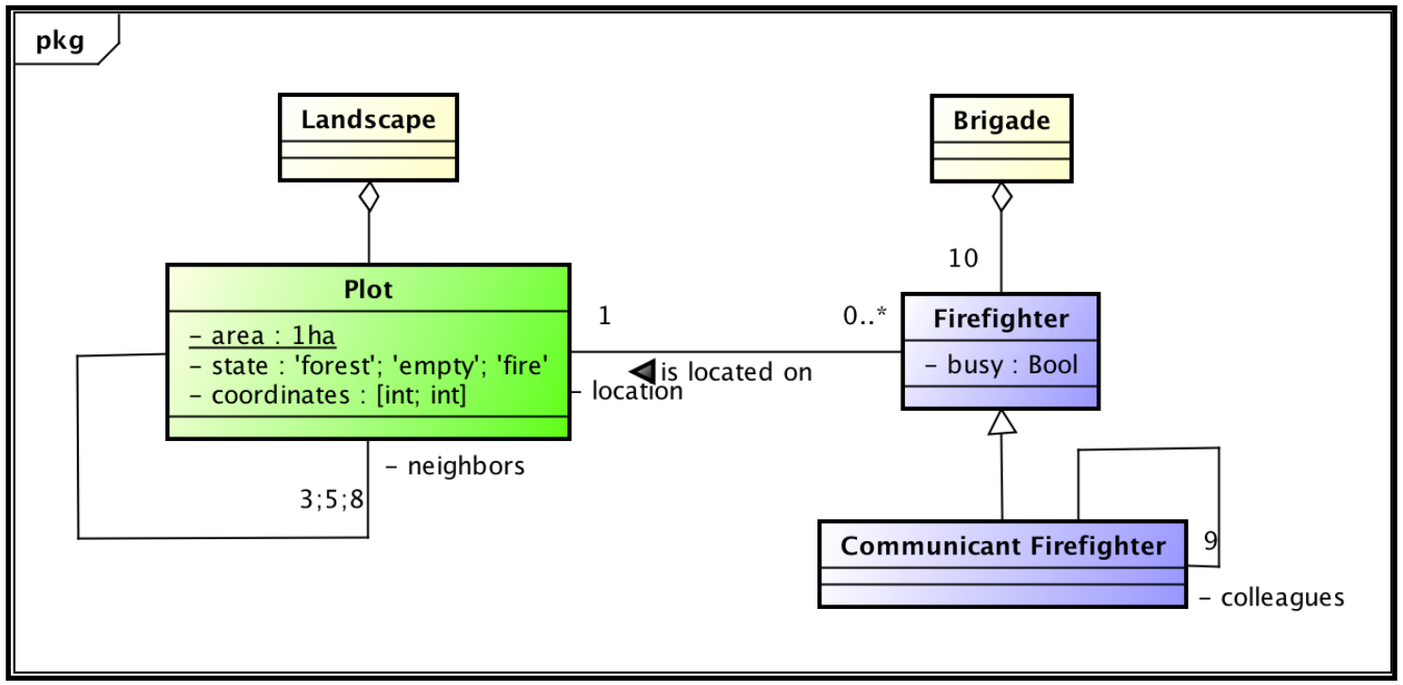
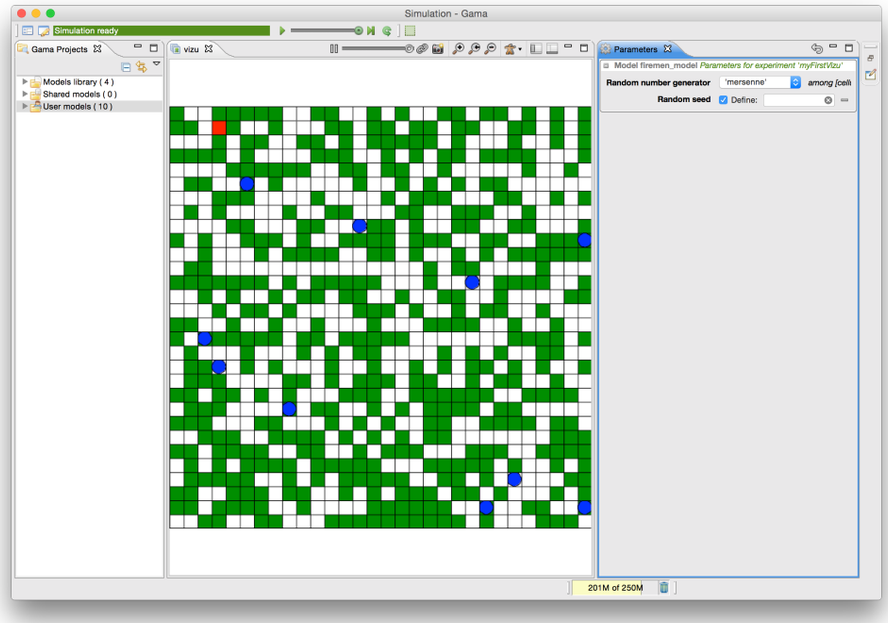
Firefighter model: Structure and initialisation





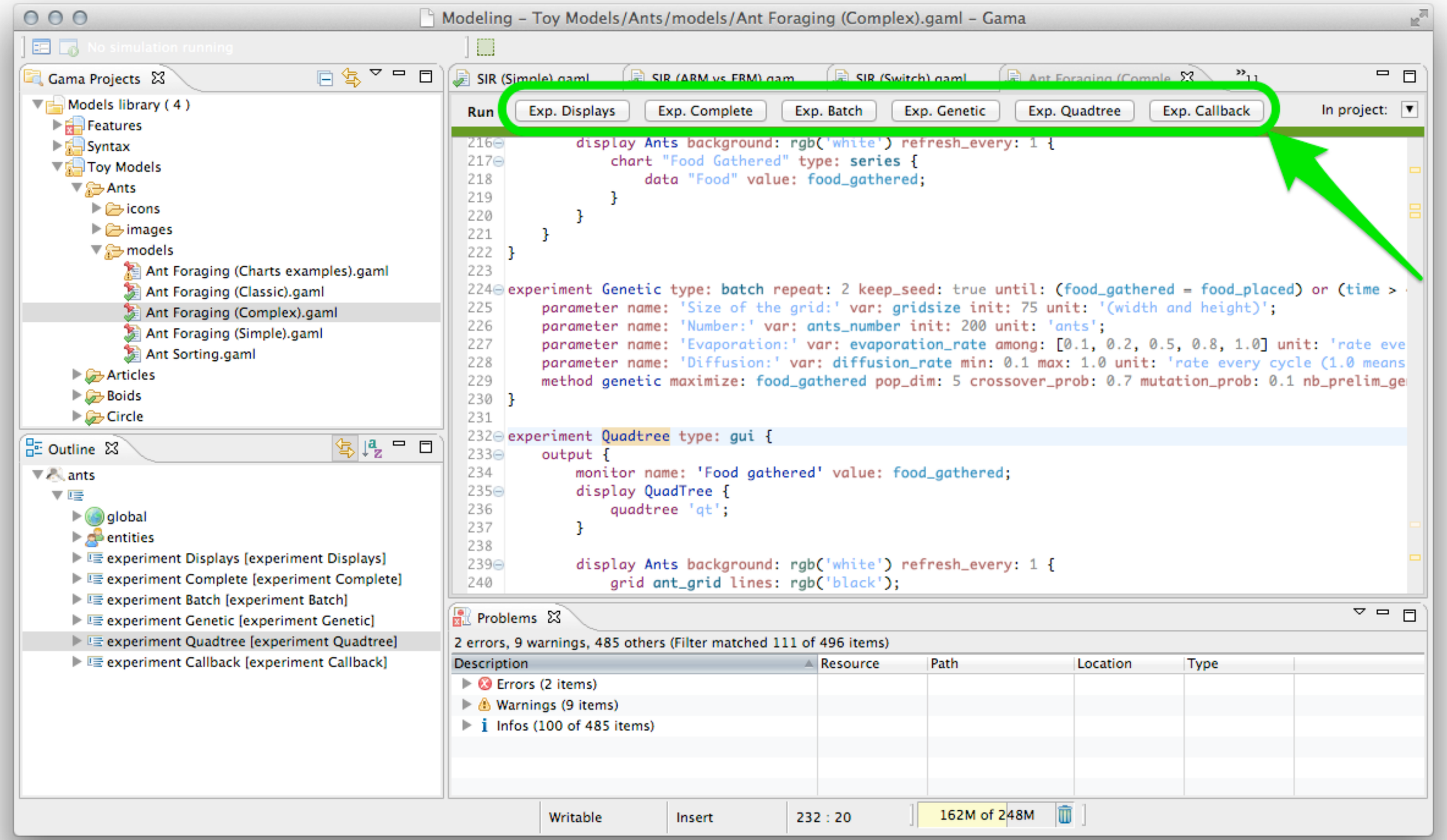
Benoit Gaudou (Univ. Toulouse 1)
Patrick Taillandier (INRAE)

GAMA: first demo

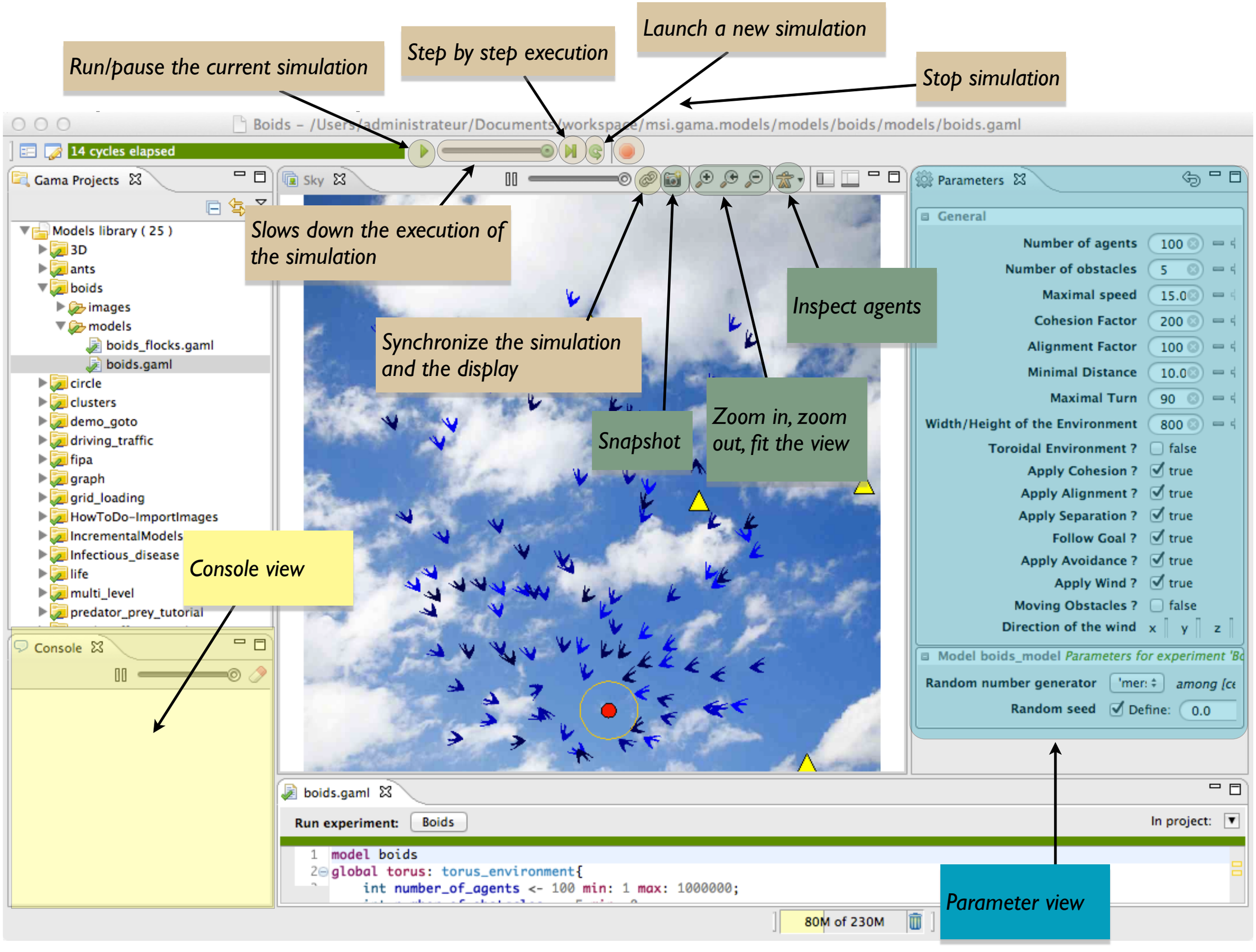
- Run GAMA
  - workspace
- Open a model in the model library
  - show editor
    - compilation errors
  - run it (show the link between button and experiments)
    - simple example
    - multiple display example
    - batch mode
- Create a model
  - Create a first GAMA project
  - Create a first GAMA model

Loading an experiment

Click on the desired **experiment** button to load it: an experiment define a simulation execution context

