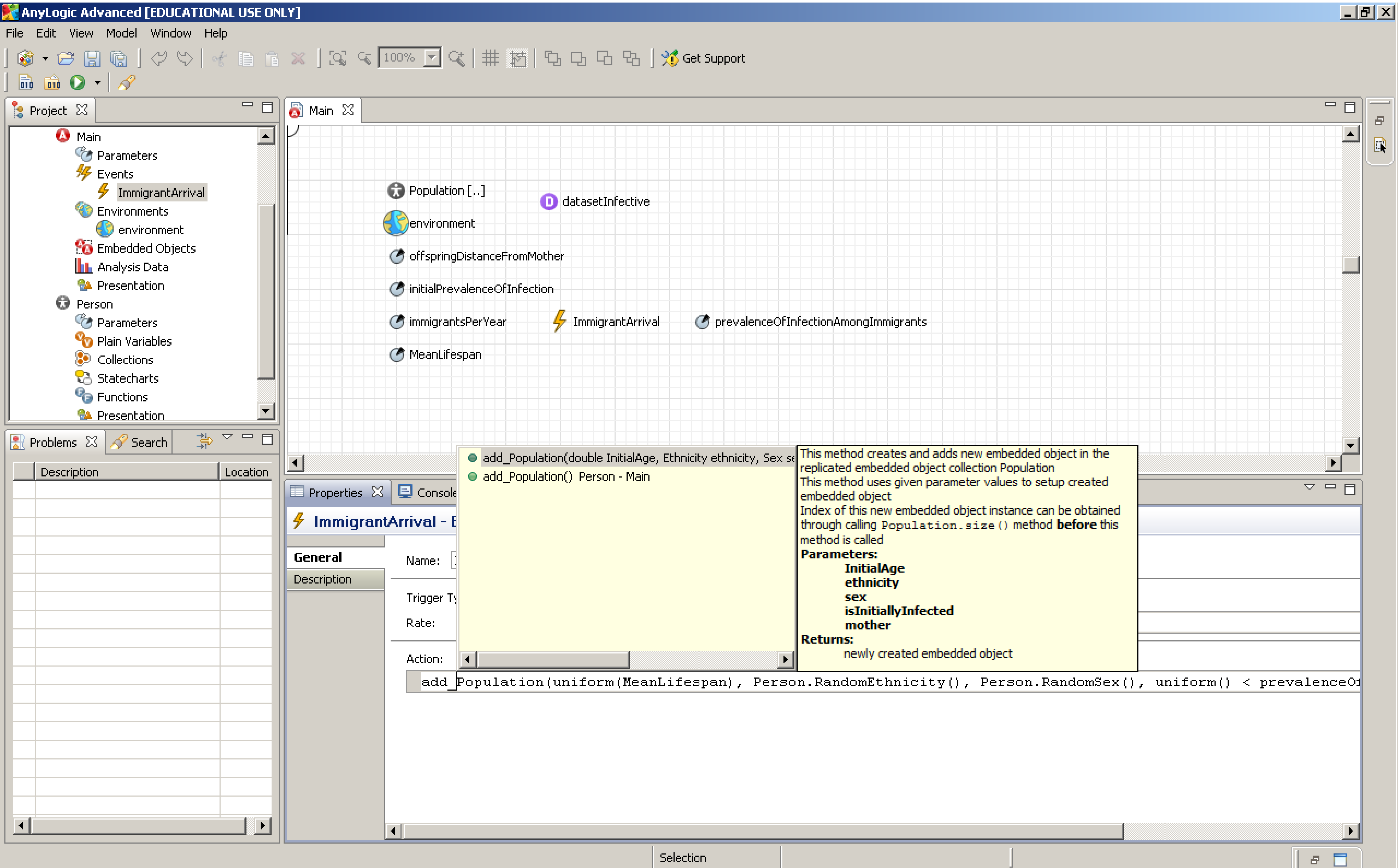
Trigger Type: Rate

Rate: immigrantsPerYear

Action:

```
add_Population(uniform(MeanLifespan), Person.RandomEthnicity(), Person.RandomSex(), uniform() < prevalenceOfInfectionAmongImmigr
```

