

Discrete Intra-Agent Dynamics: Statecharts

Nathaniel Osgood

Using Modeling to Prepare for Changing
Healthcare Needs

Duke-NUS

April 16, 2014

Discrete Agent Dynamics

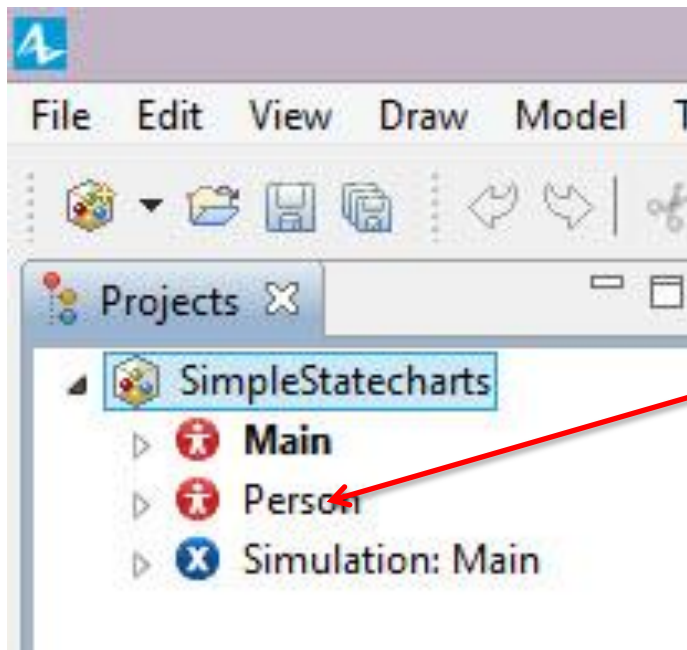
- Frequently we can represent agent behaviour using as transitioning among a set of mutually exclusive and collectively exhaustive states in a “state chart”
- For a given simple statechart, the agent is in exactly one state at a time
- Fixed transitions between states define possible evolution
- The transitions between states occur instantaneously, based on some condition

Hands on Model Use Ahead



Load Previous Built [& Provided] Model:
HeterogeneousAgents
Save as **SimpleStatecharts**

Open Up Canvas for “Person” (In case it is not already open)



Double Click Here

Add a Statechart Entry Point

The screenshot displays the AnyLogic Professional interface. The main workspace shows a statechart diagram with a start node (a circle with a dot) and two entry points labeled 'income' and 'sex'. A red oval highlights the 'statechart' entry point, with a red arrow pointing to it from the text 'The associated text is the name of the statechart!'. The left sidebar shows the 'Projects' tree with 'SimpleStatecharts*' expanded, containing 'Main' and 'Person'. The right sidebar shows the 'Palette' with various components like 'Agent', 'Parameter', 'Event', etc. The bottom panel shows the 'Person - Agent Type' properties. The 'Name' field is set to 'Person'. The 'Parameters preview' section shows 'income:' and 'sex:' fields. The 'Agent actions' section is also visible. The status bar at the bottom indicates '1 error(s)' and 'Time units: minutes'.

AnyLogic Professional

File Edit View Draw Model Tools Help

Projects

- SimpleStatecharts*
 - Main
 - Person
 - Presentation
 - oval
 - Statecharts
 - statechart
 - Parameters
 - Links to agents
 - Simulation: Main

Person Main

income

sex

statechart

The associated text is the name of the statechart!

Person - Agent Type

Name: Person ☐ Ignore

Parameters preview

income:

☒ No delimiter ☐ Add separator ☐ Begin section

sex:

Agent actions

On startup

SimpleStatecharts

1 error(s)

Time units: minutes

Add a Initial State

AnyLogic Professional

File Edit View Draw Model Tools Help

100%

Get Support...

Projects

- SimpleStatecharts*
- Main
- Person
 - Presentation
 - oval
 - Statecharts
 - statechart
 - Parameters
 - Links to agents
 - Simulation: Main

Person

income

sex

Susceptible

statechart

When this really connects,
The circle should be green
(see tip at end of presentation)

Palette

General

- Agent
- Parameter
- Event
- Dynamic Event
- Variable
- Collection
- Function
- Table Function
- Custom Distribution
- Schedule

Progress Properties

Person - Agent Type

Name: Person ☐ Ignore

Parameters preview

income:

☒ No delimiter ☐ Add separator ☐ Begin section

sex:

Agent actions

On startup

SimpleStatecharts

Time units: minutes

X=...14

Add a New State

The screenshot displays the AnyLogic Professional interface for editing a statechart. The main workspace shows a statechart with two states: 'Susceptible' and 'Infective'. A transition labeled 'statechart' connects the start to the 'Susceptible' state. The 'Infective' state is currently selected, and its properties are shown in the bottom panel.

Projects:

- SimpleStatecharts*
 - Main
 - Person
 - Presentation
 - oval
 - Statecharts
 - statechart
 - Unresolved
 - Parameters
 - Links to agents
 - Simulation: Main

Palette:

- General
 - Agent
 - Parameter
 - Event
 - Dynamic Event
 - Variable
 - Collection
 - Function
 - Table Function
 - Custom Distribution
 - Schedule

Properties Panel: Infective - State

Name: ☒ Show name ☐ Ignore

Fill color:

Entry action:

Exit action:

Description:

1 error(s)

Time units: minutes

Add a Transition

AnyLogic Professional

File Edit View Draw Model Tools Help

Projects

- SimpleStatecharts*
- Main
- Person
 - Presentation
 - oval
 - Statecharts
 - statechart
 - Parameters
 - Links to agents
 - Simulation: Main

Person Main

income

sex

statechart

Susceptible

Infective

When this really connects on both sides, circles should be green

Progress Properties

Person - Agent Type

Name: Person ☐ Ignore

Parameters preview

income:

☒ No delimiter ☐ Add separator ☐ Begin section

sex:

Agent actions

On startup:

Time units: minutes X=...=8

Tip: Beware Loose Connections

The screenshot displays the AnyLogic Professional interface. The main workspace shows a statechart diagram for a 'Person' agent. The diagram includes a start node, a 'statechart' block, and two states: 'Susceptible' and 'Infective'. A transition labeled 'transition1' connects 'Susceptible' to 'Infective'. A warning icon (a red circle with an exclamation mark) is visible near the transition, indicating a 'Loose Connection' error. The 'Properties' panel at the bottom shows the details for 'transition1 - Transition', including its name, trigger (Rate), and rate (0.1). The 'Description' panel is also visible. The 'Palette' on the right lists various modeling elements like Agent, Parameter, Event, etc. The status bar at the bottom indicates '2 error(s)' and 'Time units: minutes'.

AnyLogic Professional

File Edit View Draw Model Tools Help

Projects

- SimpleStatecharts*
- Main
- Person
 - Presentation
 - oval
 - Statecharts
 - statechart
 - Unresolved
 - Parameters
 - Links to agents
 - Simulation: Main

Person Main

income

sex

statechart

Susceptible

Infective

transition1 - Transition

Name: transition1 ☐ Show name ☐ Ignore

Triggered by: Rate

Rate: 0.1

Action:

Guard:

Description

SimpleStatecharts

2 error(s)

Time units: minutes

Corrected

AnyLogic Professional

File Edit View Draw Model Tools Help

Projects

- SimpleStatecharts*
 - Main
 - Person
 - Presentation
 - oval
 - Statecharts
 - statechart
 - statechart
 - Susceptible
 - transition
 - Infective
 - transition1
 - Parameters
 - Links to agents
 - Simulation: Main

Person Main

income

sex

statechart

Susceptible

Infective

Palette

General

- Agent
- Parameter
- Event
- Dynamic Event
- Variable
- Collection
- Function
- Table Function
- Custom Distribution
- Schedule

Progress Properties

transition1 - Transition

Name: transition1 ☐ Show name ☐ Ignore

Triggered by: Rate

Rate: 0.1

Action:

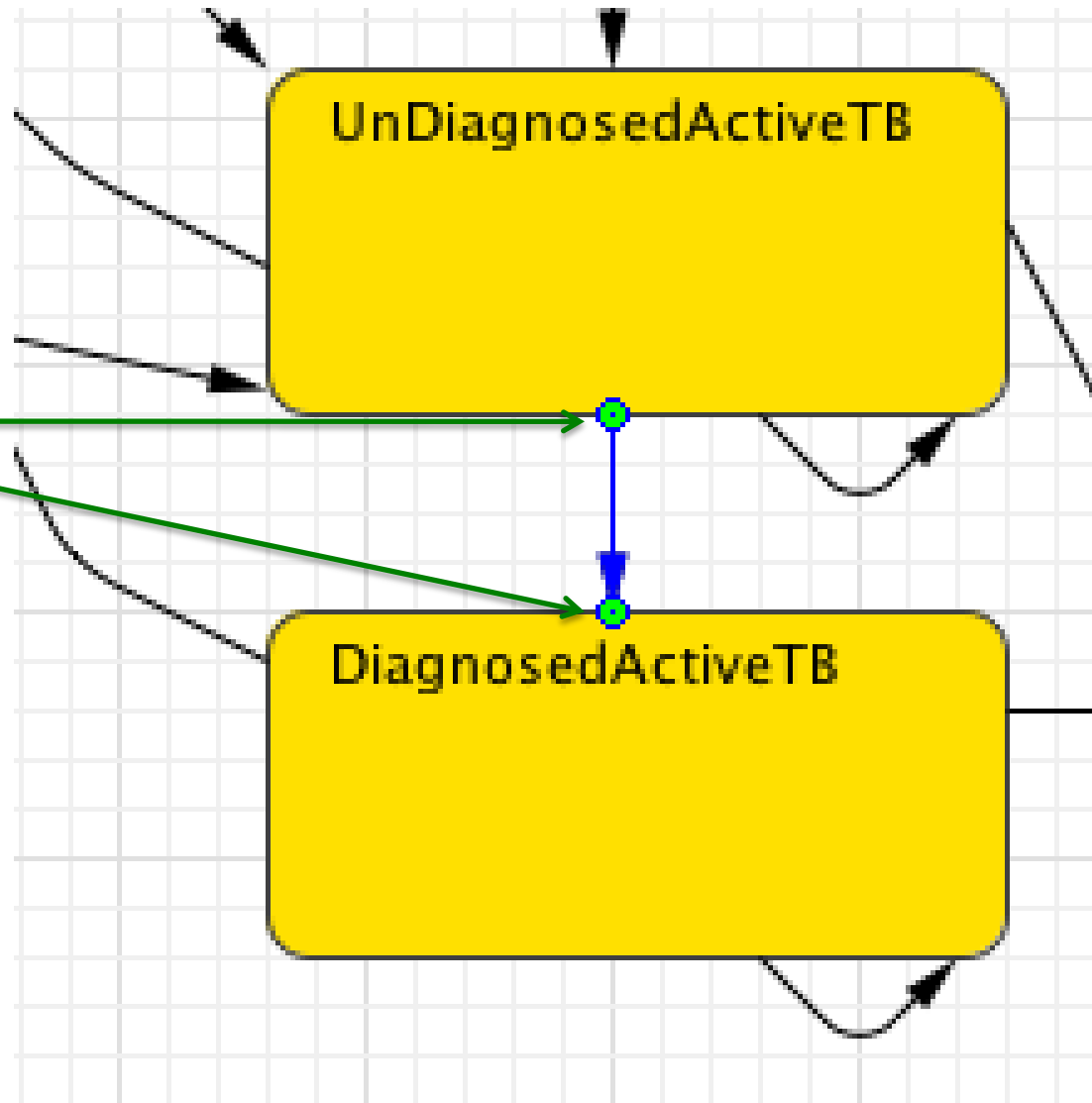
Guard:

Description

Time units: minutes

Tip: Confirming Transition Connectivity

- Ensure that both sides of the transition show **green circles** when connected
 - Otherwise, may appear connected but will actually be disconnected!