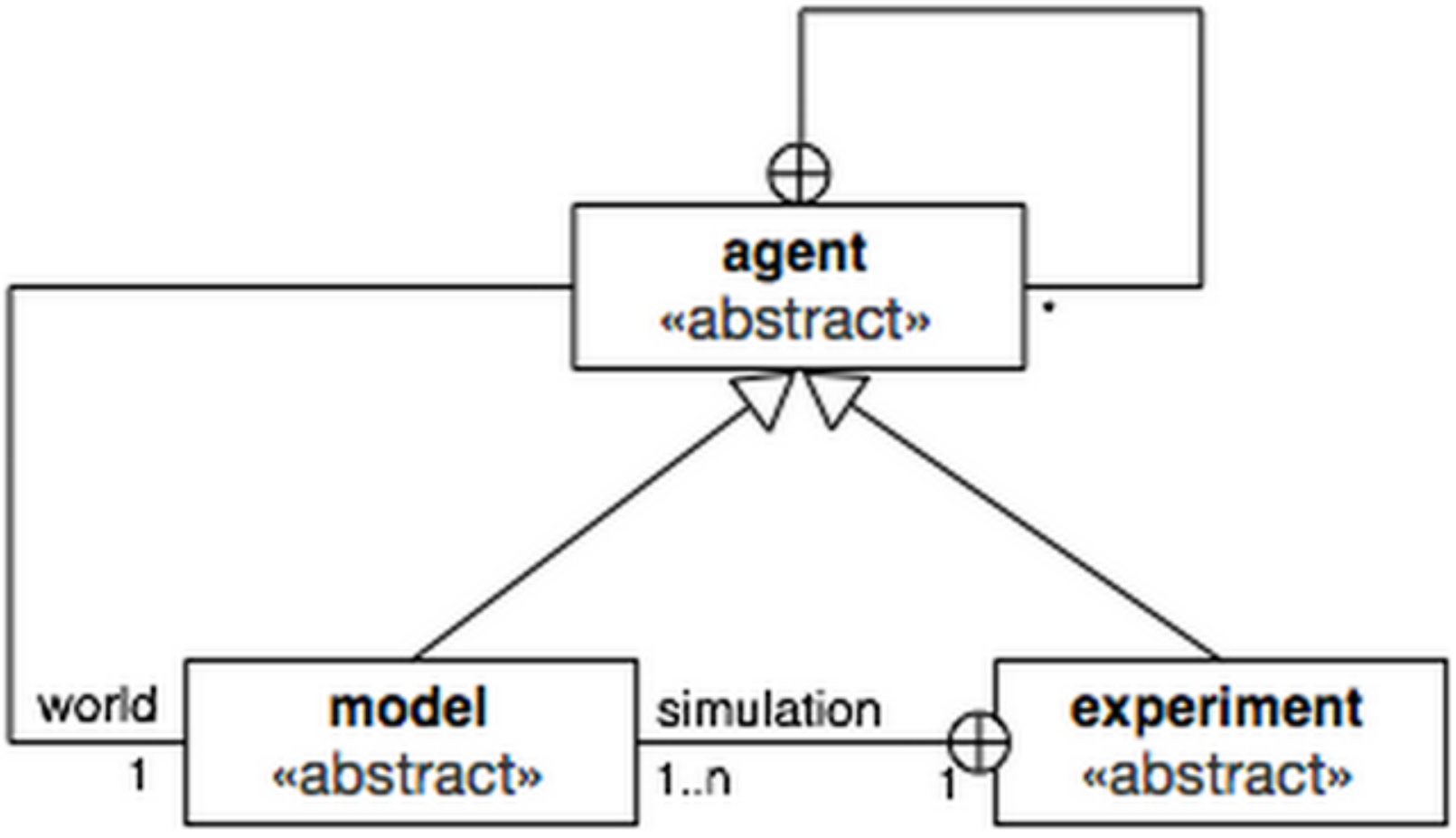


Simulation Interface



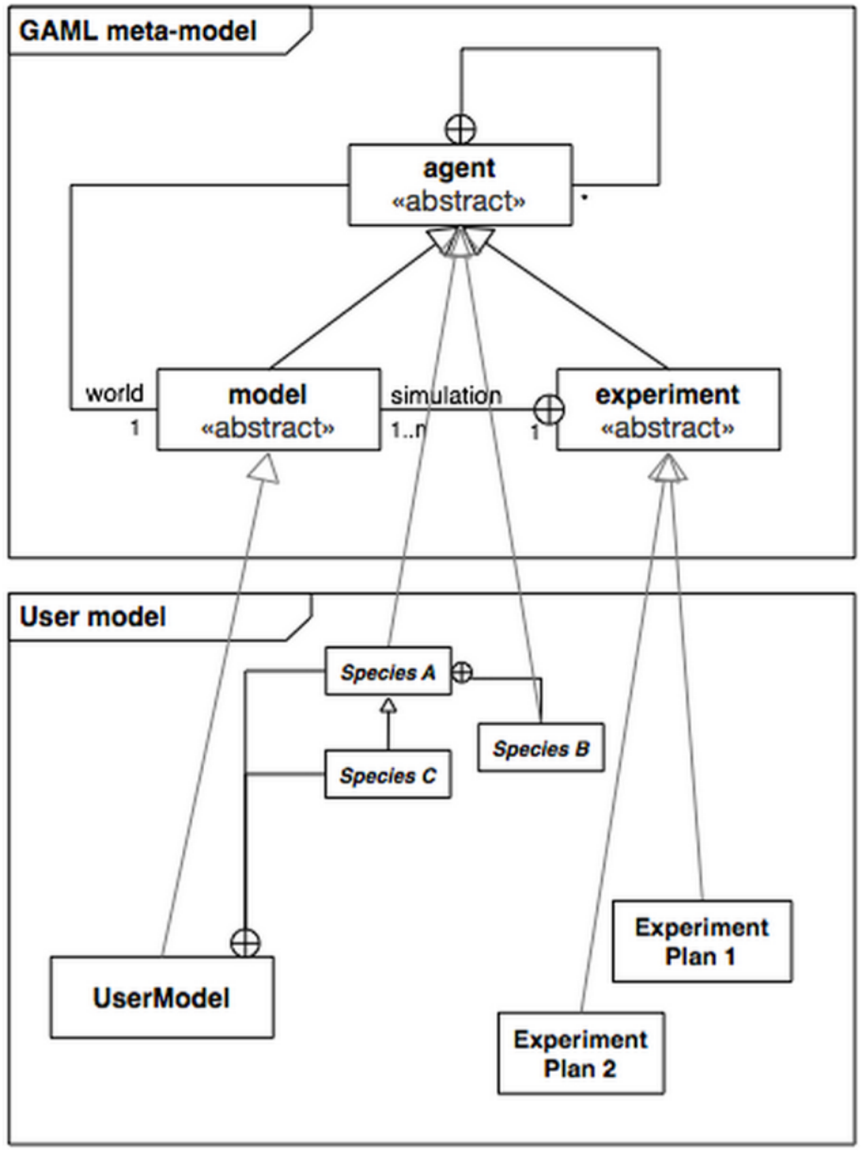
GAMA metamodel / a framework

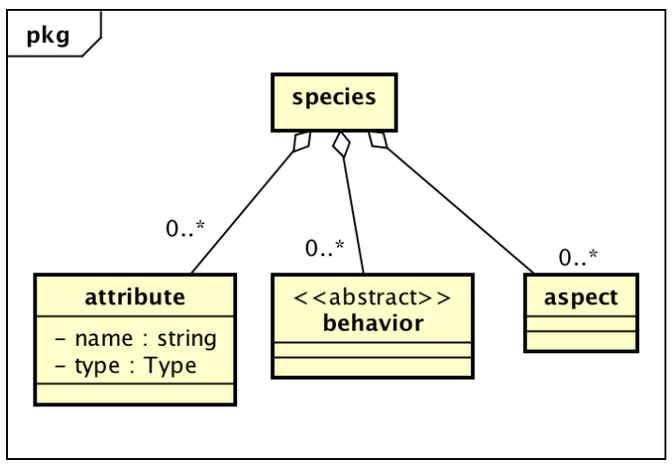
- In GAMA: everything is agent!



Instantiation of GAMA metamodel on a particular model

- **Model** (a.k.a. global): global variables, actions, dynamics environment and initialization.
- **Species** (and **Grid**): agent species. A species/grid is a UML class. Several species blocks can be defined.
- **Experiment** : simulation execution context, in particular inputs and outputs. Several experiment blocks can be defined.

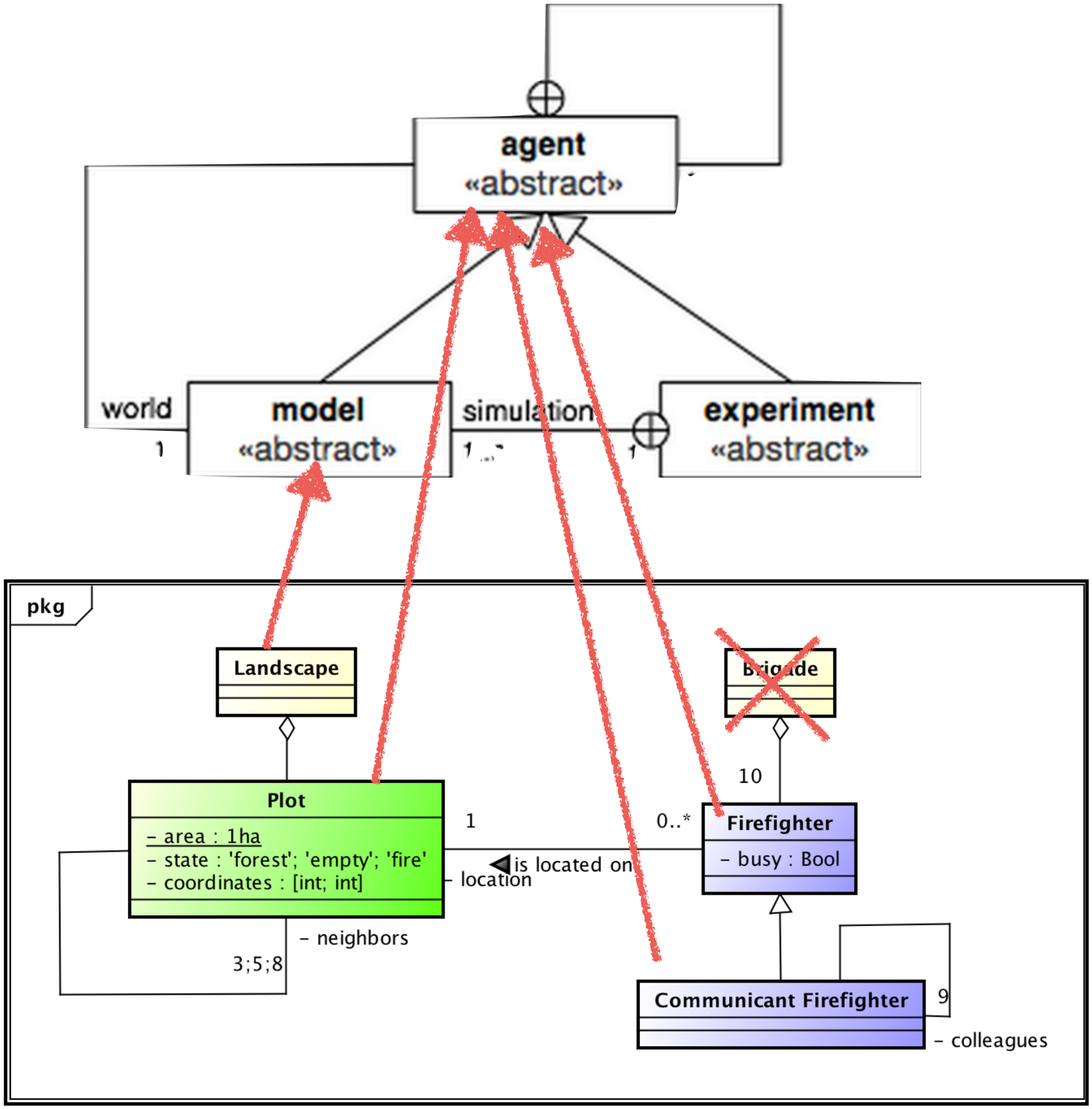




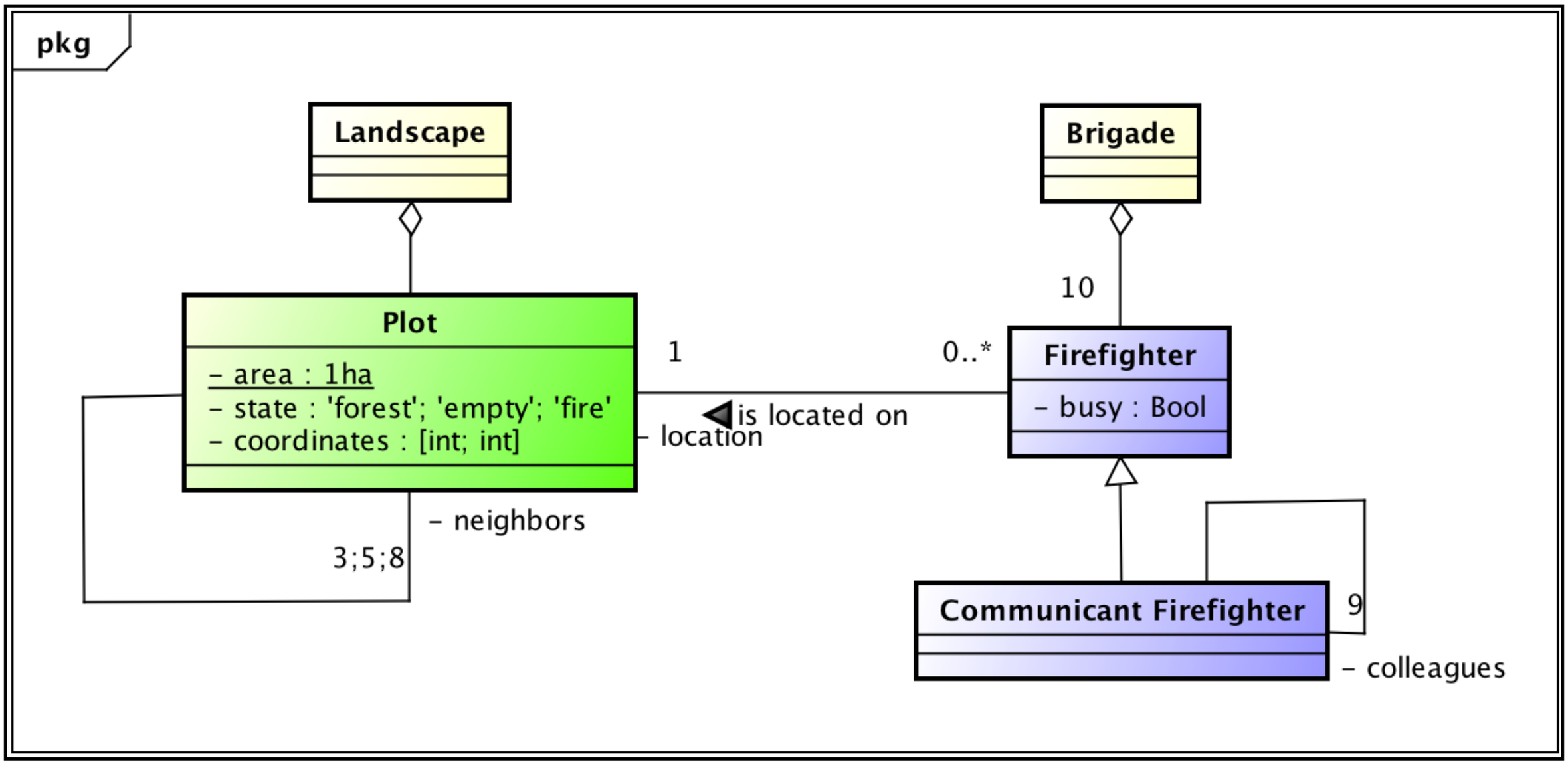
Implementation of the model

```
model firemen

global { }

grid plot {
    list<plot> neighbors;
    string state;

    rgb color;
}

species firefighter {
    bool busy;

    plot my_plot;
}

species communicant_firefighter parent:firefighter {
    list<communicant_firefighter> colleagues;
}
```