Add Population Options – Note Customization to Context

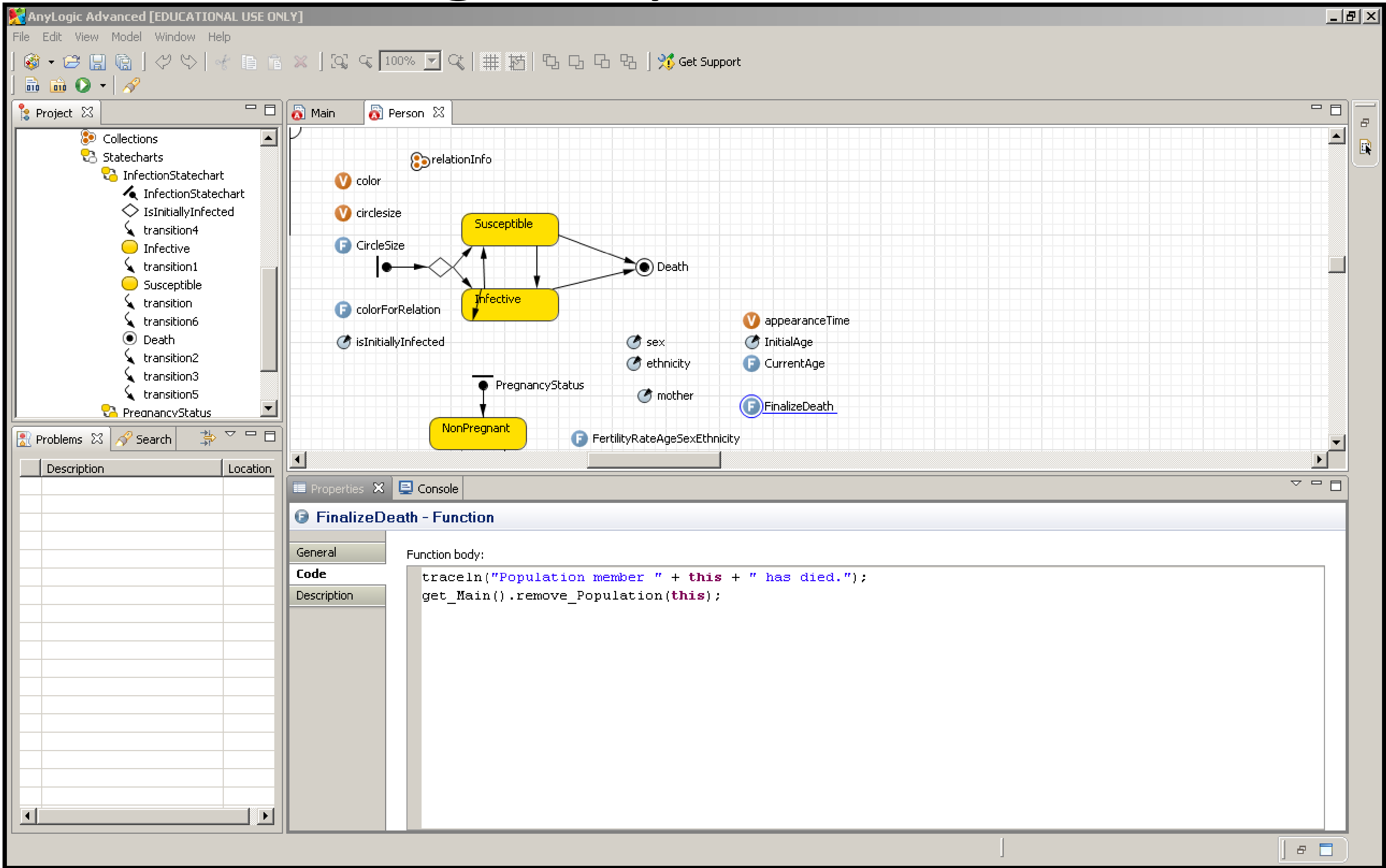


● add_Population(double InitialAge, Ethnicity ethnicity, Sex se
● add_Population() Person - Main