Add a Recovery Transition

The screenshot displays the AnyLogic Professional interface. The main workspace shows a statechart for a 'Person' agent. It includes a start node, an 'income' event, a 'sex' parameter, and two states: 'Susceptible' and 'Infective'. A transition labeled 'transition1' connects the 'Susceptible' state to the 'Infective' state. The 'Properties' panel at the bottom is open for 'transition1 - Transition'. It shows the following configuration:

- Name: transition1
- Triggered by: Timeout
- Timeout: 1
- Action: (empty field)
- Guard: (empty field)

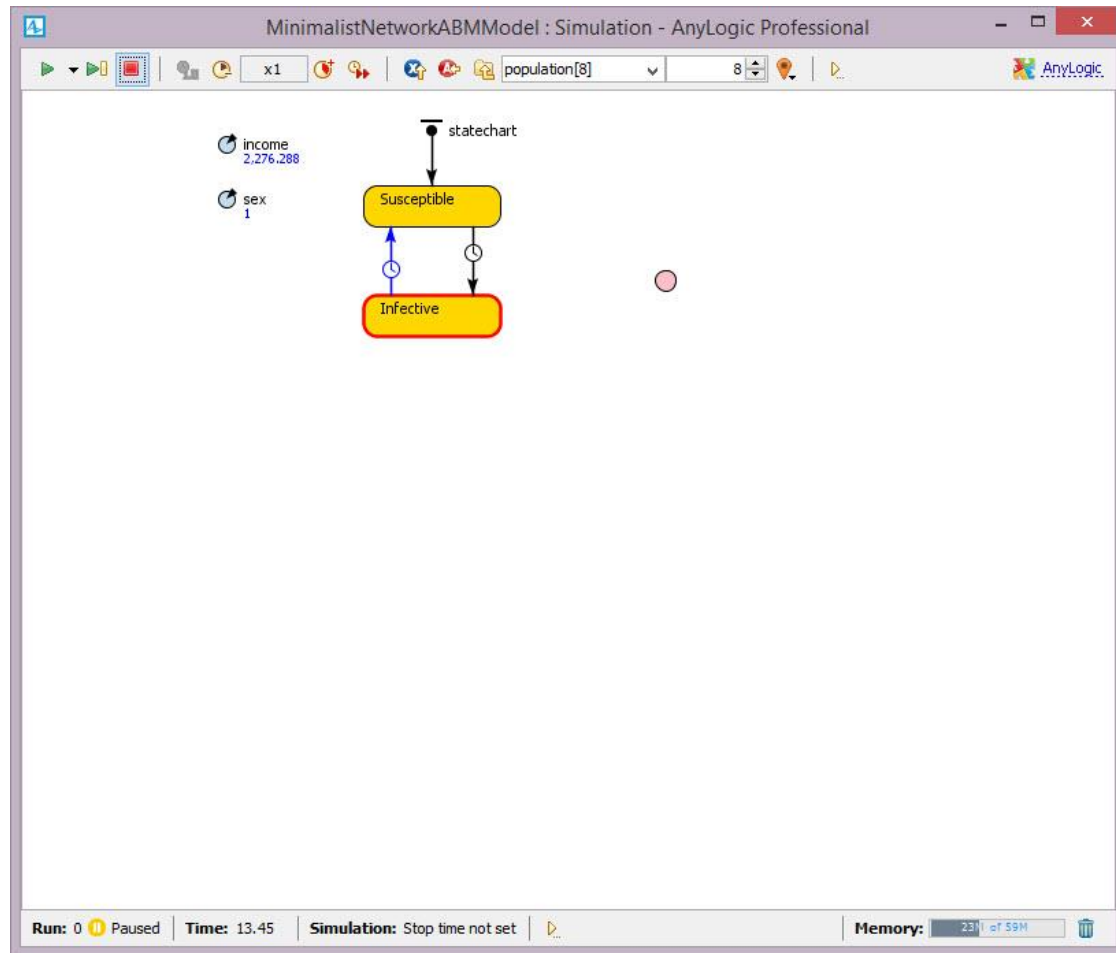
The 'Description' section is also visible but empty.

SimpleStatecharts

Time units: minutes

X=...12

Agent State Changes over Time



Add a Variable “color” of “type” “Color”

The screenshot displays the AnyLogic Professional interface. The main workspace shows a statechart diagram with states like 'Susceptible' and 'Infective', and variables like 'income', 'sex', and 'color'. The 'color' variable is highlighted with a red oval. A red arrow points from the text 'Make sure this is in lower case!' to the 'color' variable in the diagram. A green arrow points from the text '“Color” is the name of a Java class, and is thus capitalized' to the 'Color' type in the variable properties. The 'color - Variable' properties window is open, showing the variable name as 'color', visible as 'yes', type as 'Other', and initial value as 'BLACK'. The 'Color' type is highlighted with a green circle. The 'Initial value' field is highlighted with a blue oval. The 'Advanced' and 'Description' tabs are visible at the bottom of the properties window. The 'Palette' on the right shows various simulation components like Agent, Parameter, Event, etc. The 'Projects' pane on the left shows the project structure.

Make sure this is in lower case!

“Color” is the name of a Java class, and is thus capitalized

Fill in the type and Initial Value (watch for correct case!!)

Making “color” Green when Becoming Susceptible

The screenshot displays the AnyLogic Professional interface. The main workspace shows a statechart diagram with a 'Susceptible' state (yellow oval) and an 'Infective' state (yellow rounded rectangle). A transition arrow points from 'Susceptible' to 'Infective'. The 'Susceptible' state is connected to a 'statechart' block. The diagram also includes variables for 'income', 'sex', and 'color'.

The left sidebar shows the 'SimpleStatecharts*' tree structure:

- SimpleStatecharts*
 - Main
 - Person
 - Presentation
 - oval
 - Statecharts
 - statechart
 - statechart
 - Susceptible
 - transition
 - Infective
 - transition1
 - Parameters
 - Variables
 - color
 - Links to agents
 - Simulation: Main

The right sidebar shows the 'Palette' with various elements like Agent, Parameter, Event, Dynamic Event, Variable, Collection, Function, Table Function, Custom Distribution, and Schedule.

The bottom panel shows the 'Properties' tab for the 'Susceptible - State'.

Susceptible - State

Name: ☒ Show name ☐ Ignore

Fill color:

Entry action:

Exit action:

Description

Time units: minutes

Making “color” Red when Becoming Infective

The screenshot displays the AnyLogic Professional interface for modeling a simulation. The main workspace shows a statechart for a **Person** agent. The statechart starts at a start node and transitions to the **Susceptible** state. From **Susceptible**, a transition labeled **statechart** leads to the **Infective** state. The entry action for the **Infective** state is **color=RED;**. The **Properties** panel for the **Infective - State** shows the **Name** as **Infective**, **Show name** checked, **Fill color** set to **Default**, and the **Entry action** as **color=RED;**. The **Palette** on the right lists various modeling elements like **Agent**, **Parameter**, **Event**, **Dynamic Event**, **Variable**, **Collection**, **Function**, **Table Function**, **Custom Distribution**, and **Schedule**. The **Projects** panel on the left shows the project structure, including **SimpleStatecharts***, **Main**, **Person**, **Presentation**, **Statecharts**, **Parameters**, **Variables**, **Links to agents**, and **Simulation: Main**.

Click on the Oval

The screenshot displays the AnyLogic Professional interface. The main workspace shows a statechart diagram with two states, 'Susceptible' and 'Infective', connected by transitions. A blue oval icon is positioned on the left side of the diagram. The 'Properties' panel is open, showing the 'Oval' properties. The 'Name' is 'oval', and the 'Visible' checkbox is checked. The 'Appearance' section shows the 'Fill color' set to 'sex == 0? blue: pink' and the 'Line color' set to 'color'. The 'Line width' is 5 pt and the 'Line style' is solid. The 'Position and size' section is partially visible at the bottom.

AnyLogic Professional

File Edit View Draw Model Tools Help

Projects

- SimpleStatecharts*
- Main
- Person
- Simulation: Main

Person Main

income

sex

color

statechart

Susceptible

Infective

Palette

General

- Agent
- Parameter
- Event
- Dynamic Event
- Variable
- Collection
- Function
- Table Function
- Custom Distribution
- Schedule