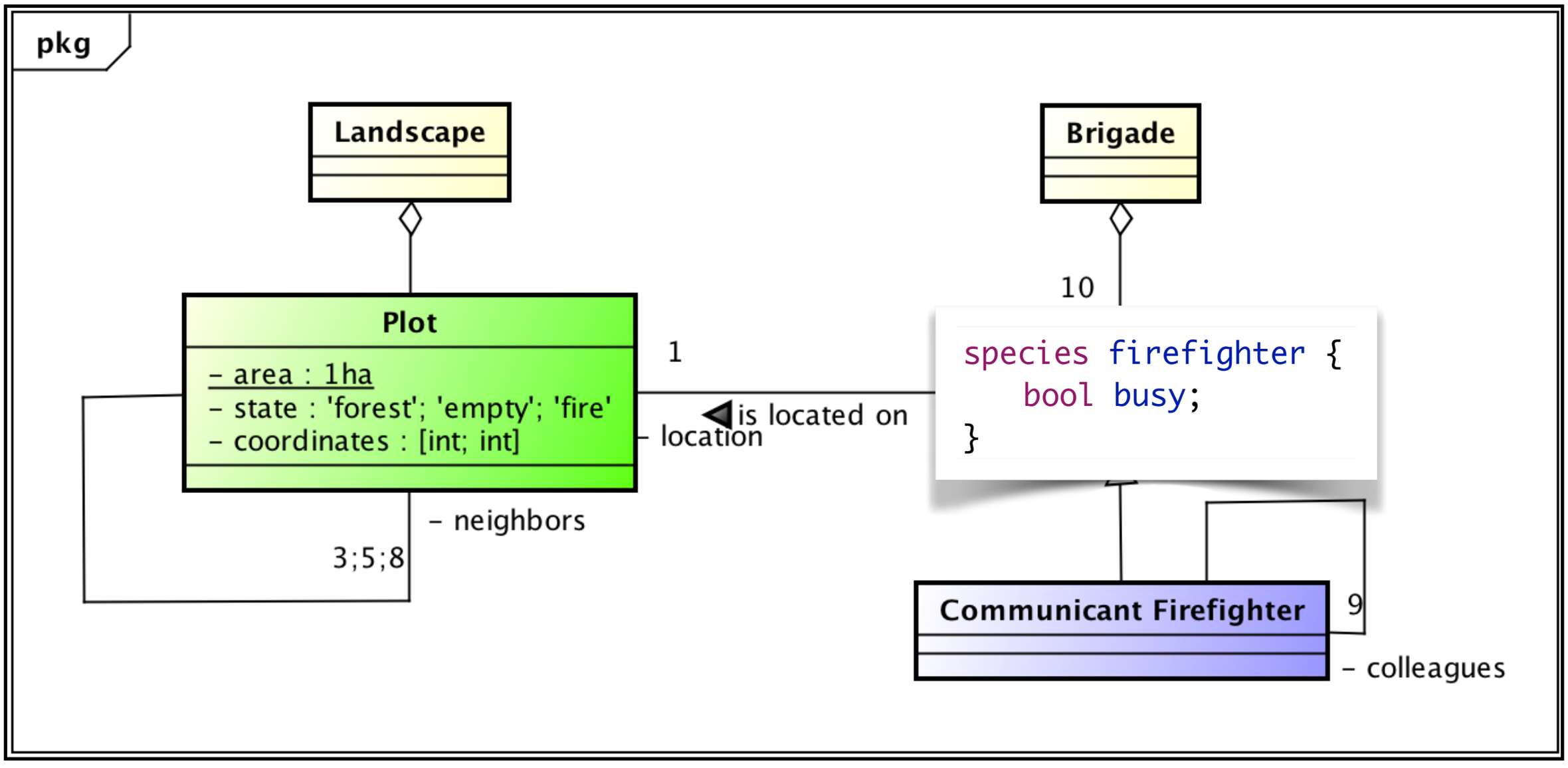
Structure: Mapping Firefighter model to GAMA Meta-model



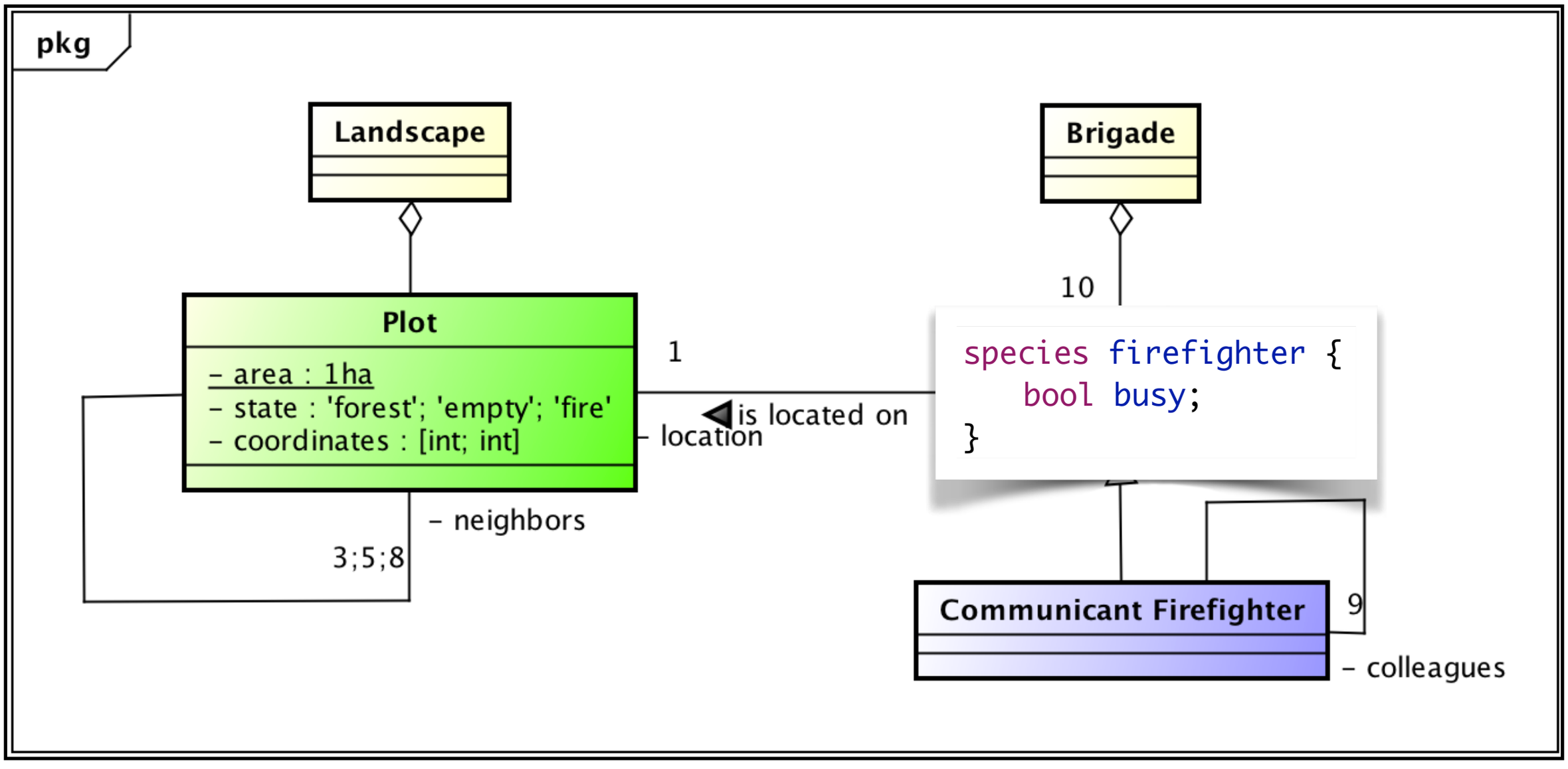
Structure: Mapping Firefighter model to GAMA Meta-model



Structure: Mapping Firefighter model to GAMA Meta-model



Structure: Mapping Firefighter model to GAMA Meta-model



- Create a first GAMA model with the model structure representing this UML class diagram.

Notes on the model.

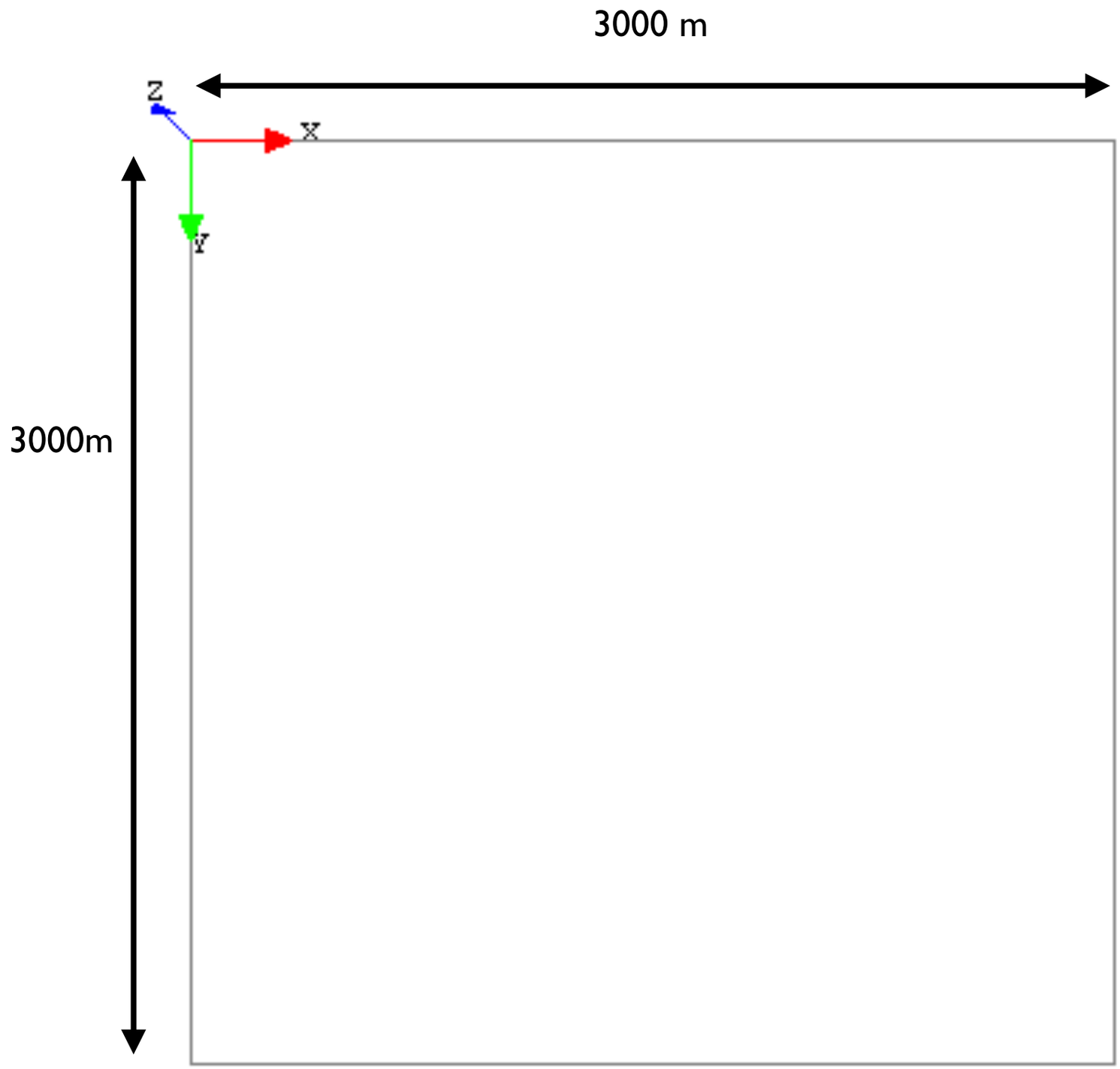
- Every kind of agent has **built-in attributes:**
  - name (a string)
  - shape (a geometry) (default value = a point)
  - location (a point) (value = the centroid of its shape)
- In addition, **grid** agents have additional built-in attributes:
  - grid_x (an integer)
  - grid_y (an integer)
  - color (a color)
  - grid_value (used when grid is created from a data file)

```
model firemen

global { }

grid plot {
    list<plot> neighbors;
    string state;

    rgb color;
}


species firefighter {
    bool busy;

    plot my_plot;
}

species communicant_firefighter parent:firefighter {
    list<communicant_firefighter> colleagues;
}
```

Space in GAMA

- In GAMA, agents have a **location** in a **reference continuous space.**

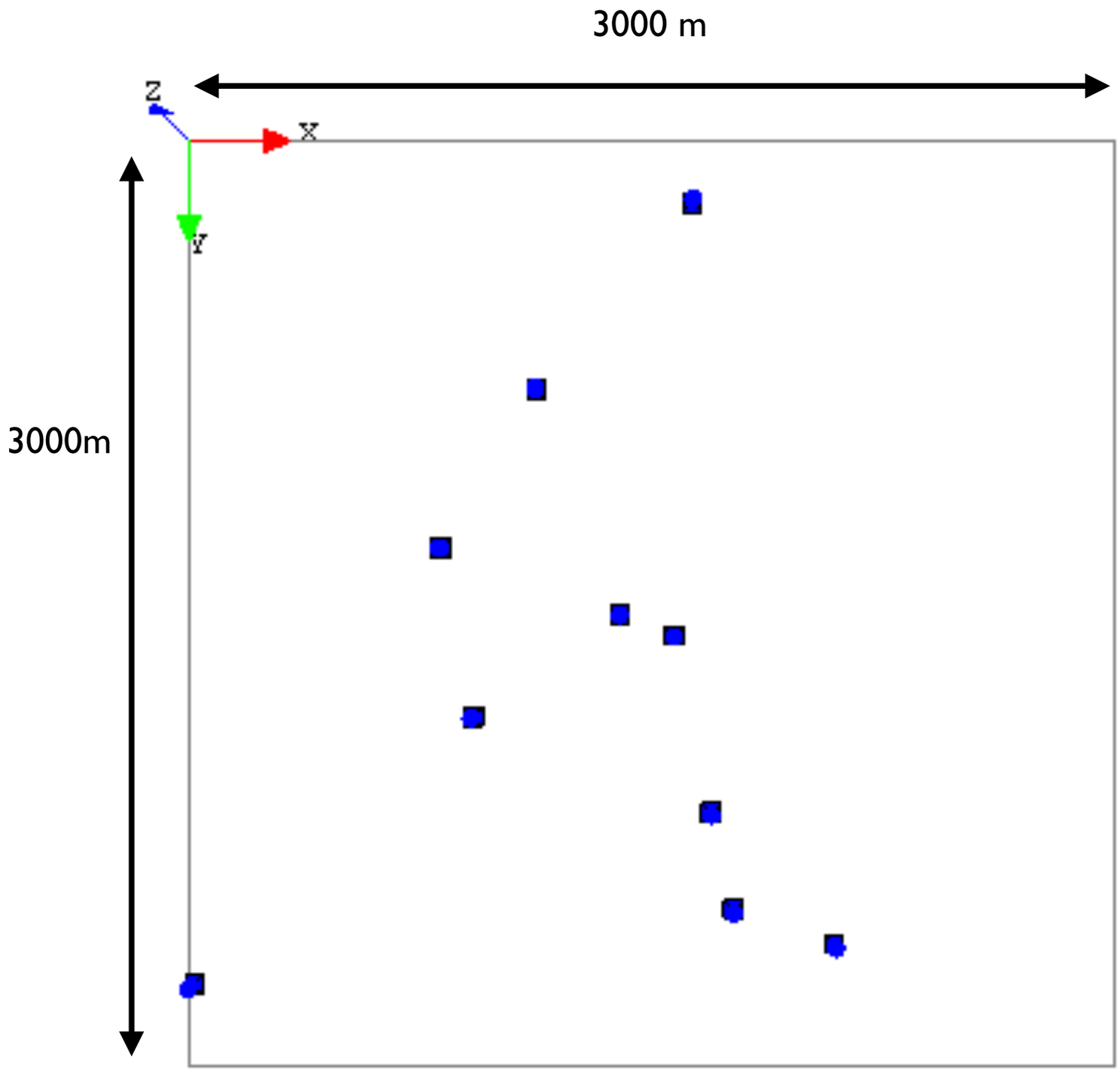
- To **create a grid of cells**, we need to create explicitly a new species with a particular **spatial organisation** (a particular topology).