This method creates and adds new embedded object in the replicated embedded object collection Population
This method uses given parameter values to setup created embedded object
Index of this new embedded object instance can be obtained through calling `Population.size()` method **before** this method is called
Parameters:
InitialAge
ethnicity
sex
isInitiallyInfected
mother
Returns:
newly created embedded object

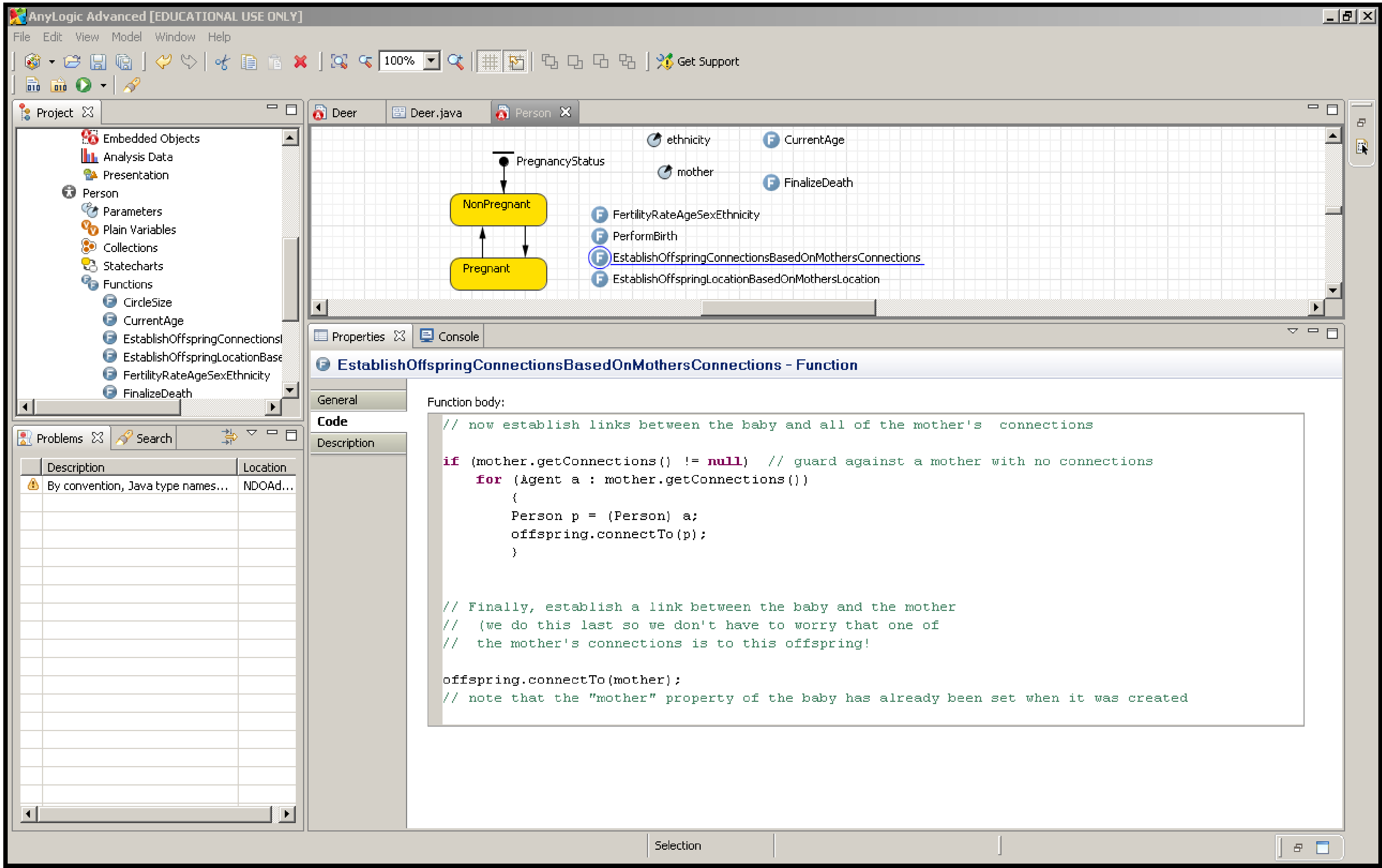
```
add_Population(uniform(MeanLifespan), Person.RandomEthnicity(), Person.RandomSex(), uniform() < prevalenceO
```