Progress Properties

oval - Oval

Name: oval ☐ Ignore ☒ Visible on upper level ☐ Icon ☐ Lock

Visible: ☒ yes

Appearance

Fill color: sex == 0? blue: pink

Line color: color

Line width: 5 pt

Line style: solid

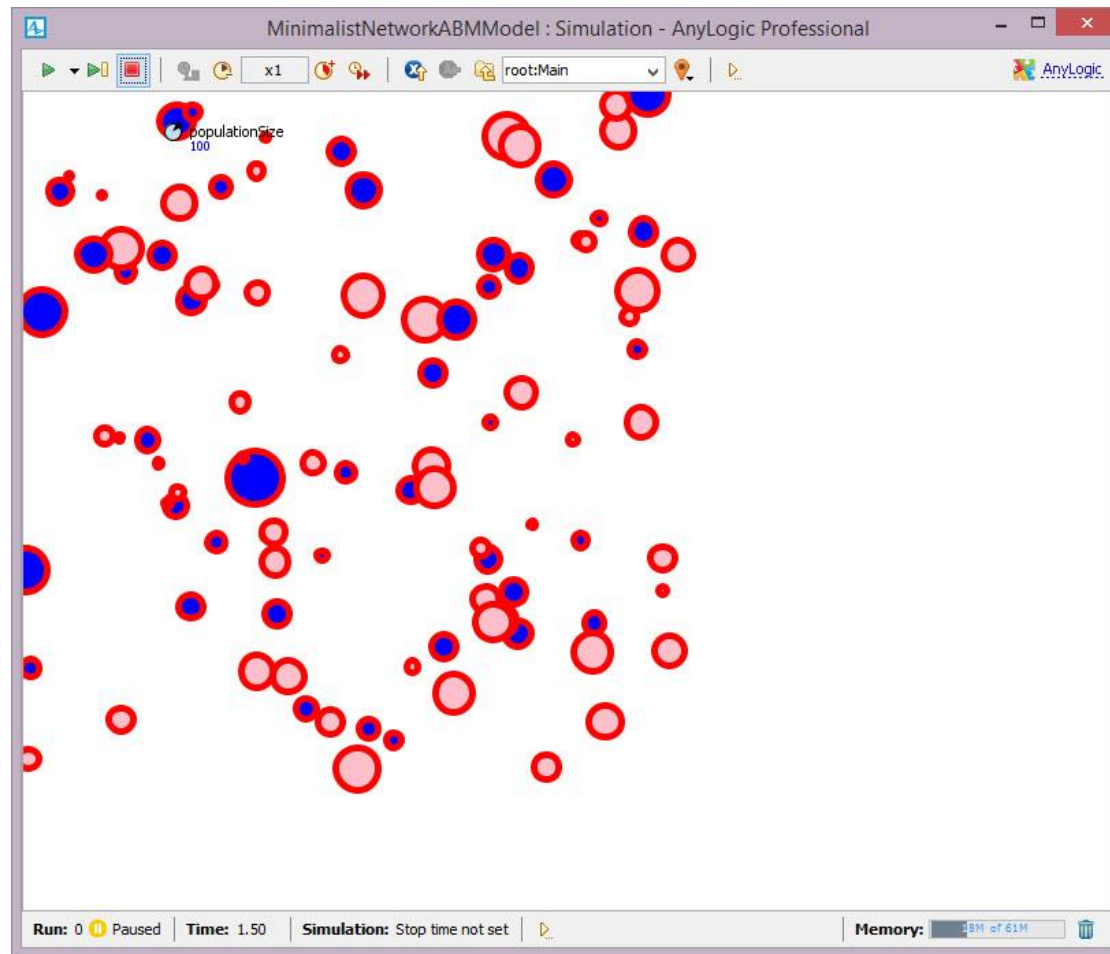
Position and size

SimpleStatecharts

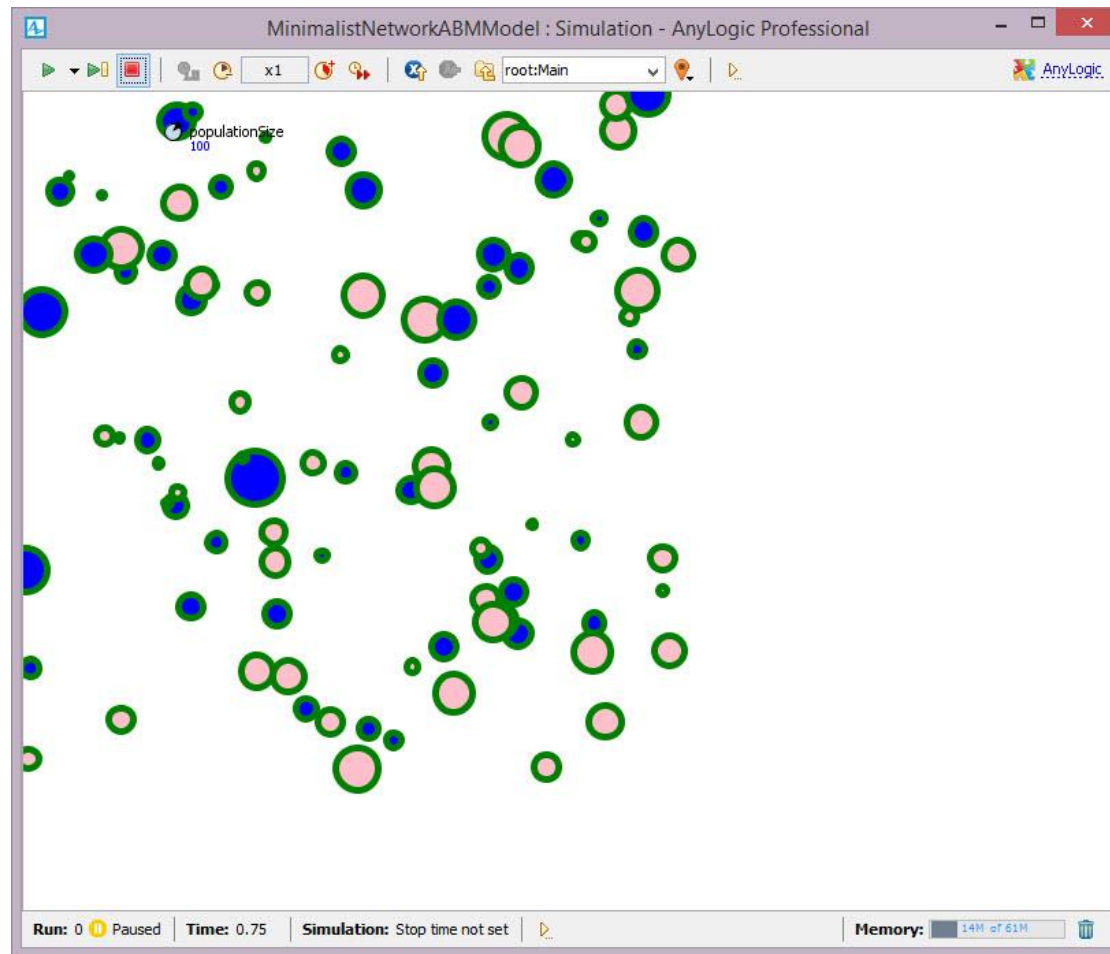
Time units: minutes

X=...=3

Colors Around Agents indicate State



Colors Around Agents indicate State



Discrete Agent Dynamics: Transitions

- Many transition conditions are possible
 - Timeout: Spending some period of time in the state
 - Fixed rate: Leave state with some fixed change per unit time
 - This is similar to “first order interarrival time”, and is conceptually linked to the operation of first-order delays in stock & flow diagrams
 - Variable rate: If desired, we can change the rate over time – but Anylogic only “notifies” changes when eg agent re-enters the state
 - Message received: We can transition when a message (any message or particular type of message) is received
 - Predicate: Only transition when condition becomes true
 - Arrival: Reach a location
- These transitions can be conditionally “routed” via branches
 - Conditions can determine to what destination state a particular transition will travel

Eliminate the Lock-Step Transitions by Using a Probabilistic Transition (Hazard)

When this really connects on both sides, circles should be green

transition - Transition

Name: transition ☐ Show name ☐ Ignore

Triggered by: Rate

Rate: 0.01

Action:

Guard:

Description

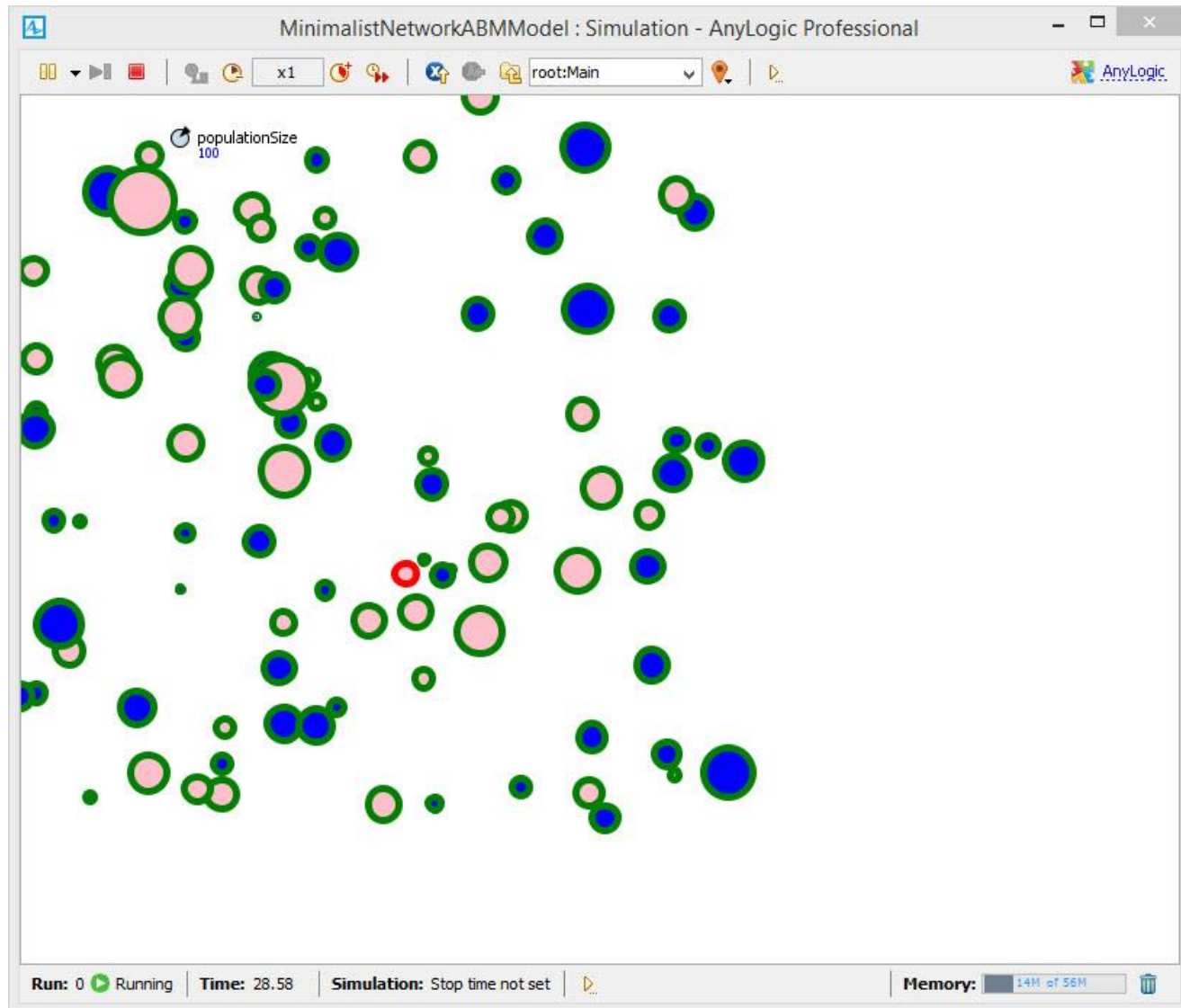
This implies mean time
Susceptible = 100