Space in GAMA

- In GAMA, agents have a **location** in a **reference continuous space.**

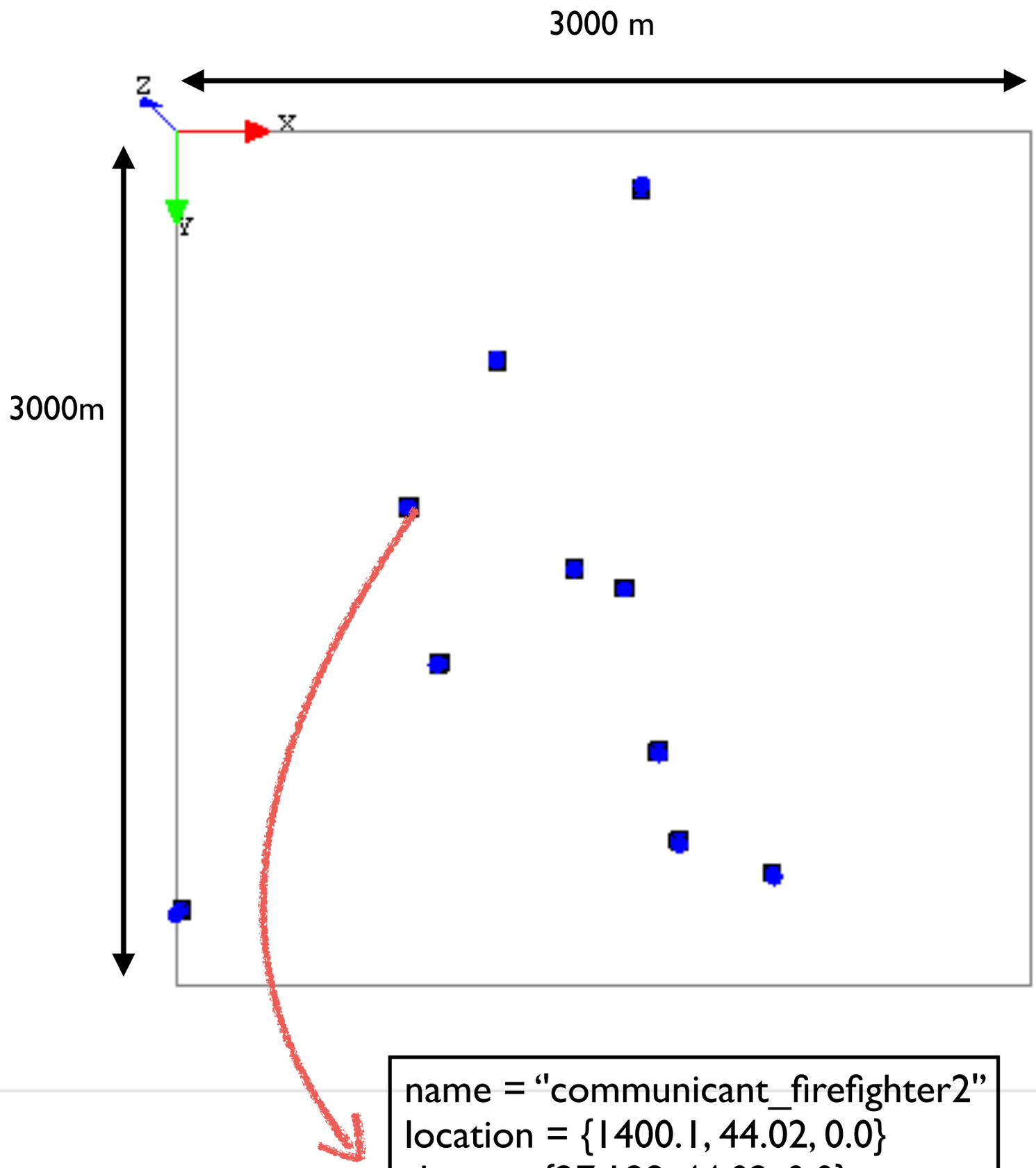
- To **create a grid of cells**, we need to create explicitly a new species with a particular **spatial organisation** (a particular topology).



Space in GAMA

- In GAMA, agents have a **location** in a **reference continuous space.**

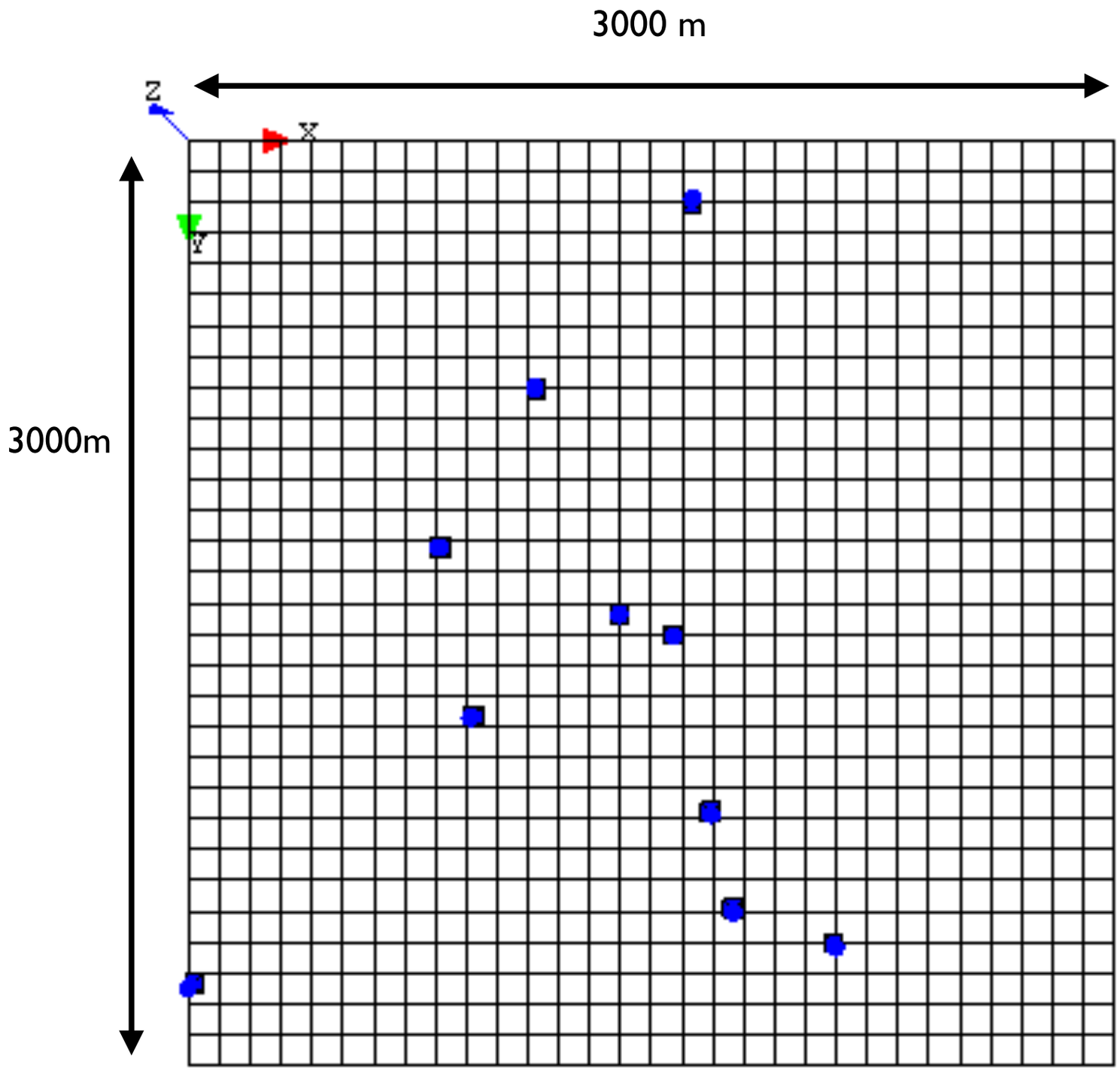
- To **create a grid of cells**, we need to create explicitly a new species with a particular **spatial organisation** (a particular topology).



Space in GAMA

- In GAMA, agents have a **location** in a **reference continuous space.**

- To **create a grid of cells**, we need to create explicitly a new species with a particular **spatial organisation** (a particular topology).

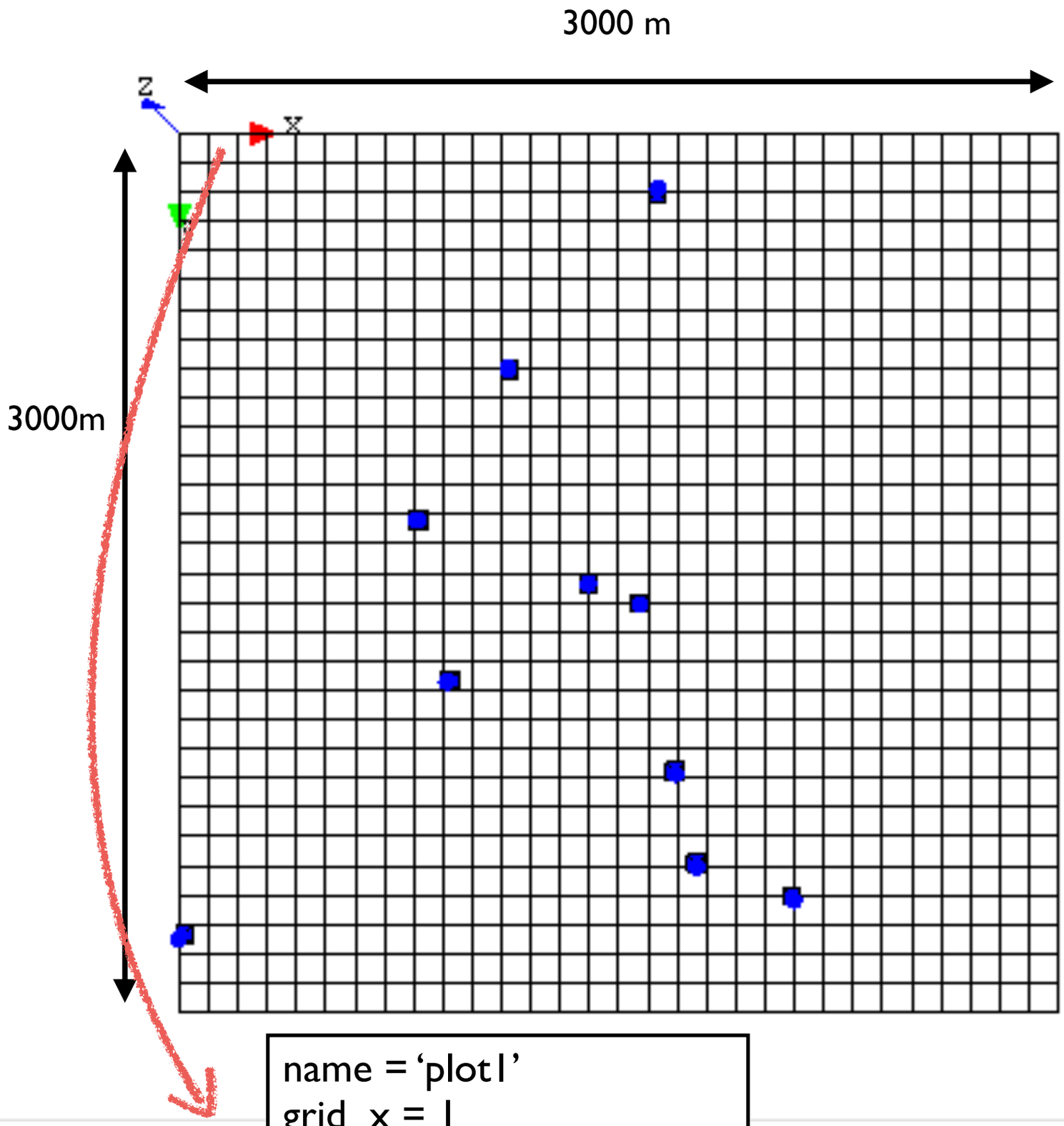


Addition of a 30x30 grid

Space in GAMA

- In GAMA, agents have a **location** in a **reference continuous space.**

- To **create a grid of cells**, we need to create explicitly a new species with a particular **spatial organisation** (a particular topology).



Addition of a 30x30 grid

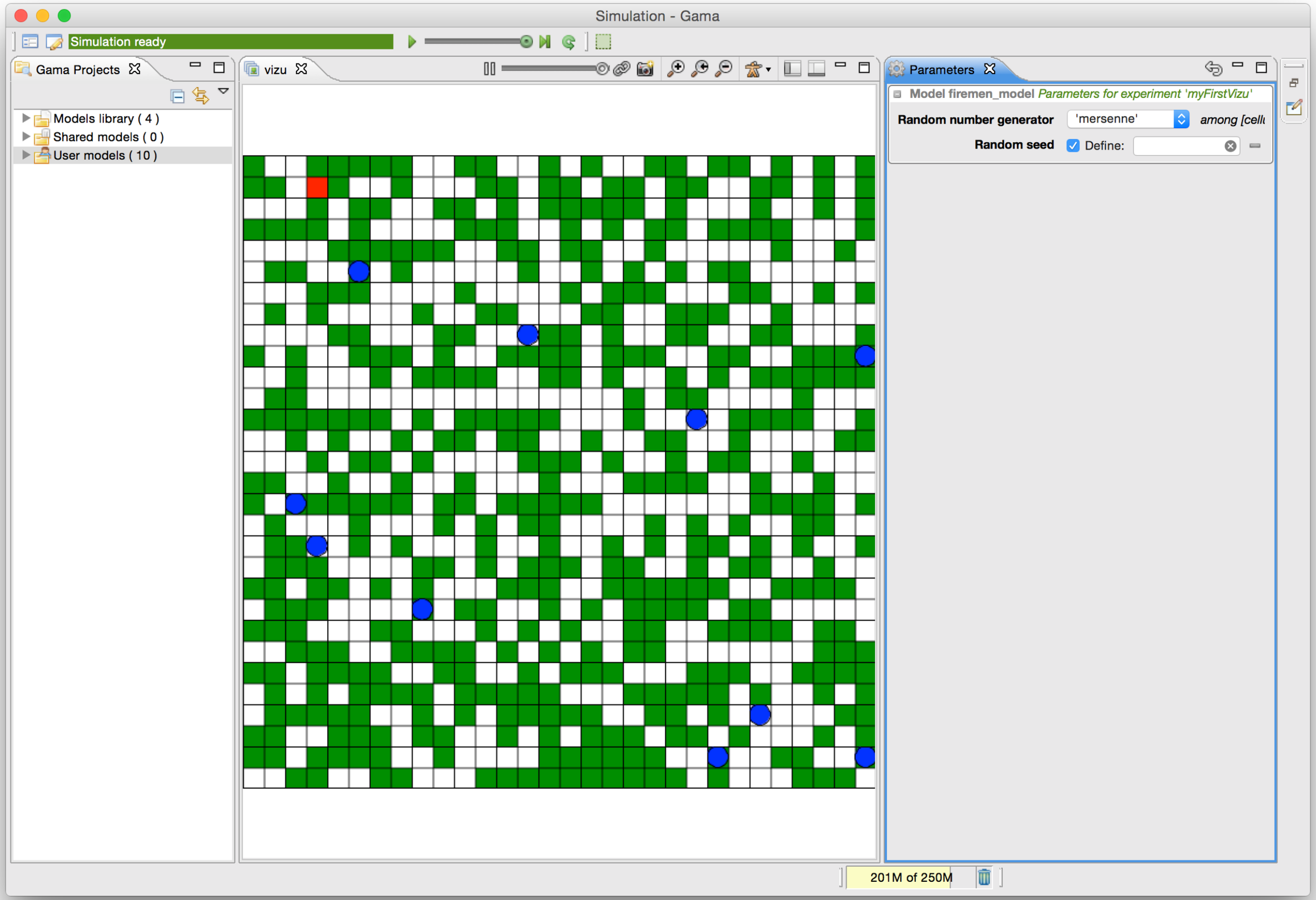
Firefighter model - Initialization

- Initialization of the global:
  - Add a global variable to choose if we will play scenario 1 or 2.
  - creation of agents
  - Initialization of the environment size