Removing a Population Member



```
traceln("Population member " + this + " has died.");
get_Main().remove_Population(this);
```

Establishing Baby's Connection Looping over Connections

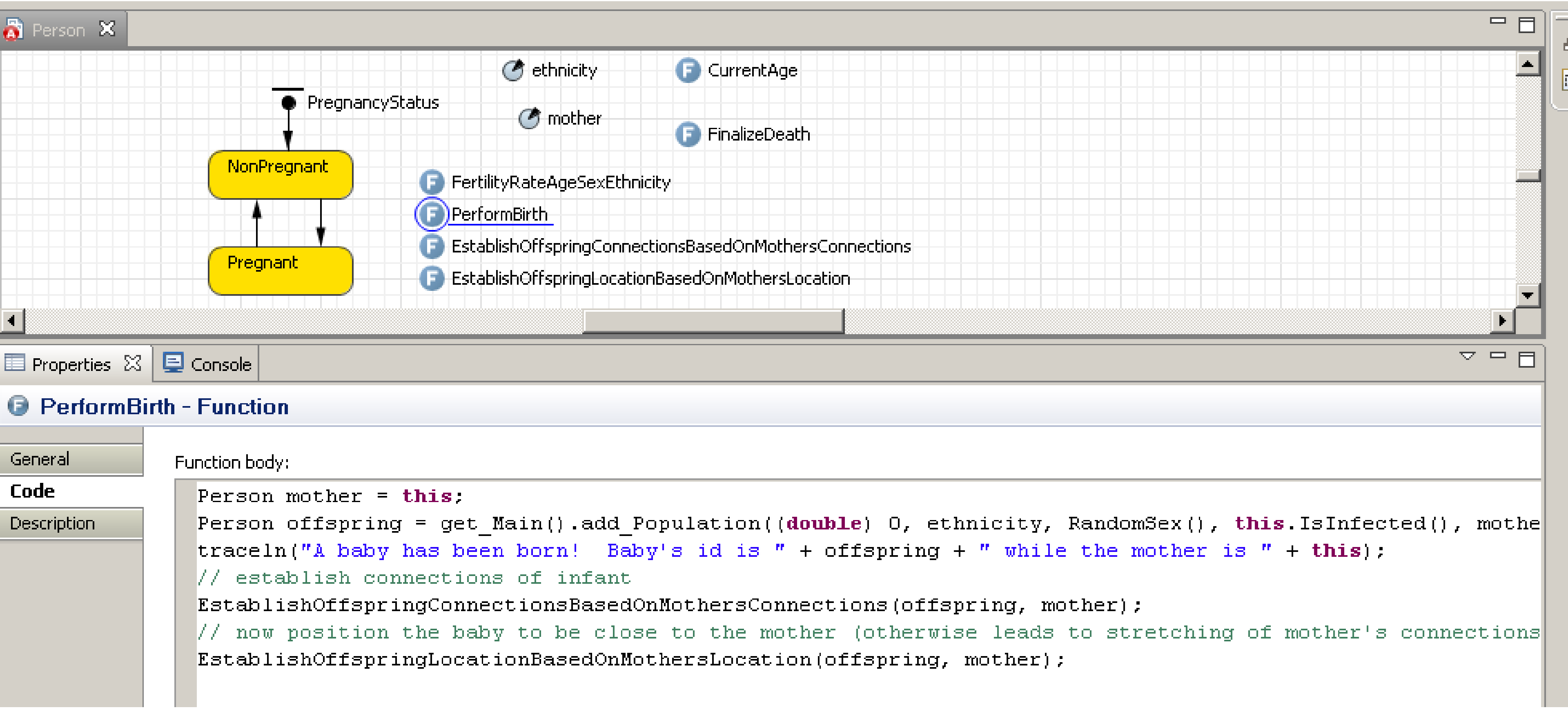


EstablishOffspringConnectionsBasedOnMothersConnections - Function

Function body:

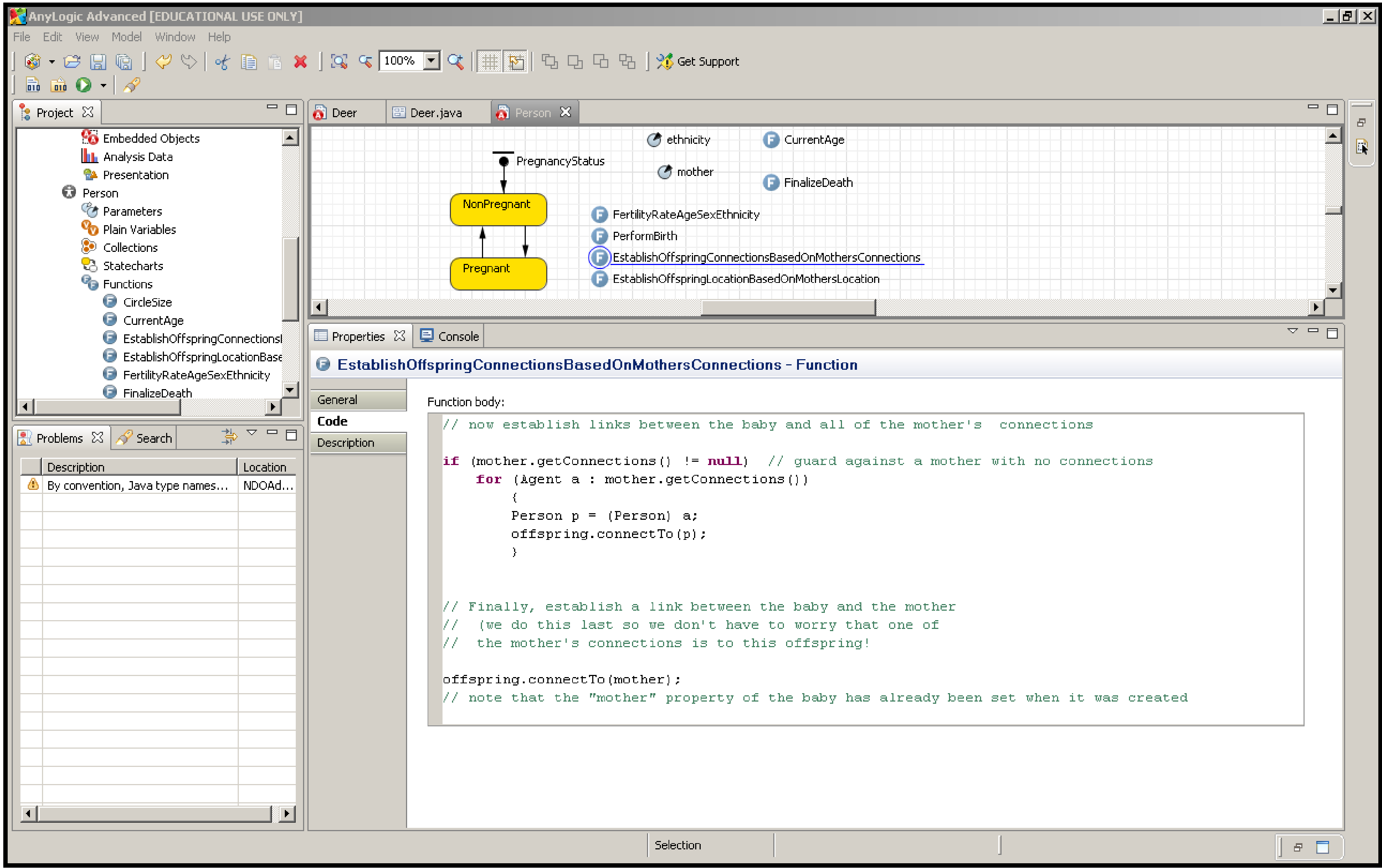
```
// now establish links between the baby and all of the mother's  connections

if (mother.getConnections() != null)  // guard against a mother with no connections
    for (Agent a : mother.getConnections())
        {
        Person p = (Person) a;
        offspring.connectTo(p);
        }


// Finally, establish a link between the baby and the mother
//  (we do this last so we don't have to worry that one of
//  the mother's connections is to this offspring!

offspring.connectTo(mother);
// note that the "mother" property of the baby has already been set when it was created
```