SimpleStatecharts

Time units: minutes

X=...51

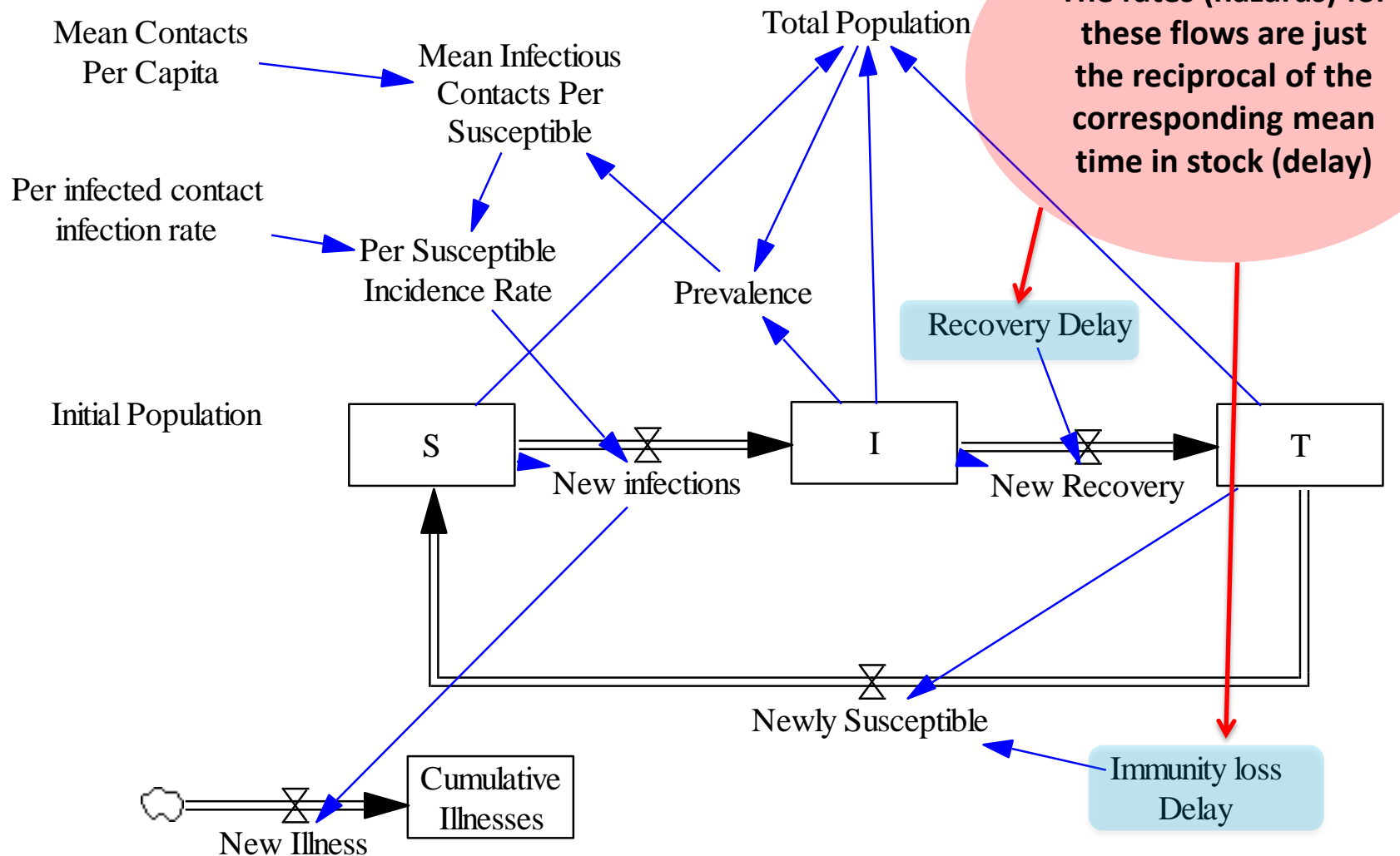
Result



Rates & Flows

- Some may have seen fixed rates before – in the form of “transition rates” in Compartment models
- Within a Compartment/SD model, a flow out of a stock was commonly set by the multiplication of the
 - State variable (Stock)
 - Some rate of transition
- We use different names for these rates
 - “Transition rates”
 - “Likelihood of transition per *Unit Time*”
 - Transition (e.g. “infection”, “mortality”) “hazard”

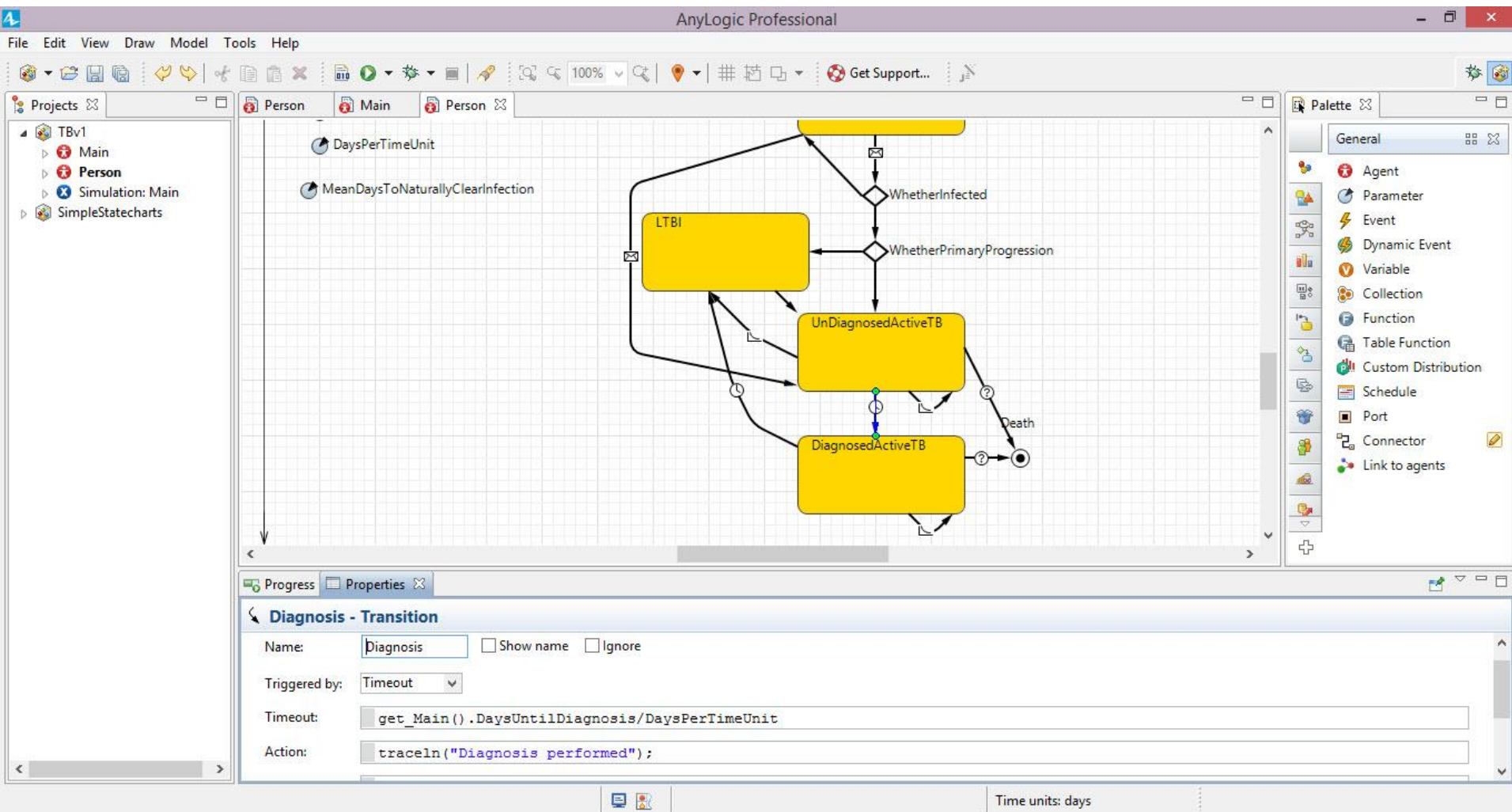
First Order Delays in Action: Simple SIT Model



Fixed Rates: Transition “Hazards”

- With “fixed rates”, we are specifying rates of transitions
- Because we are dealing with the chance that each individual transitions, we don’t need to multiply by the number of people at risk
 - Here, there is just 1 person at risk!
- As in Compartment models, these rates can change over time, but the statechart needs to be “made aware” of these changes (see later)
 - Leave & go back into current state (circular transition)
 - Trigger “change” event in Agent

Transition Type: Fixed Residence Time (Timeout)



Example of Processes Associated with Fixed Timeouts

- Aging
- Tightly defined time constants associated with natural history
 - While these may be described as associated with a broad distribution (e.g. with a 1st or 2nd order delay), much of that variability may be due to heterogeneity
 - *For a given person, these may be quite specific in duration \Rightarrow Can capture through a timeout*

What Happens if this Depends on a Timeout?

- Set the “Infection” transition to Trigger based on a “Timeout”

Progress Properties

Infection - Transition

Name: ☐ Show name ☐ Ignore

Triggered by:

Timeout:

Action:

Guard:

► Description

This will report when transition occurs

- Make the “Timeout” 100

Now run the model, and observe the difference

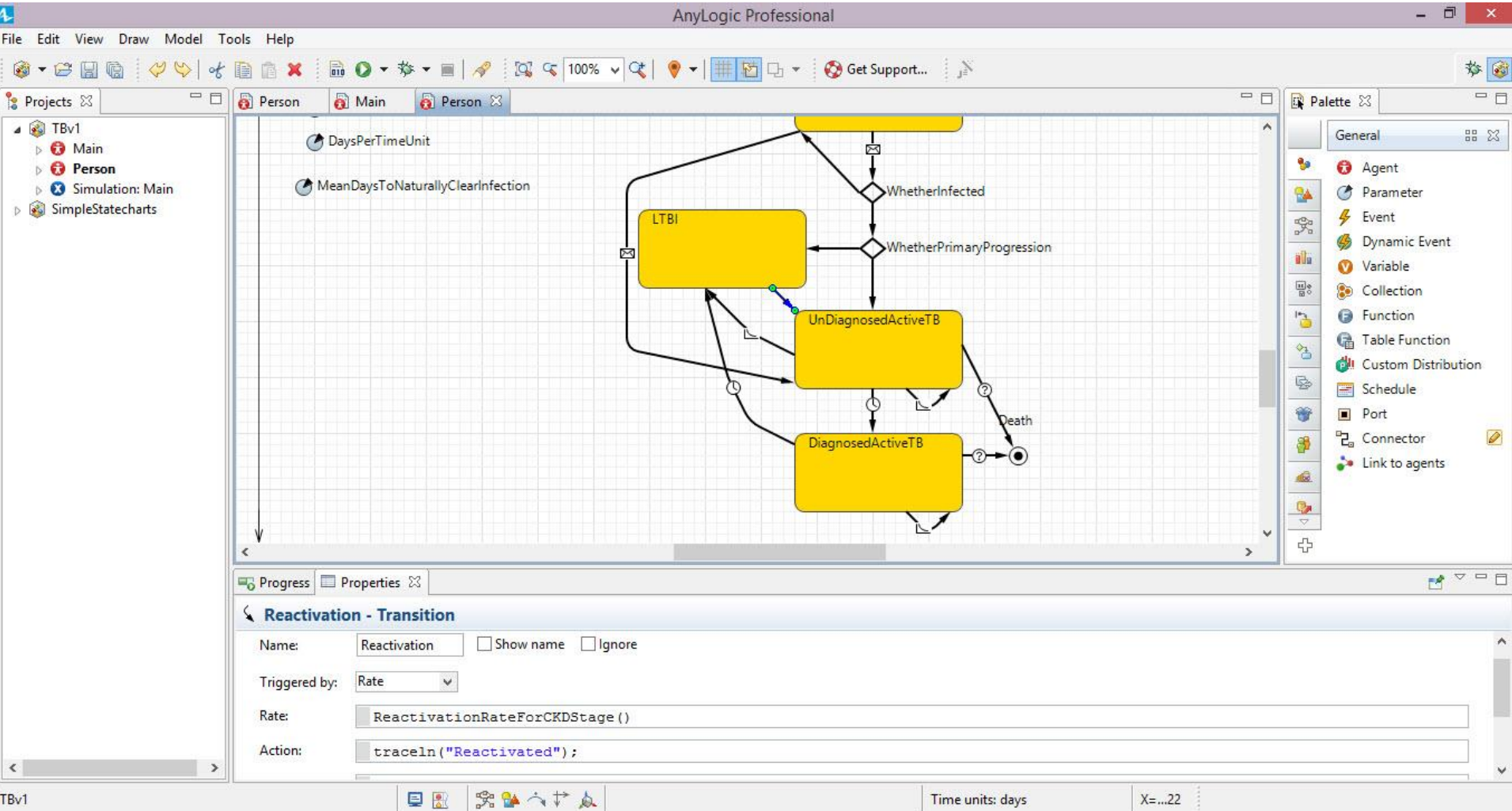


Hands on Model Use Ahead

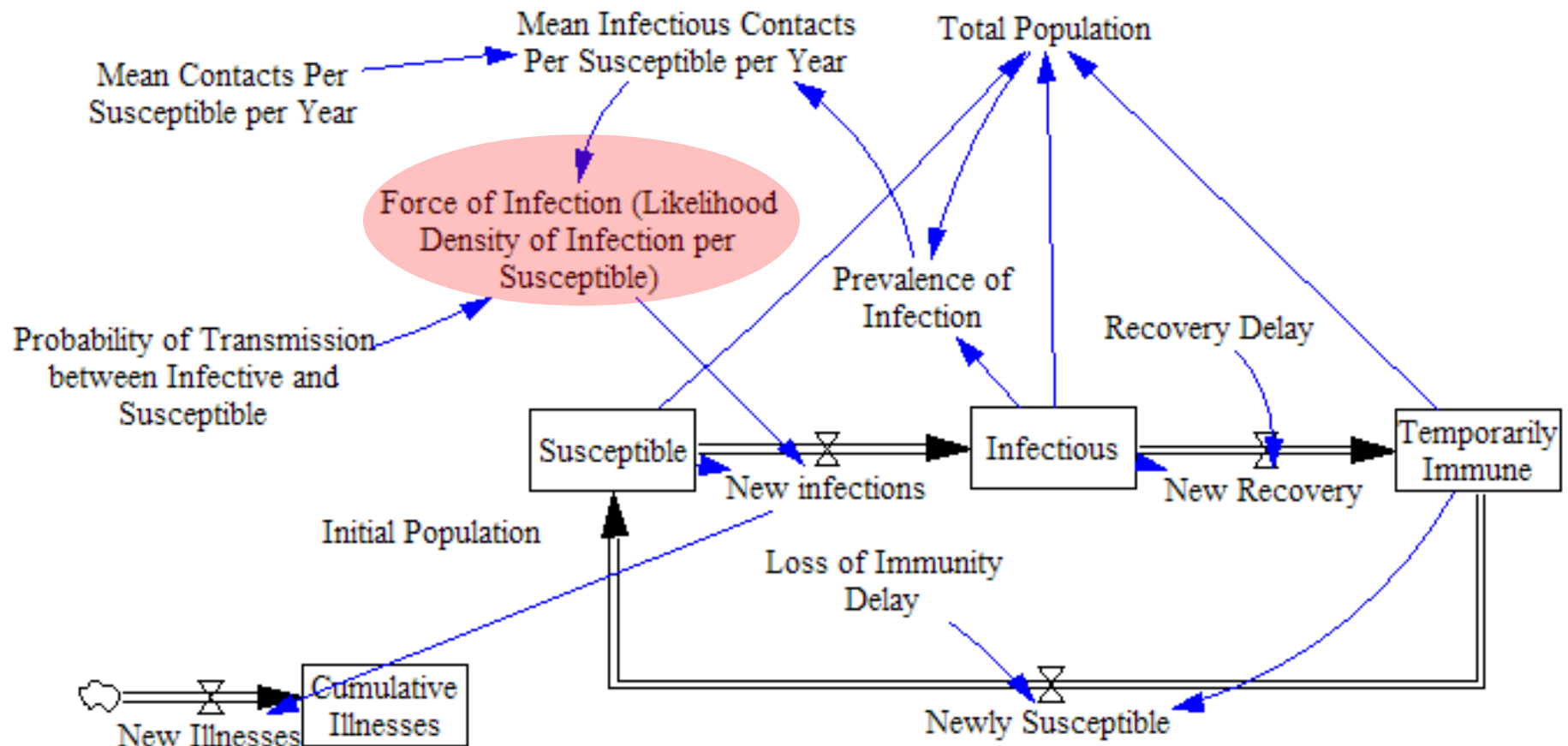


Load model: TBv1.alp

Transition Type: Variable Rate



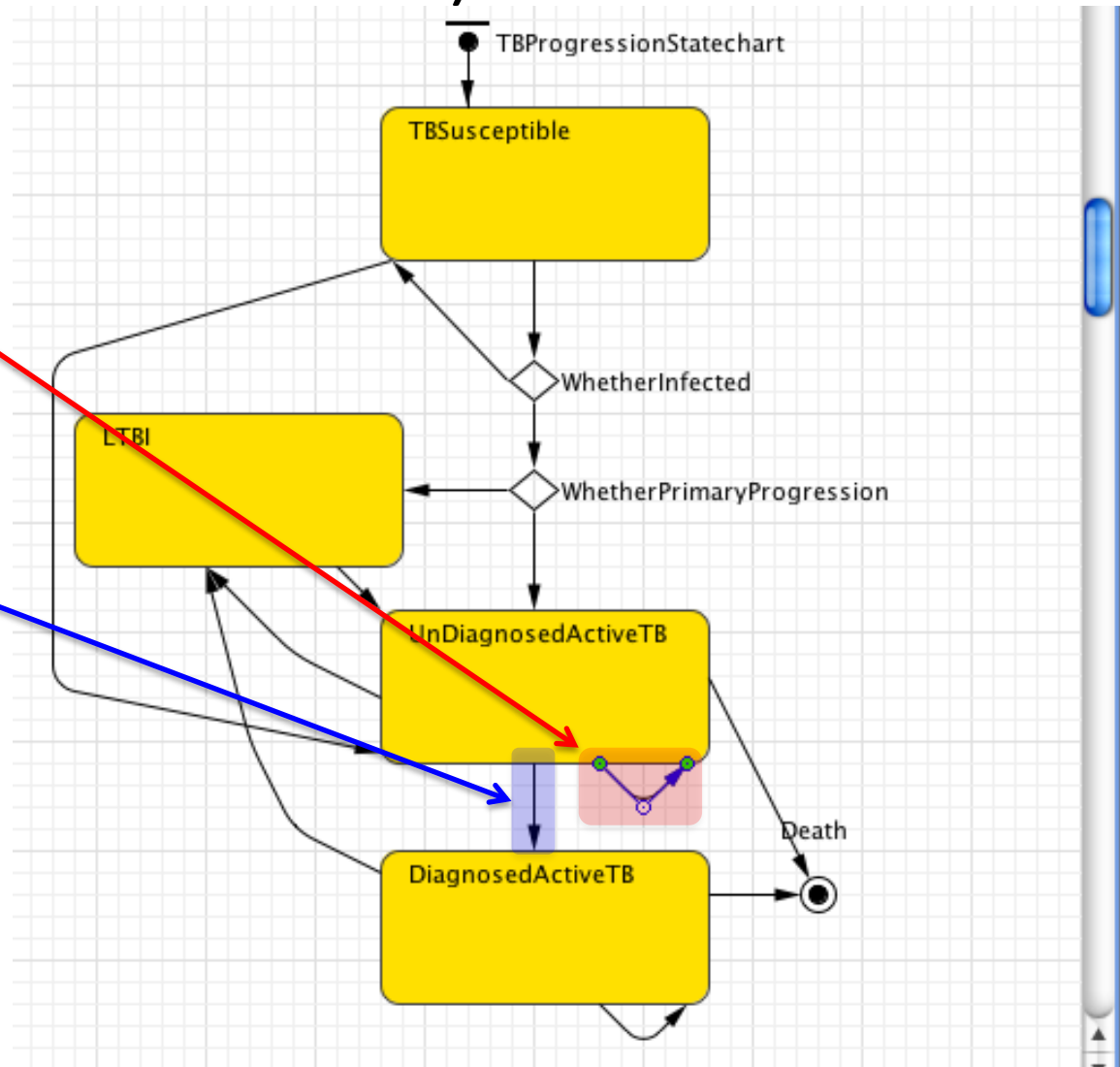
Example Transition Rate/Hazard



Special Elements: Self-Transition

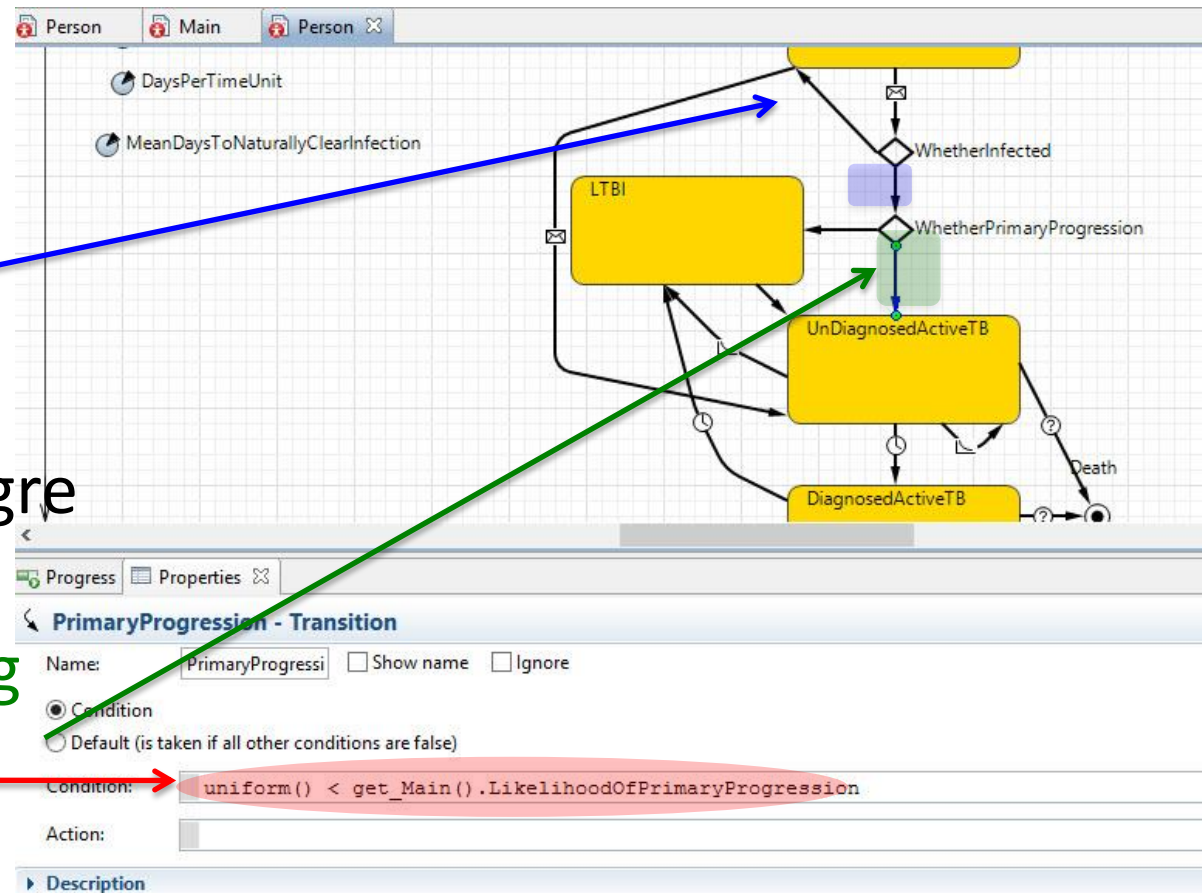