- Initialization of plot agents:
  - create 900 plots of 1ha, i.e. 30x30 plots
  - Setting randomly 50% patches to forest / green and 50% patches to clear / white
- Initialization of firemen
  - create 10 firemen randomly located, depending of the chosen scenario

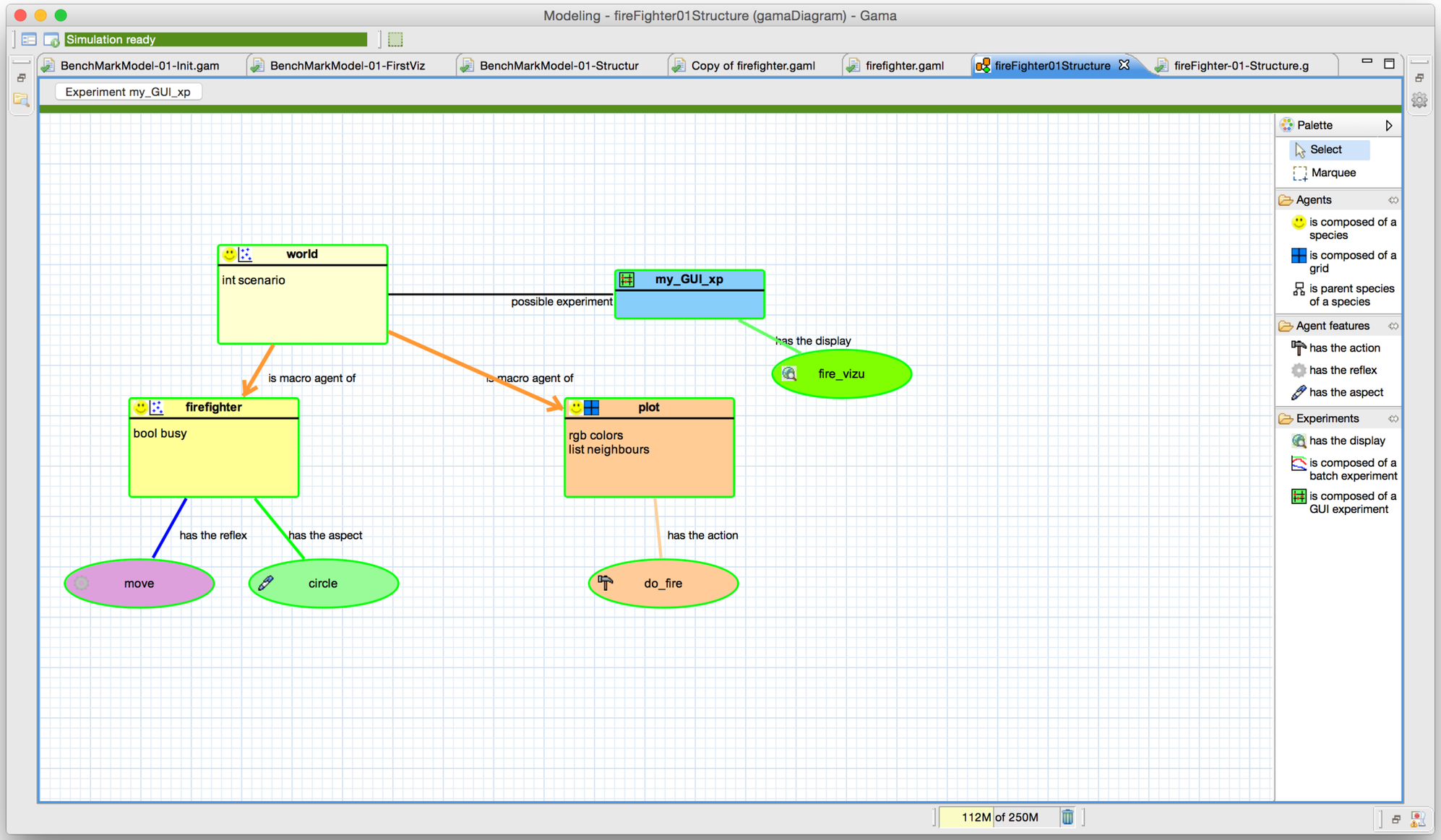
- Setting fire:
  - chose 1 plot

Firefighter model - Initialization

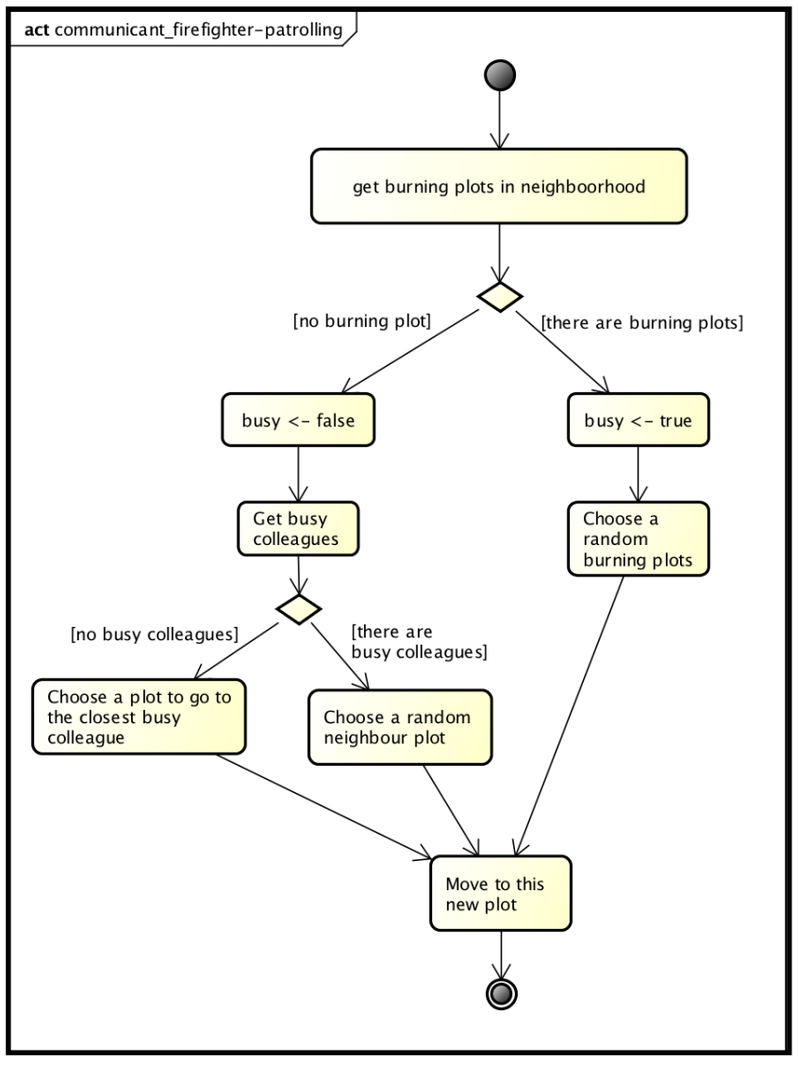


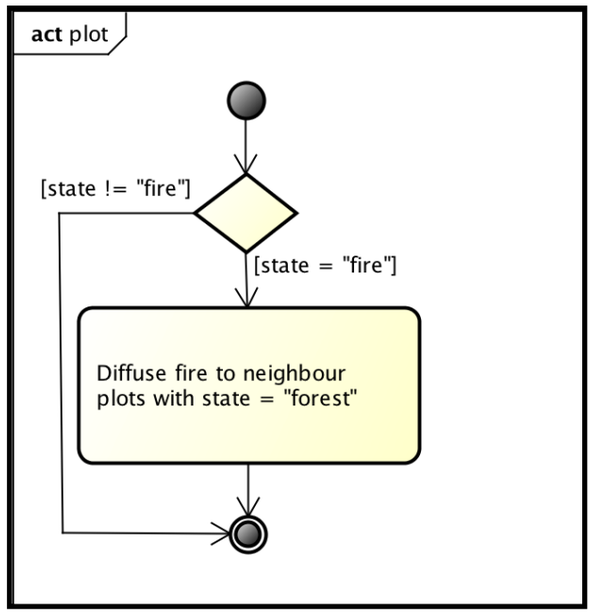
Do it with the Graphical Modeling plugin (still a beta version).

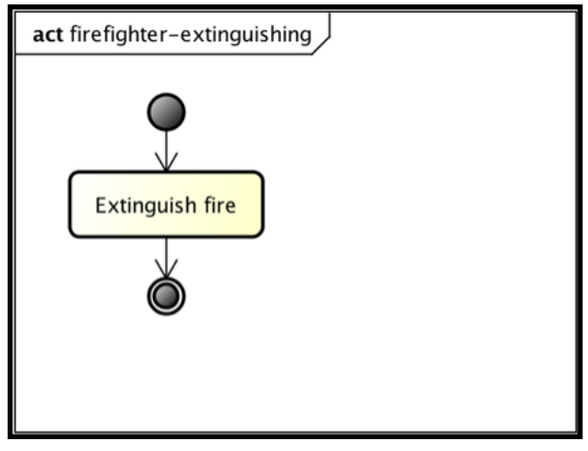
- GAMA: install extension Graphical Modelling
- Create new diagram
- Generate GAML model