Code to Perform Birth



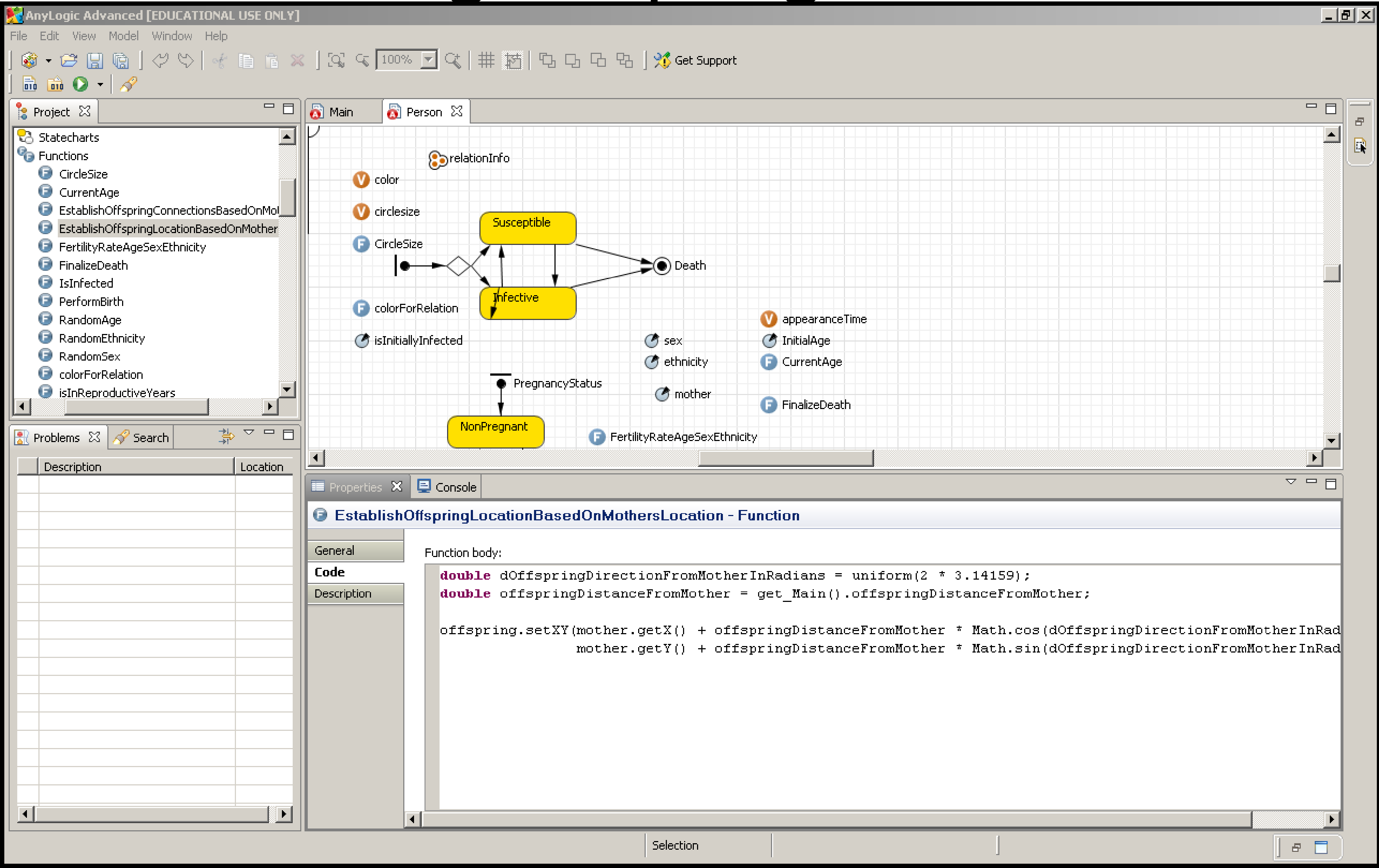
```
Person mother = this;
Person offspring = get_Main().add_Population((double) 0, ethnicity, RandomSex(), this.IsInfected(), mothe
traceln("A baby has been born!  Baby's id is " + offspring + " while the mother is " + this);
// establish connections of infant
EstablishOffspringConnectionsBasedOnMothersConnections(offspring, mother);
// now position the baby to be close to the mother (otherwise leads to stretching of mother's connections
EstablishOffspringLocationBasedOnMothersLocation(offspring, mother);
```