(Use if Wish To Have State Register Changing Out-transition rates)

The **self-transition** will “make the state realize” that the rate associated with any out transition (e.g. **this one**) has changed

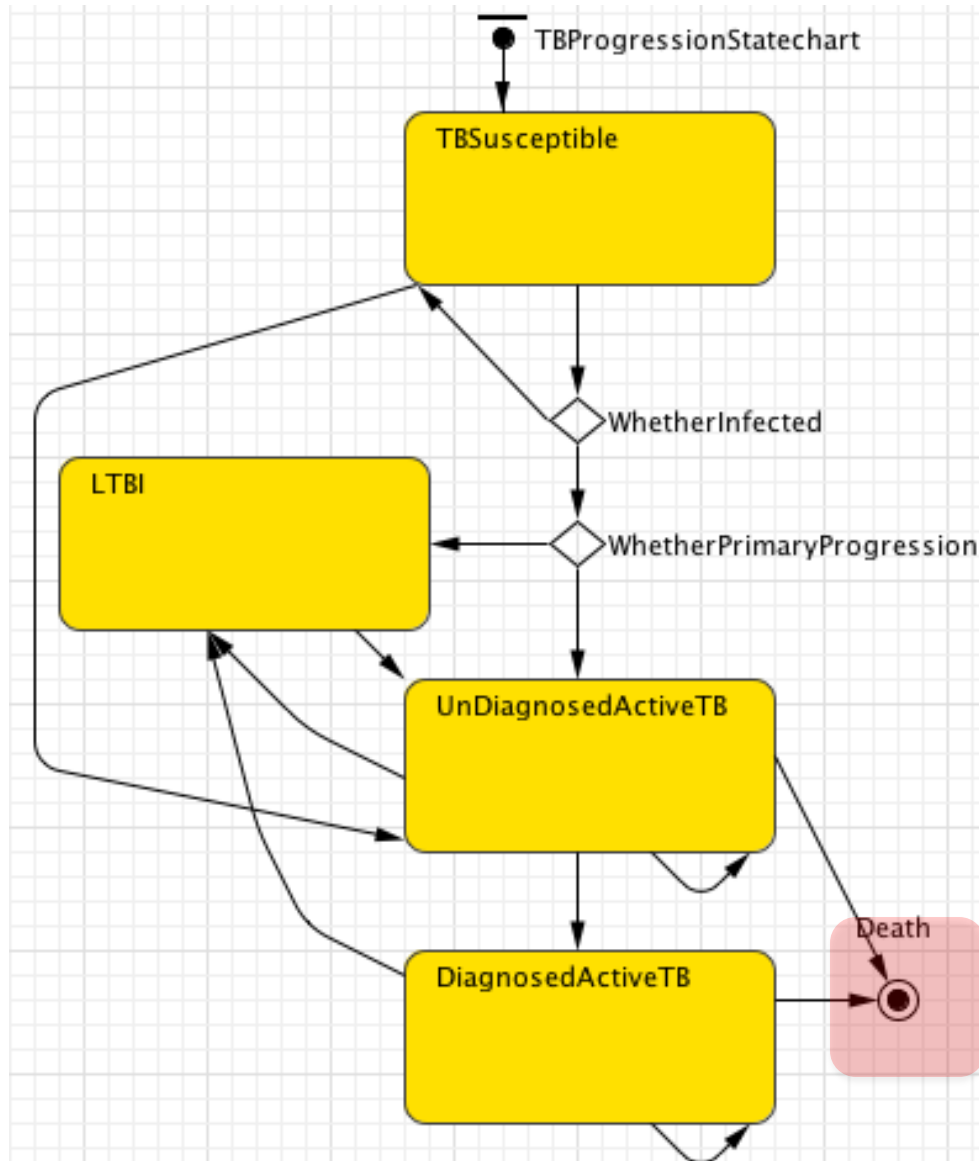


Example Conditional Transition

The incoming transition into “WhetherPrimaryProgression” will be routed to this outgoing transition if this condition is true