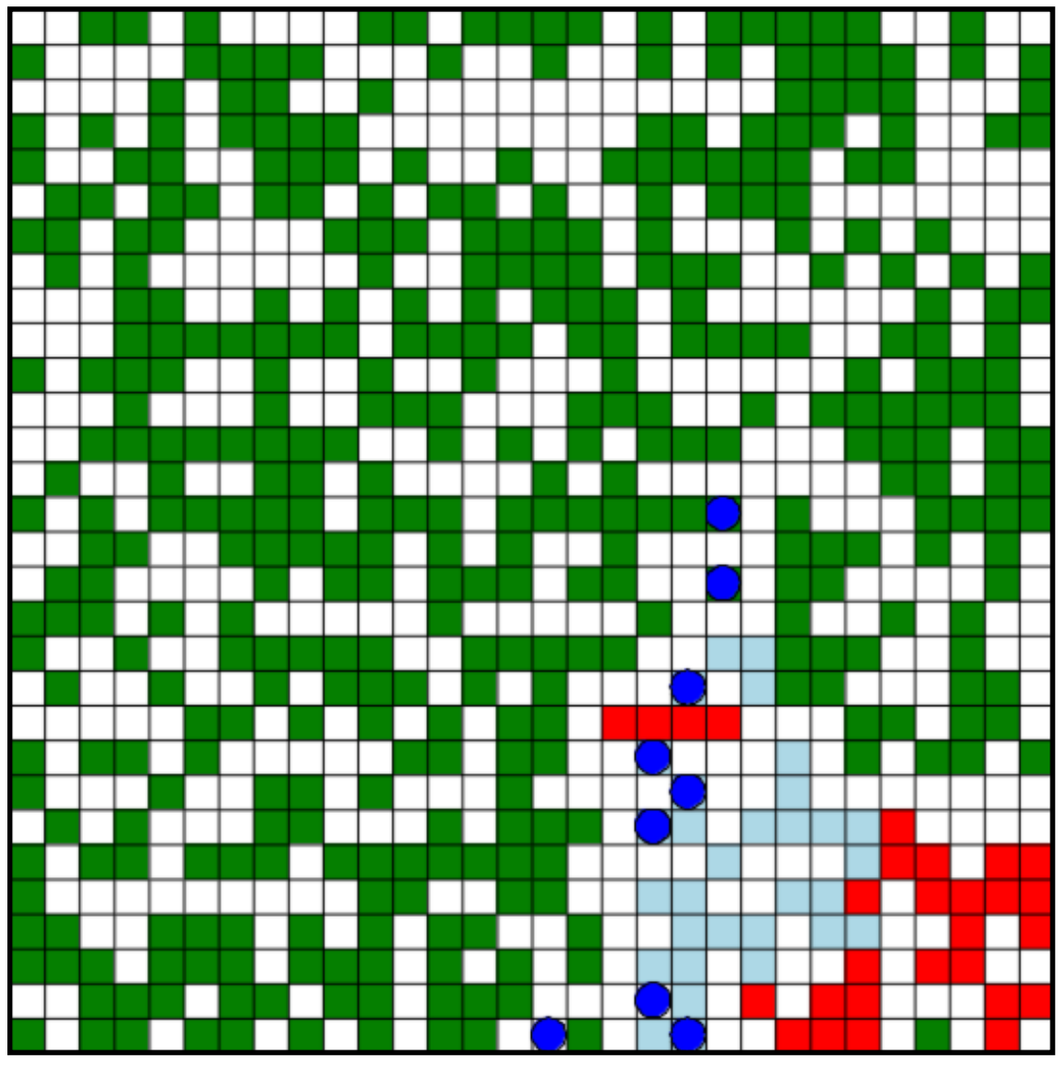


Firefighter model : Dynamics



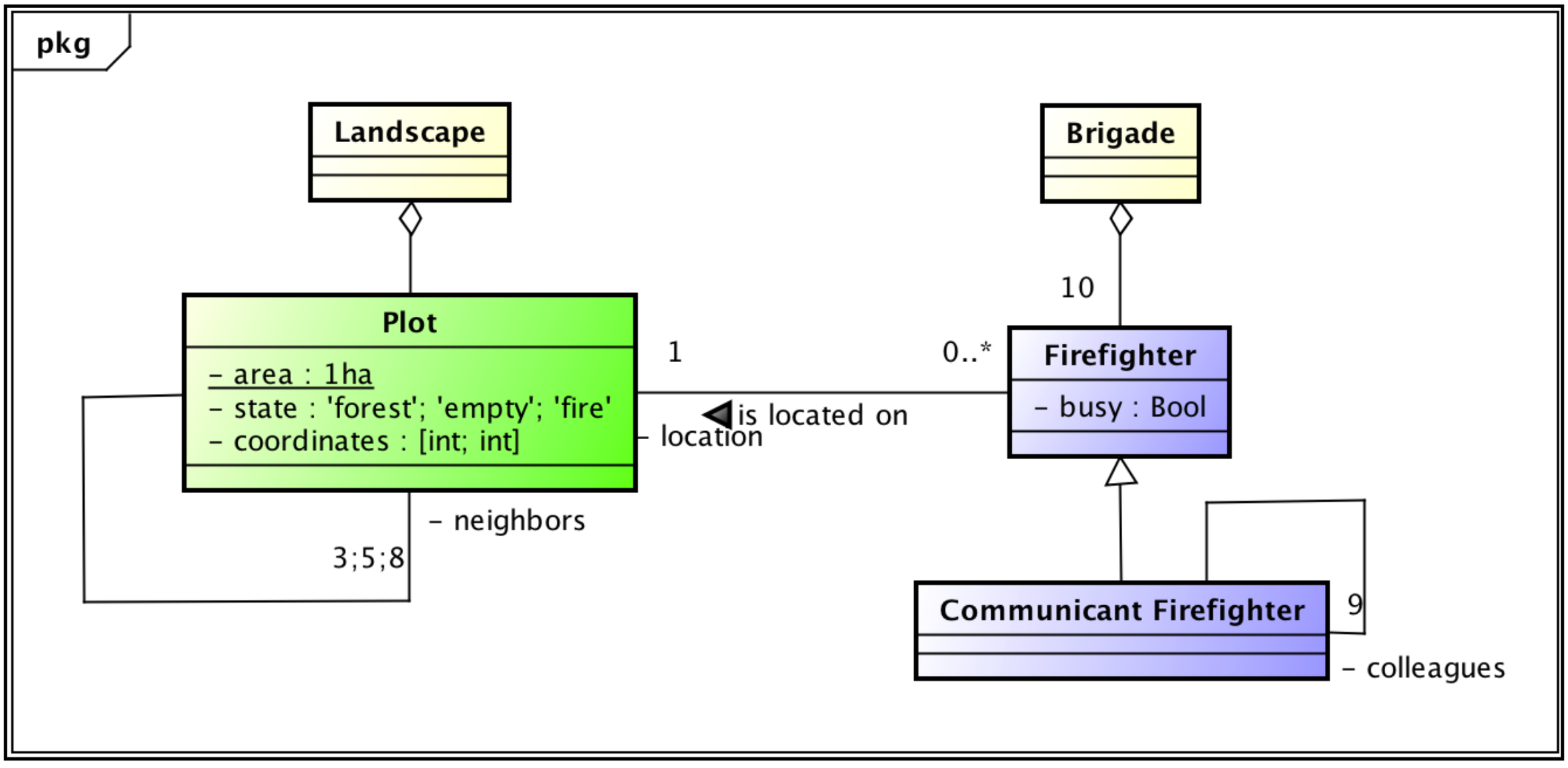




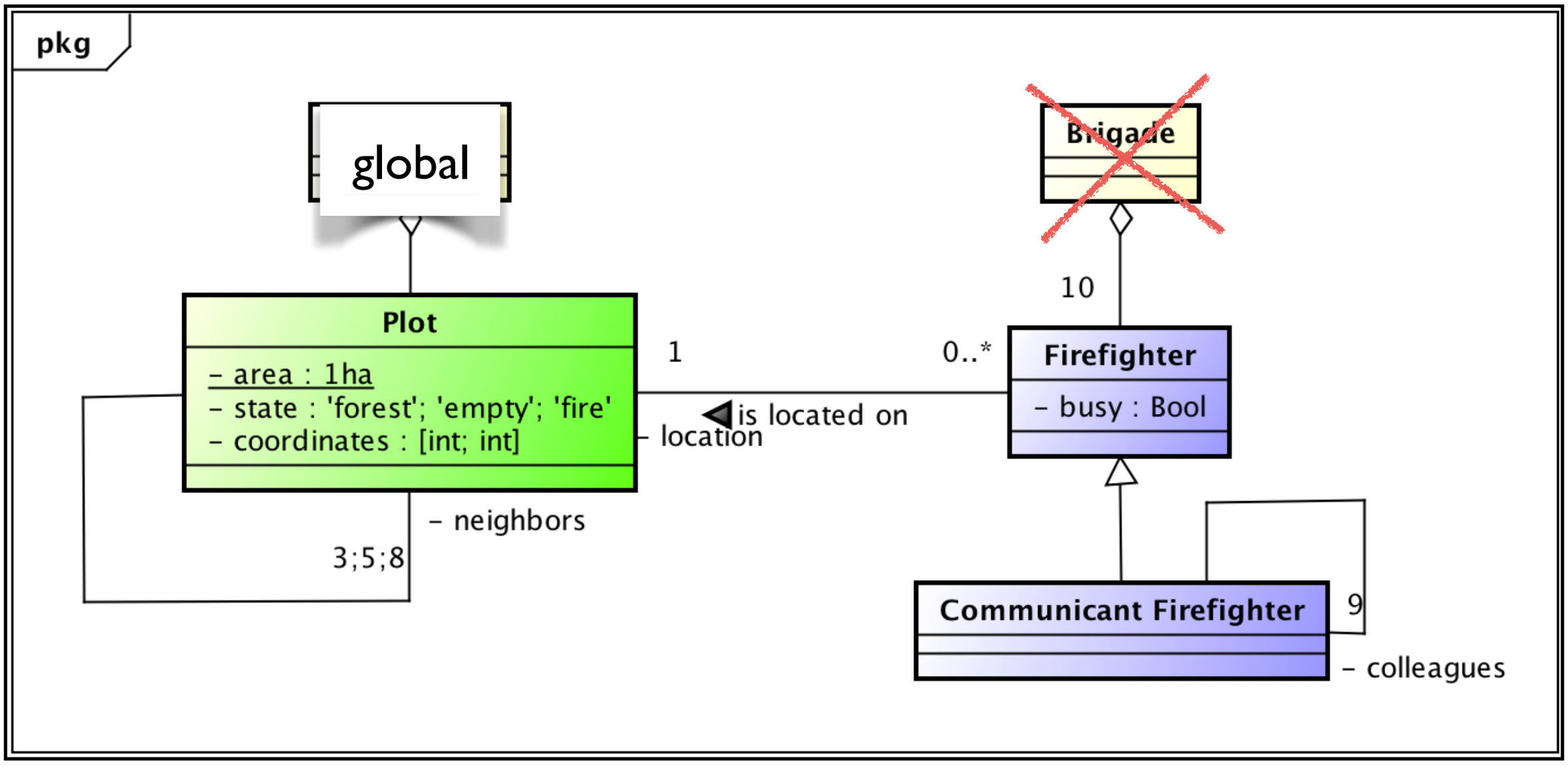


Benoit Gaudou (Univ. Toulouse 1)
Patrick Taillandier (INRAE)

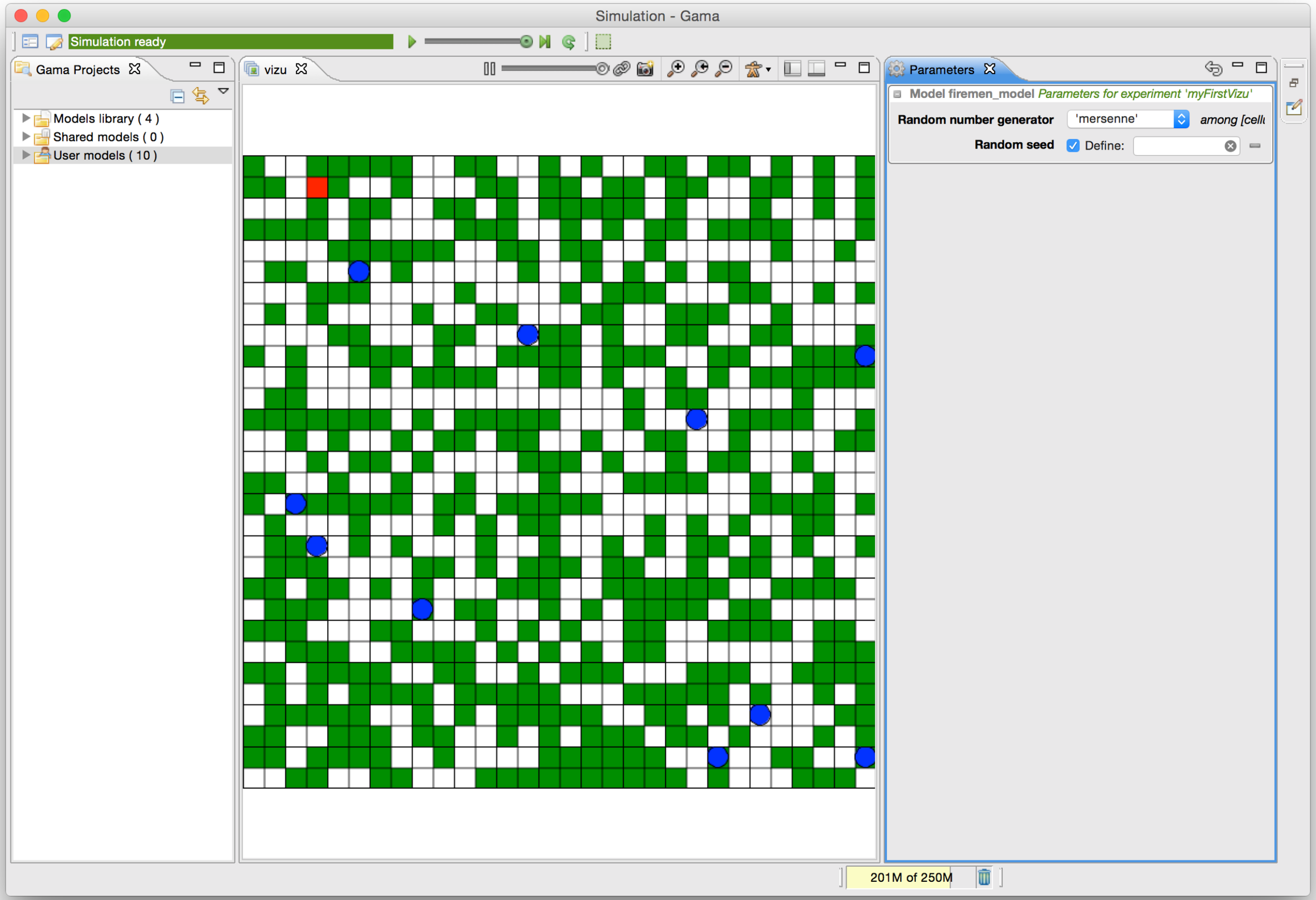
Reminder


pkg
Landscape
Brigade
10
Plot
- area : 1ha
- state : 'forest'; 'empty'; 'fire'
- coordinates : [int; int]
1
0..*
Firefighter
- busy : Bool
is located on
- location
- neighbors
3;5;8
Communicant Firefighter
9
- colleagues

Reminder


pkg
global
Plot
- area : 1ha
- state : 'forest'; 'empty'; 'fire'
- coordinates : [int; int]
1
- location
is located on
0..*
10
Brigade
Firefighter
- busy : Bool
- neighbors
3;5;8
Communicant Firefighter
9
- colleagues

Firefighter model - Result of initialization

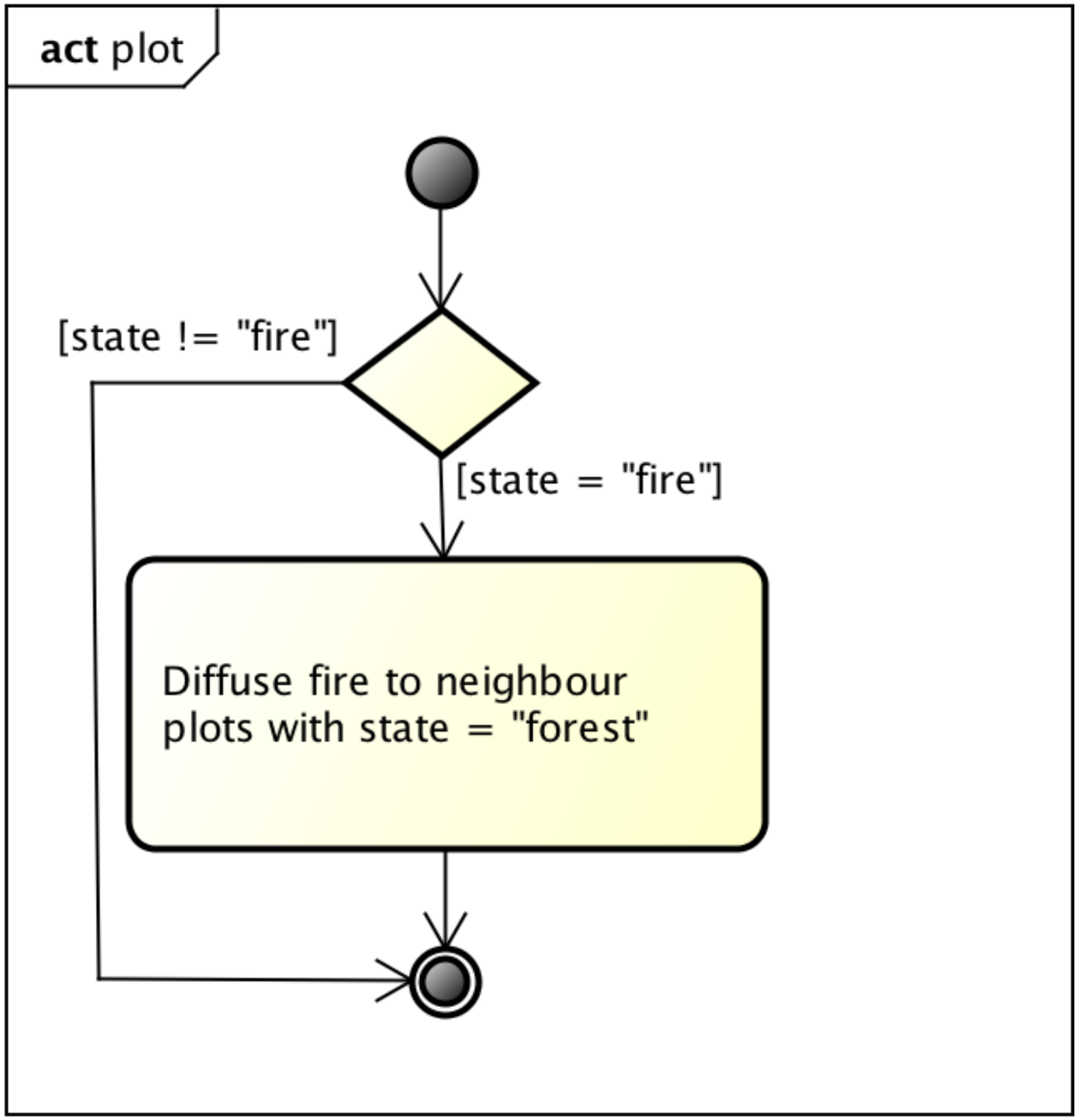


Scheduling: at each simulation step, GAMA executes agents in the following order.