Establishing Baby's Connection Looping over Connections


AnyLogic Advanced [EDUCATIONAL USE ONLY]
File Edit View Model Window Help
100%
Get Support
Project
Embedded Objects
Analysis Data
Presentation
Person
Parameters
Plain Variables
Collections
Statecharts
Functions
CircleSize
CurrentAge
EstablishOffspringConnectionsI
EstablishOffspringLocationBase
FertilityRateAgeSexEthnicity
FinalizeDeath
Deer
Deer.java
Person
ethnicity
CurrentAge
PregnancyStatus
mother
FinalizeDeath
NonPregnant
Pregnant
FertilityRateAgeSexEthnicity
PerformBirth
EstablishOffspringConnectionsBasedOnMothersConnections
EstablishOffspringLocationBasedOnMothersLocation
Properties
Console
EstablishOffspringConnectionsBasedOnMothersConnections - Function
General
Code
Description
Function body:
Problems
Search
Description
Location
By convention, Java type names…
NDOAd…
Selection

```
// now establish links between the baby and all of the mother's  connections

if (mother.getConnections() != null)  // guard against a mother with no connections
    for (Agent a : mother.getConnections())
        {
        Person p = (Person) a;
        offspring.connectTo(p);
        }


// Finally, establish a link between the baby and the mother
//  (we do this last so we don't have to worry that one of
//  the mother's connections is to this offspring!

offspring.connectTo(mother);
// note that the "mother" property of the baby has already been set when it was created
```

Setting Offspring Location



EstablishOffspringLocationBasedOnMothersLocation - Function

Function body:

```
double dOffspringDirectionFromMotherInRadians = uniform(2 * 3.14159);
double offspringDistanceFromMother = get_Main().offspringDistanceFromMother;

offspring.setXY(mother.getX() + offspringDistanceFromMother * Math.cos(dOffspringDirectionFromMotherInRad
                mother.getY() + offspringDistanceFromMother * Math.sin(dOffspringDirectionFromMotherInRad
```