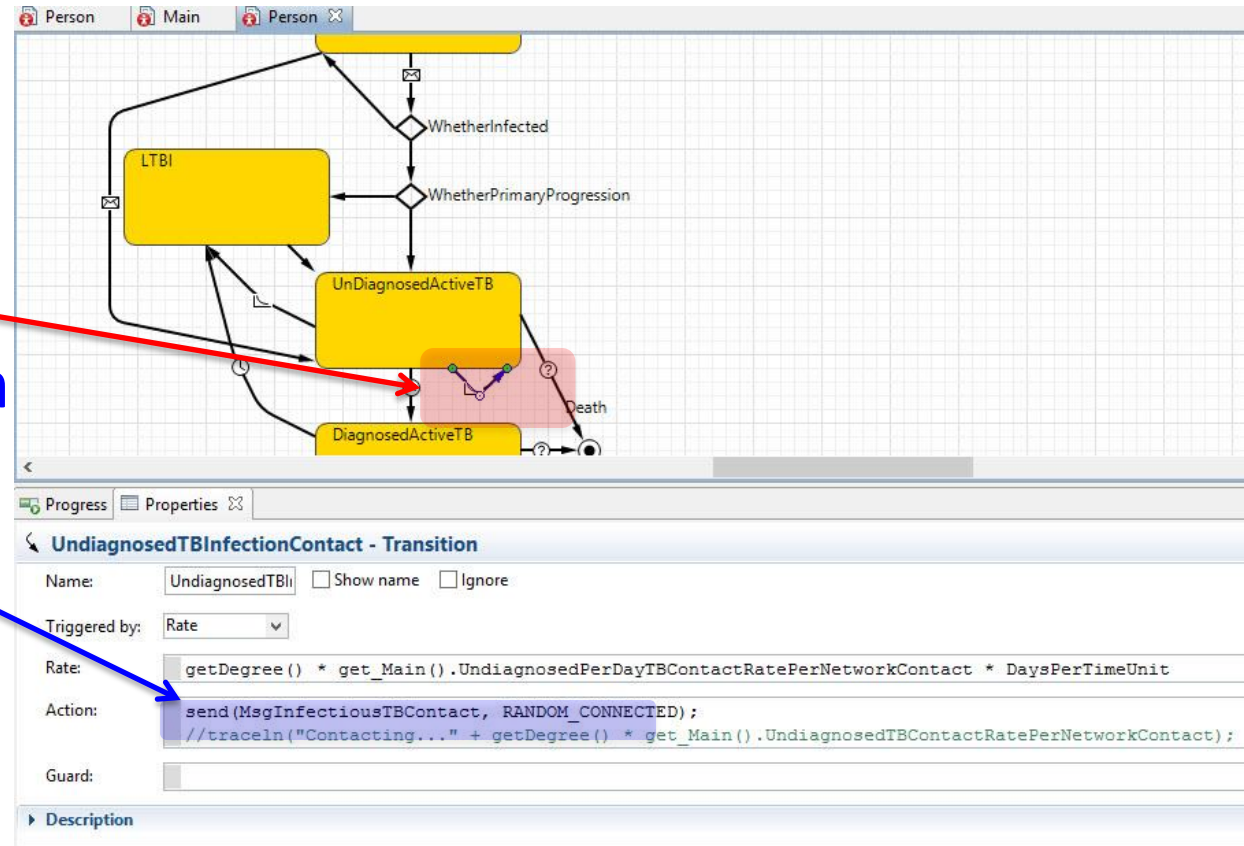
Special Elements: Exit Point



Special Elements: Self-Transition

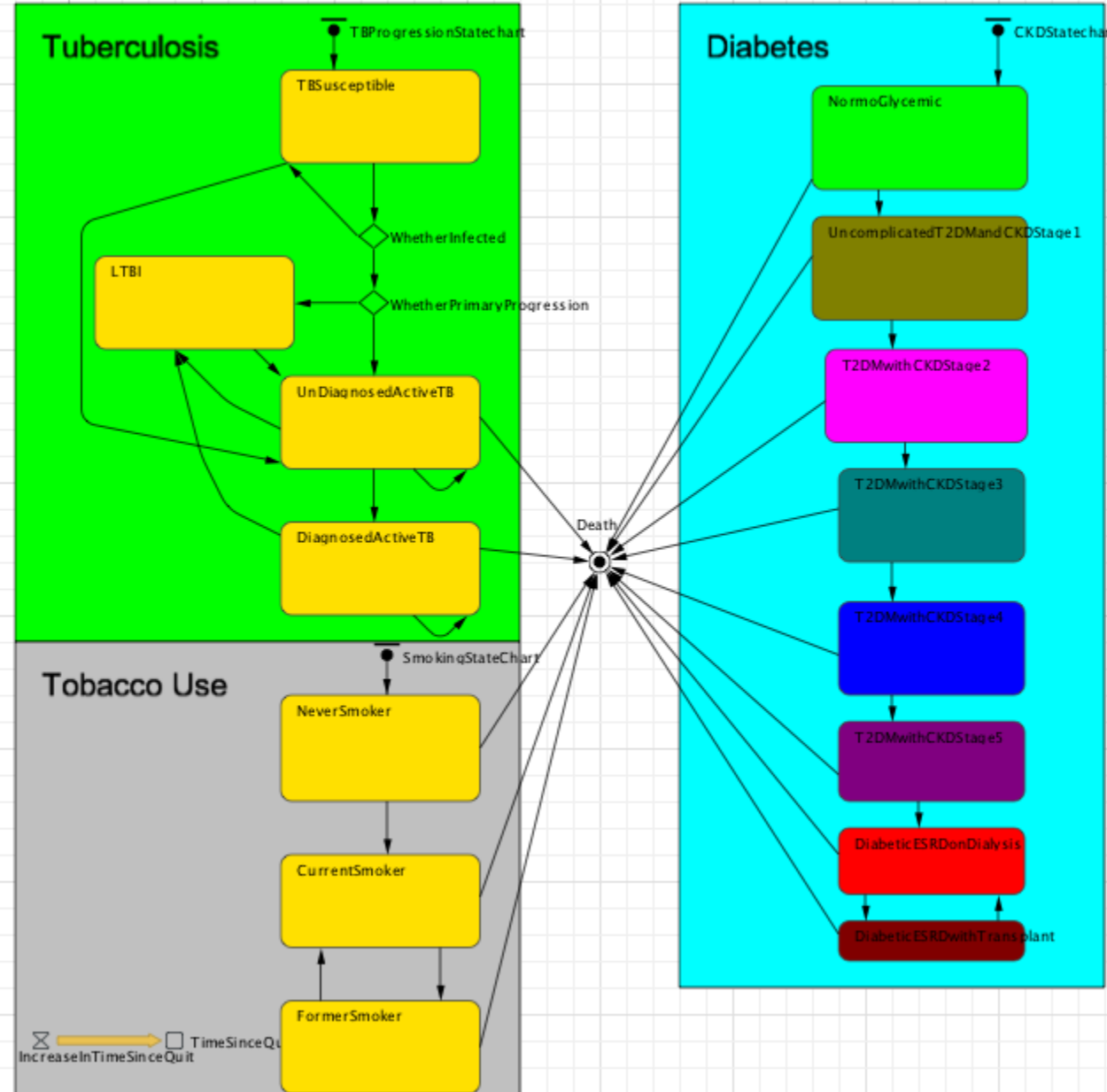
(Use if Wish To Trigger an Action w/o Leaving State)

The **self-transition** will invoke **this action** when it occurs



Parallel Statecharts

- By default, each statechart evolves independently.
- If coupling is desired, can make transitions/actions dependent on state of other statecharts

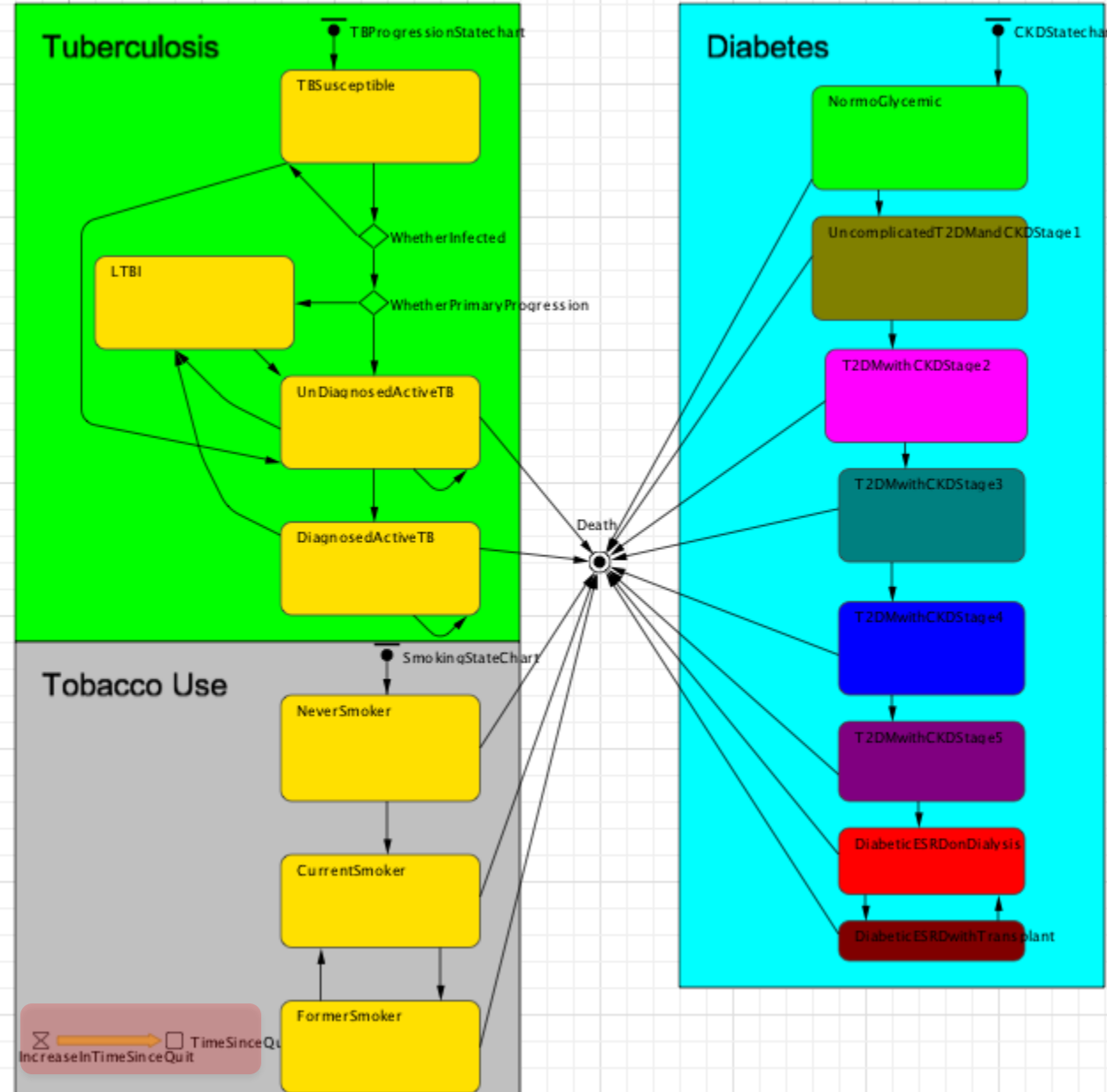


Comparison with Aggregate Stock & Flows

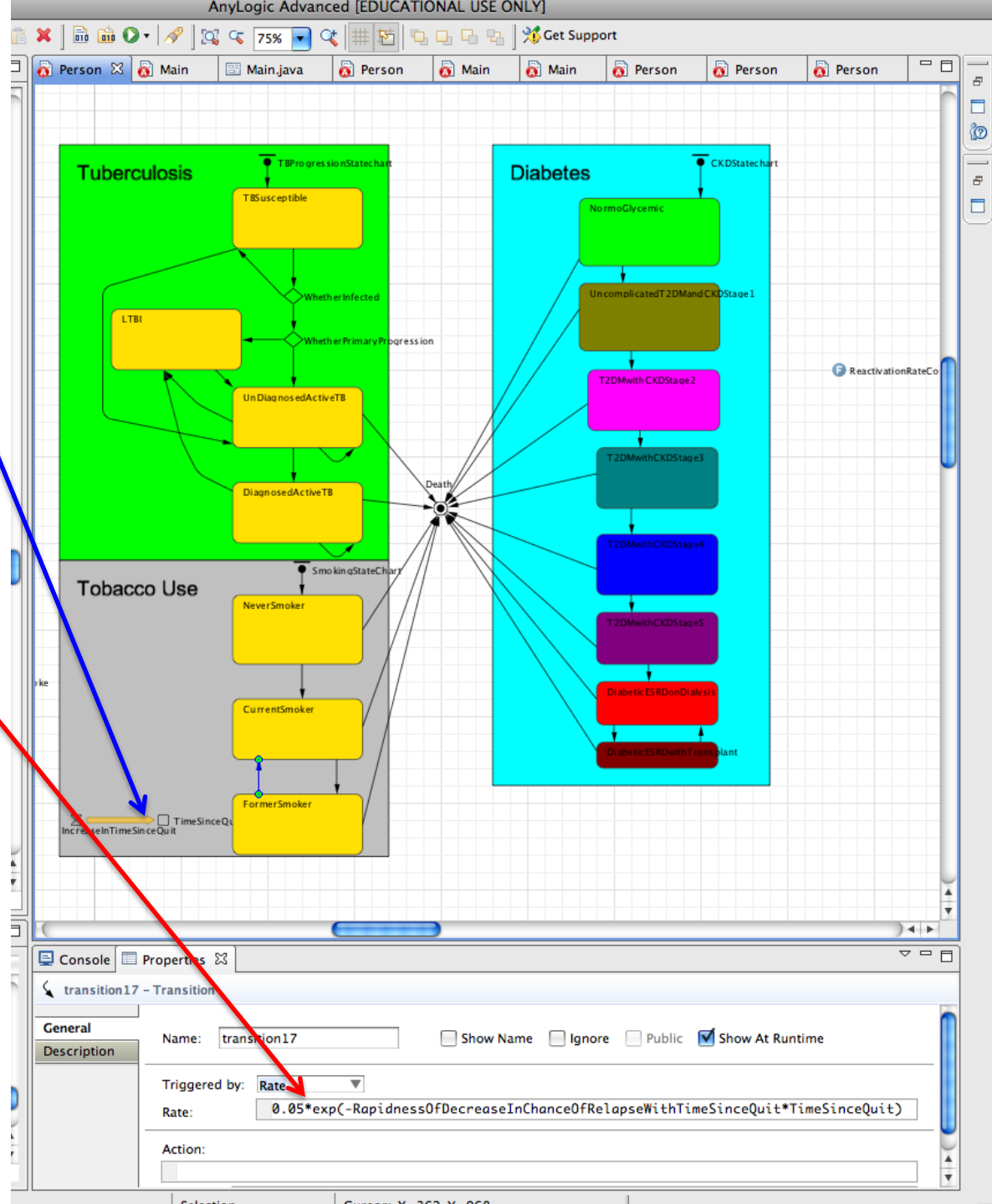
- As for aggregate stocks & flow, individuals' states are discrete
- Unlike aggregate stocks & flows
 - One state within a given statechart is active at a time
 - For parallel flows (e.g. comorbidities), there is no need for considering all combinations of the possible states
 - We can keep track of how long an individual is in a given state & adjust the transition rate accordingly

Parallel Transitions

- Example recording the residence time in a state (via a stock with unit inflow -- i.e. just accumulates the time present in that state)



- The **residence time** in the state determines the **transition rate** out of that state.
- Transition rates depending on residence time are generally not possible with aggregate models