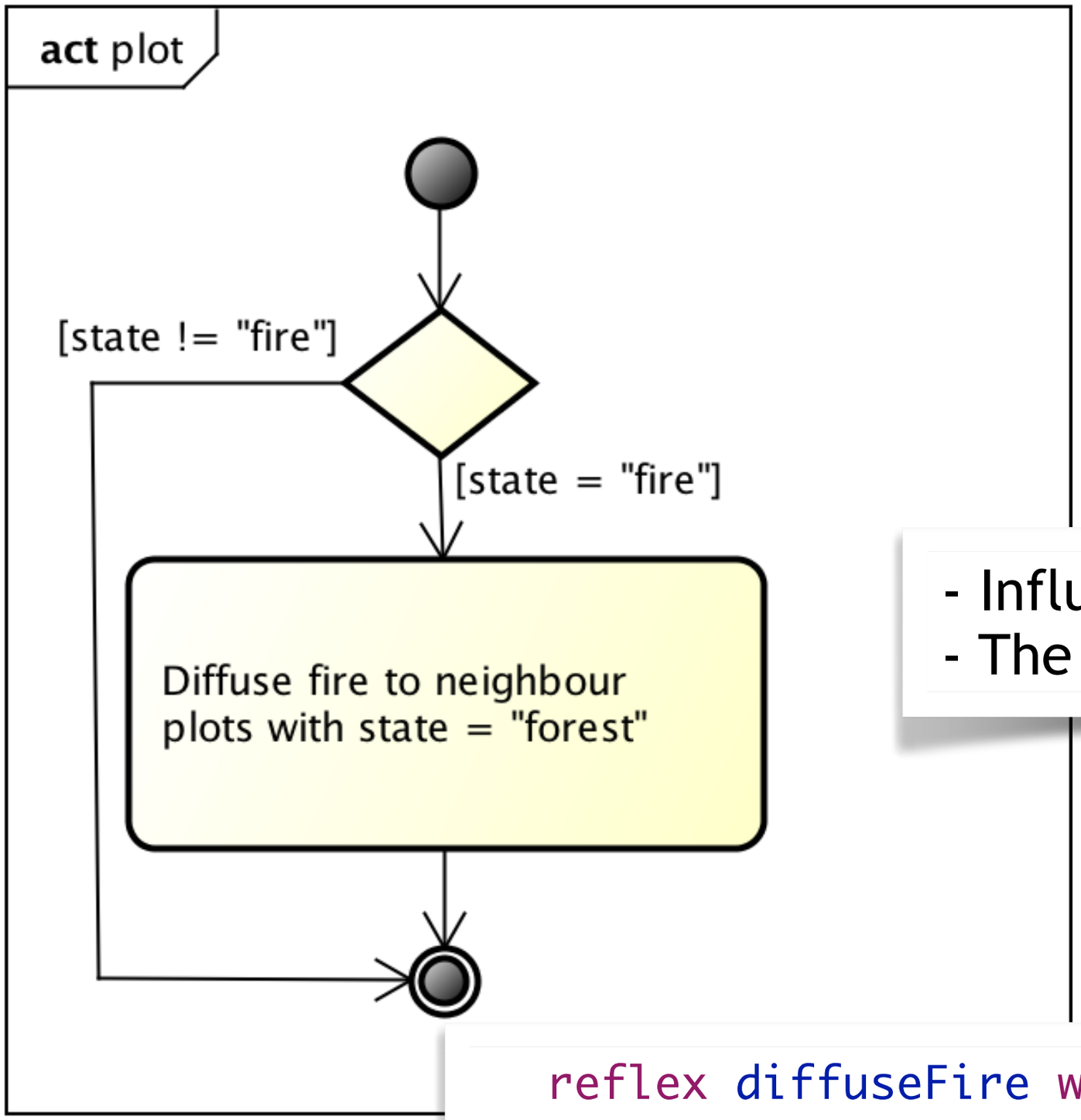
```
For the world agent,
 its behaviors (e.g. its reflexes, in the
   order)

For each species s Do
 For each agent a of s Do
 its behaviors (e.g. its reflexes, in the
   order)

For each agent of grid Do
 its behaviors (e.g. its reflexes, in the
   order)
```

The scheduler can be modified at hand

Dynamics of plots



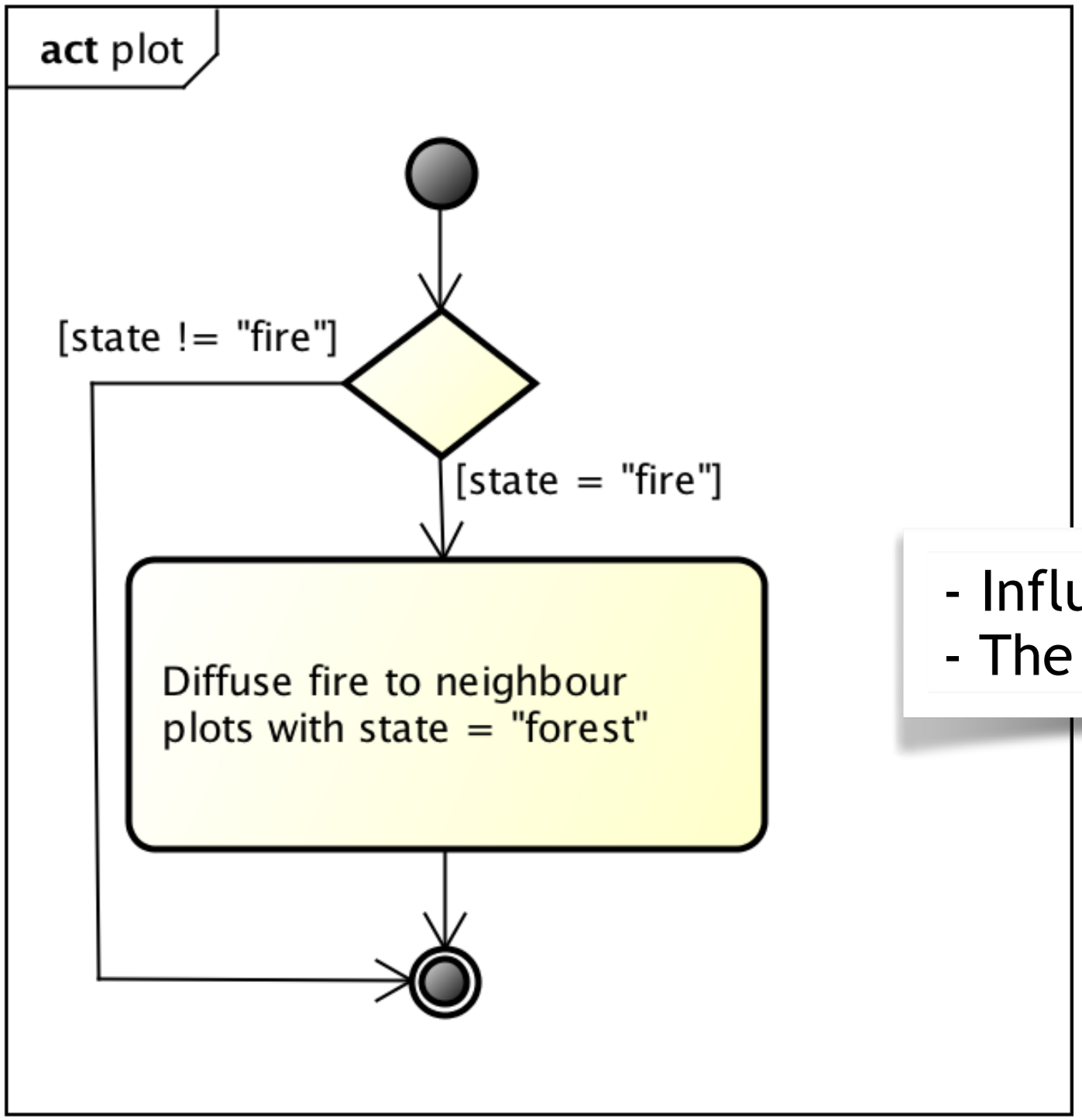
Dynamics of plots



- Influence of the number of neighbors
- The diffusion is too fast!

```
reflex diffuseFire when: (state = "fire") {
   ask (neighbors where (each.state = "forest")) {
      state <- "fire" ;
      color <- #red ;
   }
}
```

Dynamics of plots



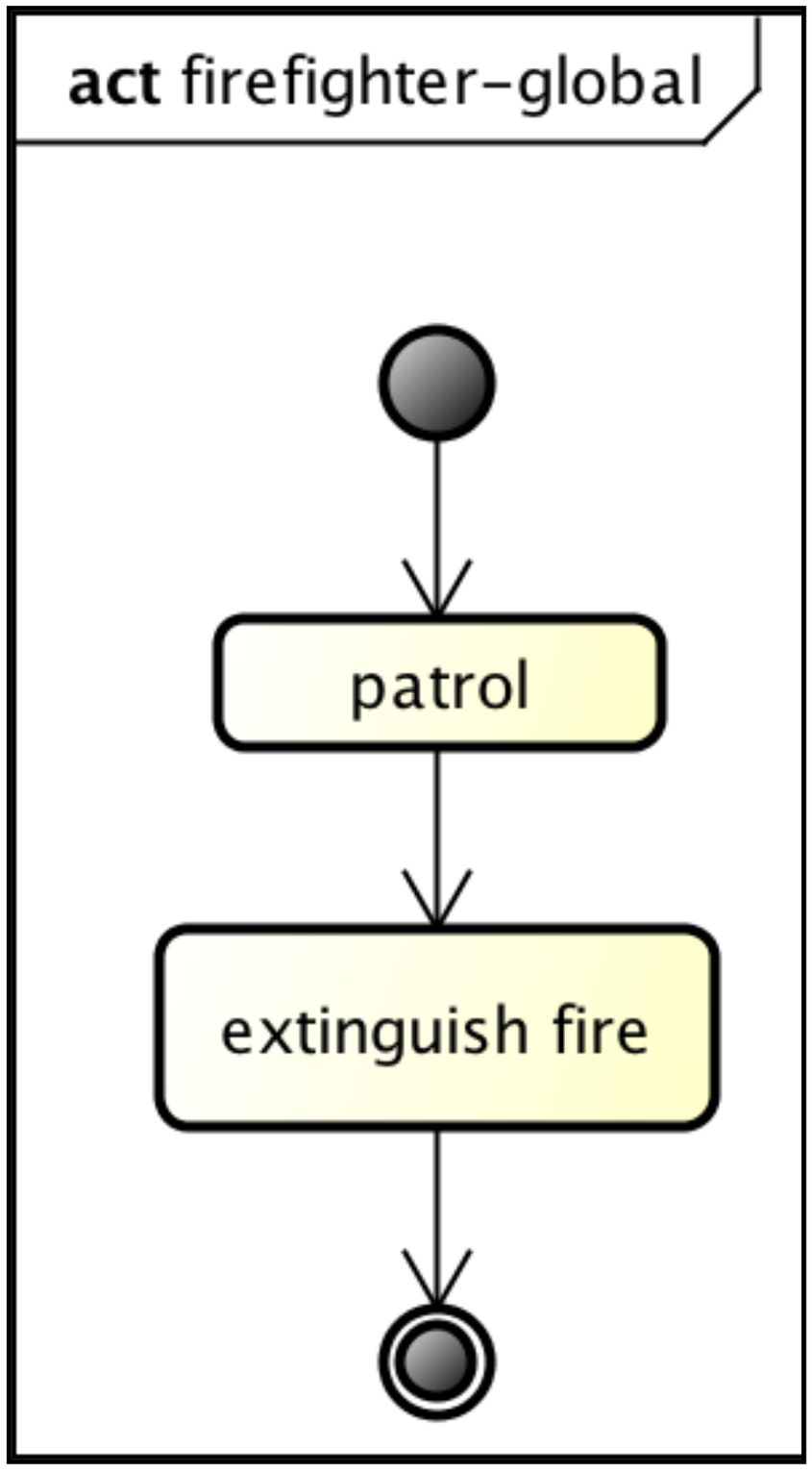
- Influence of the number of neighbors
- The diffusion is too fast!

-> Change the scheduling of the plots (schedules facet) to choose at the beginning of the step only plots which are on fire.