Hands on Model Use Ahead



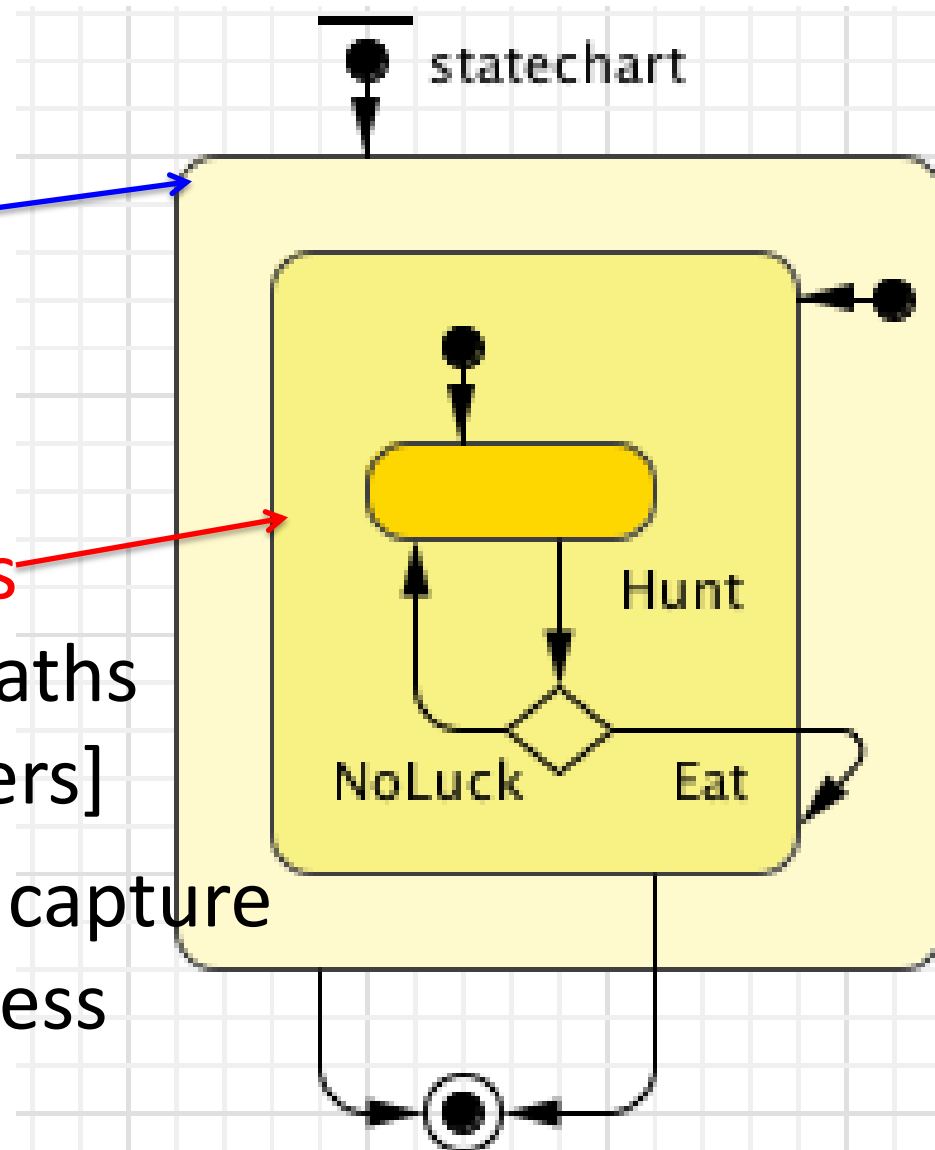
Load Sample Model:

Predator-Prey Agent Based

(Via “ExamplesModels” under “Help” Menu)

Advanced Element: Hierarchical States

- The **outermost state** captures time since born (for natural deaths)
- The **middle-state captures** time since last ate (for deaths by hunger). [Eating reenters]
- The inner state transition capture hunting frequency & success



Natural Death Transition

AnyLogic Professional

File Edit View Draw Model Tools Help

Projects

- Predator Prey Agent Based
 - Hare
 - Lynx**
 - Main
 - Simulation: Main
 - SimpleStatecharts

Width HaveBabies

CellWidth

cell

statechart

Hunt

NoLuck

Eat

tran3 - Transition

Name: tran3 ☐ Show name ☐ Ignore

Triggered by: Timeout

Timeout: `get_Main().LynxLifeExpectancy`

Action:

Guard:

Description

Predator Prey Agent Based

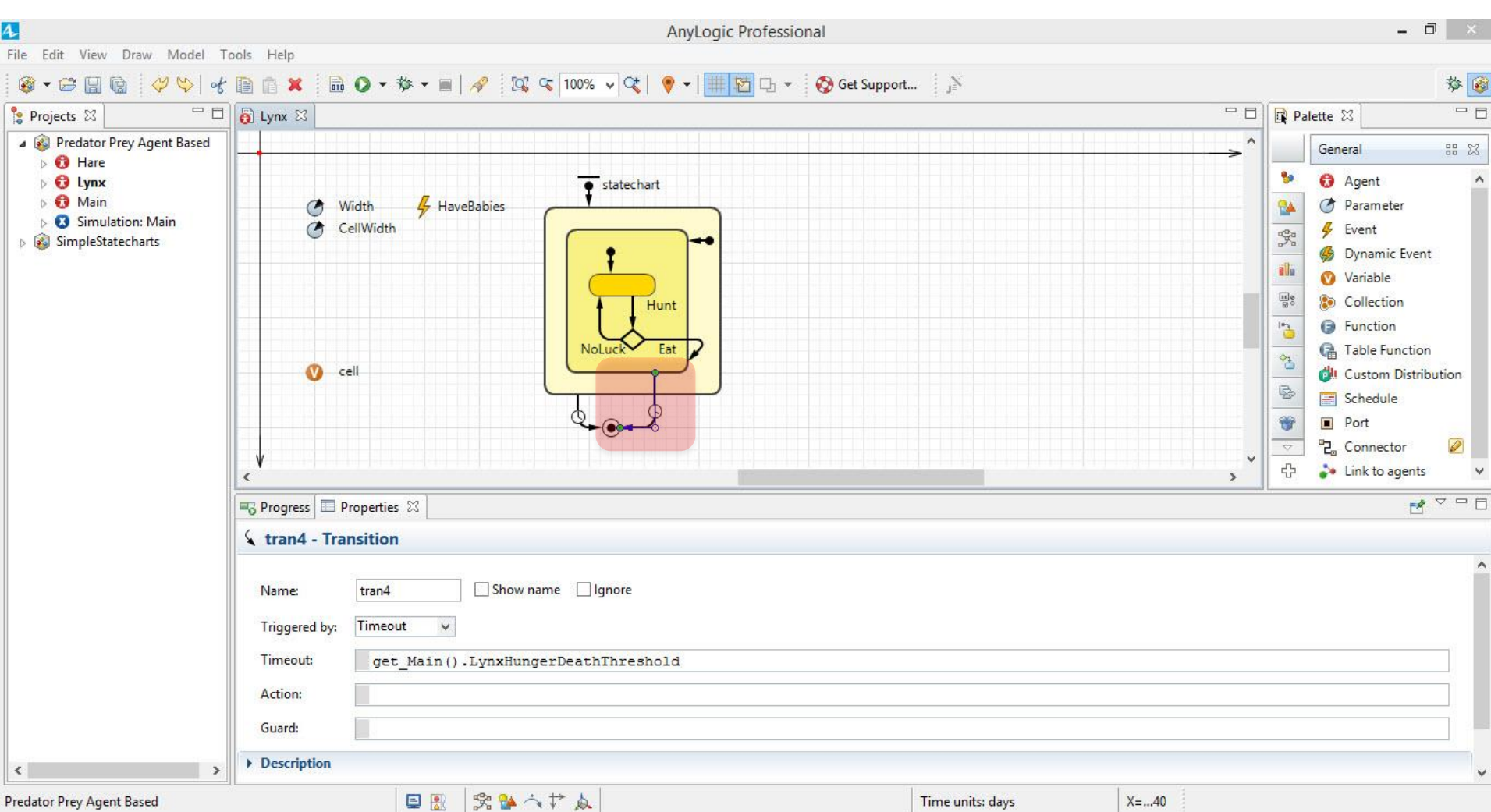
Time units: days

X=...84

The screenshot displays the AnyLogic Professional interface. The main workspace shows a statechart for a 'Lynx' agent. The statechart has three states: 'Hunt' (a yellow rounded rectangle), 'NoLuck' (a yellow rounded rectangle), and 'Eat' (a yellow rounded rectangle). Transitions are labeled 'Hunt' (from NoLuck to Hunt), 'NoLuck' (from Hunt to NoLuck), and 'Eat' (from Hunt to Eat). A transition named 'tran3' is highlighted with a red box, leading from the 'Eat' state to a final state. The bottom panel shows the properties for 'tran3 - Transition': Name: tran3, Triggered by: Timeout, Timeout: `get_Main().LynxLifeExpectancy`, Action: (empty), Guard: (empty). The left sidebar shows the project structure: Predator Prey Agent Based, Lynx, Main, Simulation: Main, SimpleStatecharts. The bottom status bar shows 'Time units: days' and 'X=...84'.

Death By Hunger

(Note that Depends on Time in State – i.e. time Since last ate)



Eating Transition Leaves & Reenters Middle State

The screenshot displays the AnyLogic Professional interface. The main workspace shows a statechart for a Lynx agent. The statechart is a yellow box labeled 'statechart' containing a 'Hunt' state (yellow oval), a 'NoLuck' state (yellow oval), and an 'Eat' state (red oval). The 'Eat' state is highlighted. The 'Eat' transition is detailed in the bottom panel, showing a condition and an action.

Projects: Predator Prey Agent Based
Hare
Lynx
Main
Simulation: Main
SimpleStatecharts

Properties: Width, CellWidth, HaveBabies, cell

Statechart: statechart

Transitions: Hunt, NoLuck, Eat

Eat - Transition

Name: Eat ☒ Show name ☐ Ignore

☒ Condition
☐ Default (is taken if all other conditions are false)

Condition: `randomTrue(min(1, get_Main().HaresInCell[cell].size() / (0.5*(double)get_Main().HareMaxPerCell)))`

Action: `Main m = get_Main();
Hare prey = m.HaresInCell[cell].get(uniform_discr(0, m.HaresInCell[cell].size()-1));
prey.statechart.receiveMessage(new Object());`

Description

Time units: days

Tips on Statechart Code

- Each State & Transition has an integer index
 - This by accessed via a (static) constant holding the name of state within the statechart class (*statechart.StateName*)
- To determine length of time spent in state
 - *Statename.getLocalTime(StateIndex)*
- To determine current state
 - *statechart.getActiveSimpleState()*
- To find out if a state (either simple or composite) is currently active
 - *statechart.isStateActive(StateIndex)*