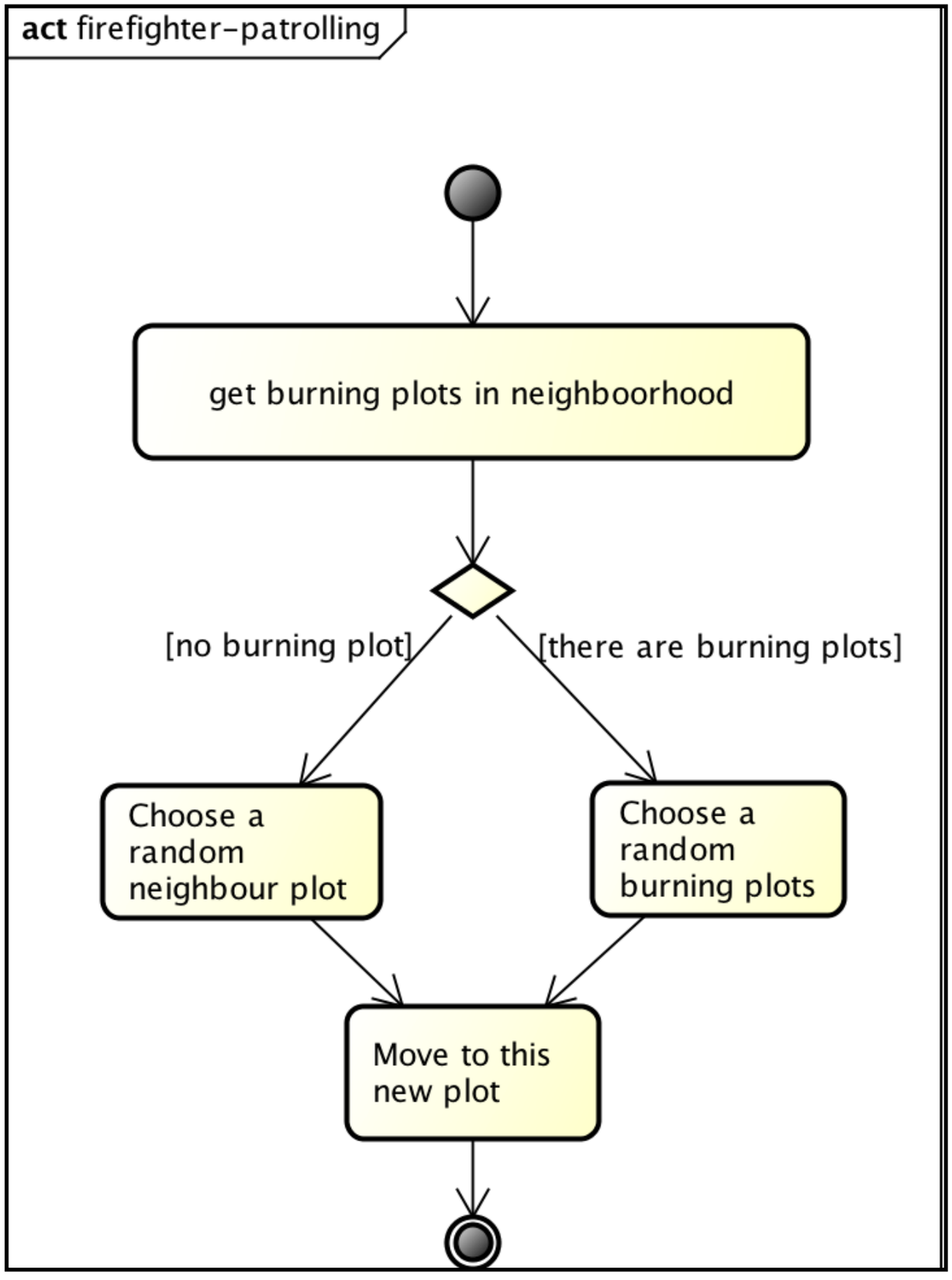
Dynamics of firefighters

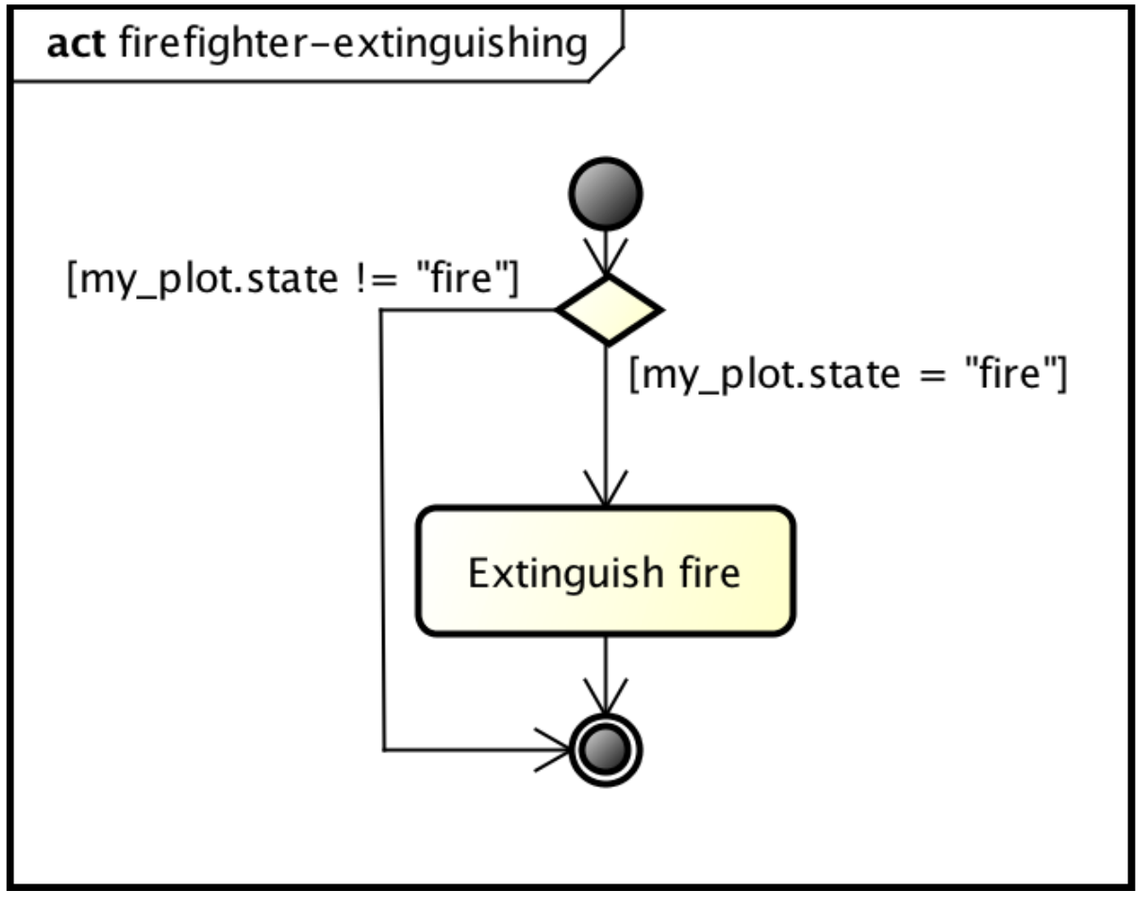
- **Implement the following diagram of the firefighters' behaviours.**

- You can implement patrol and extinguish fire as 2 reflexes (see next slide).

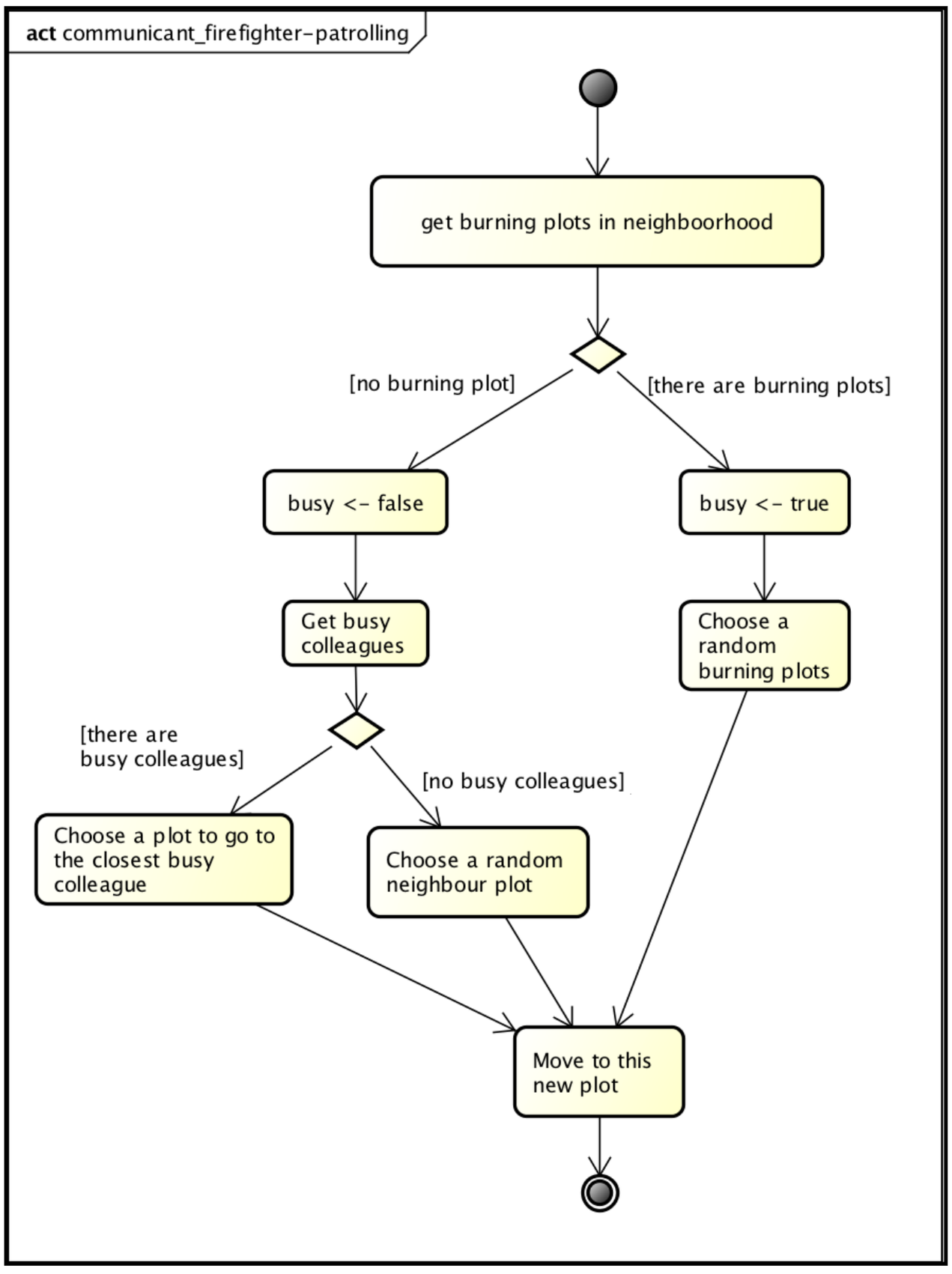


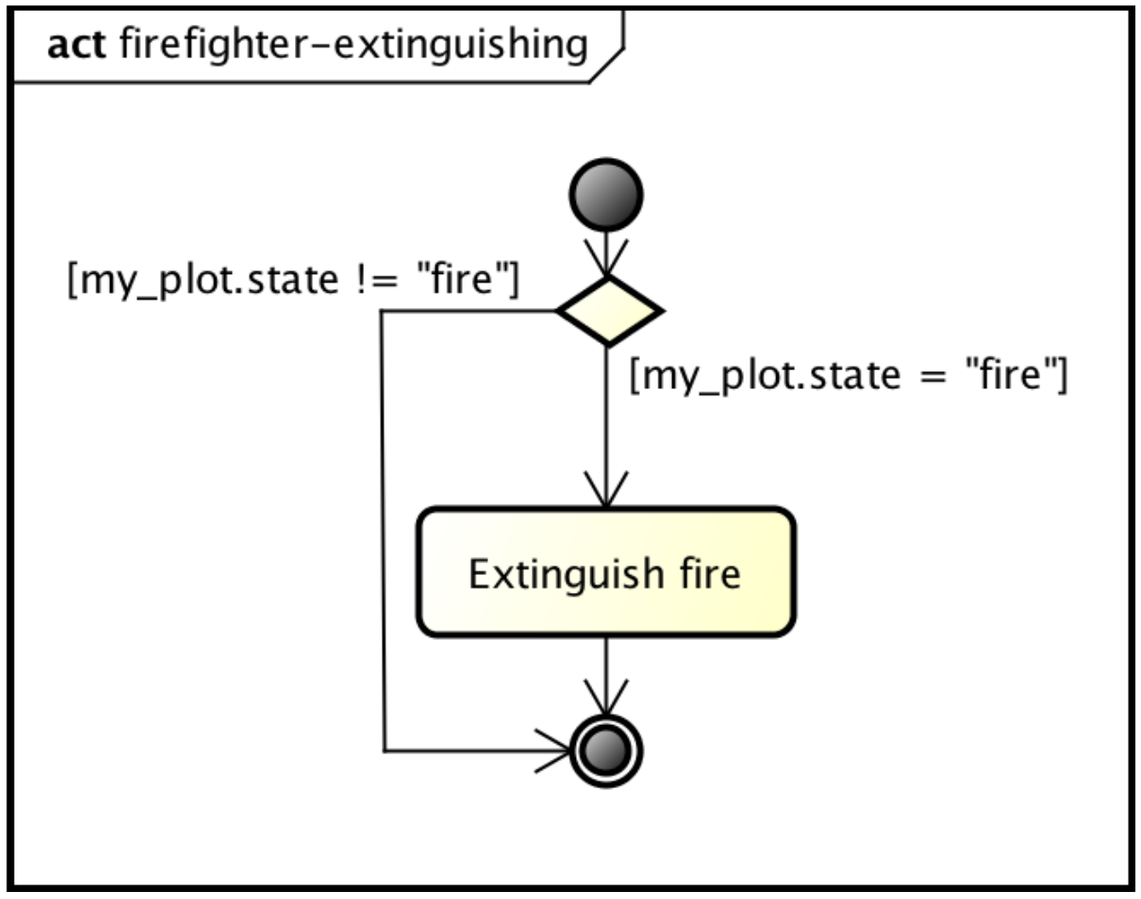
Dynamics of firefighters



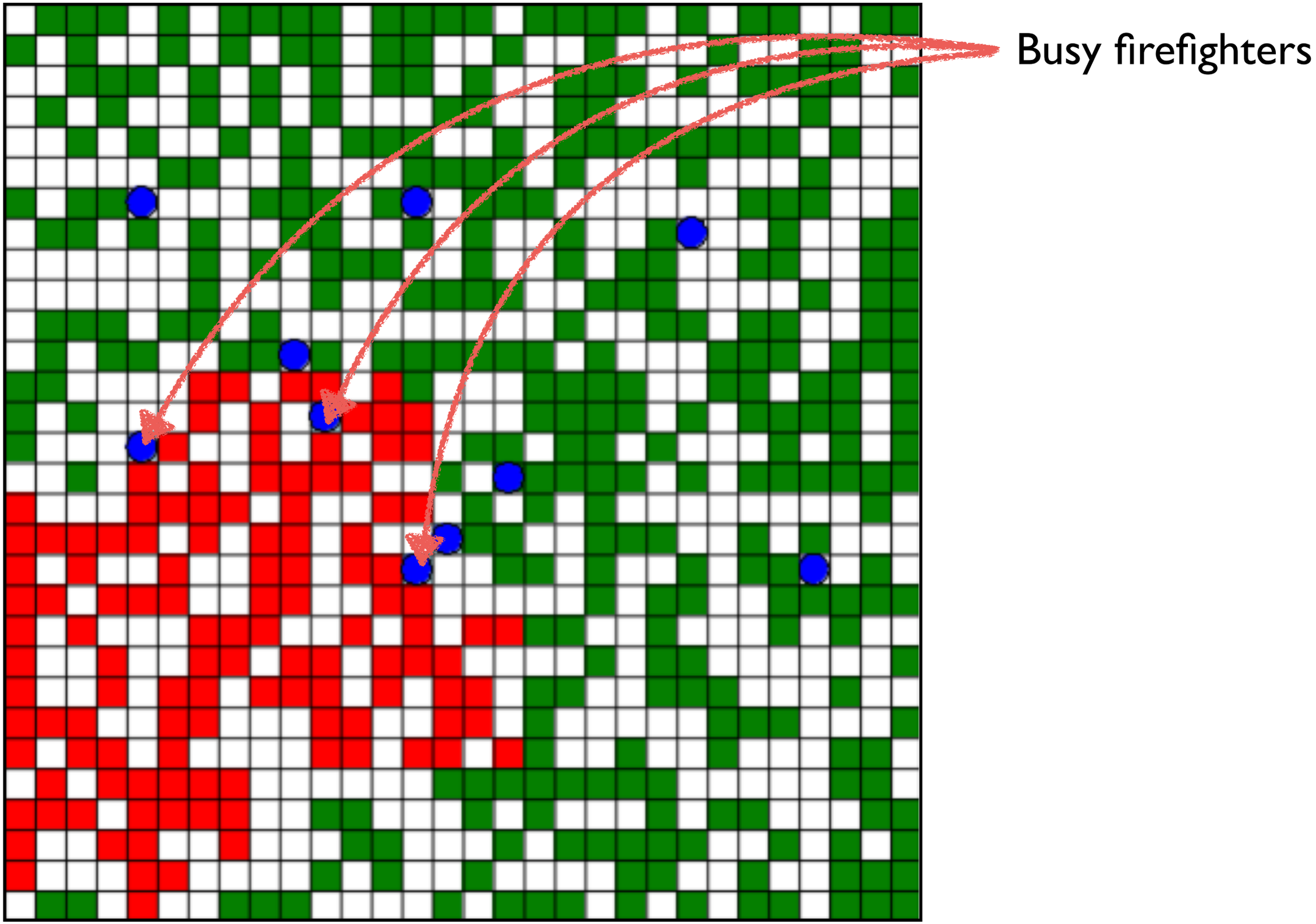


Dynamics of communicant firefighters

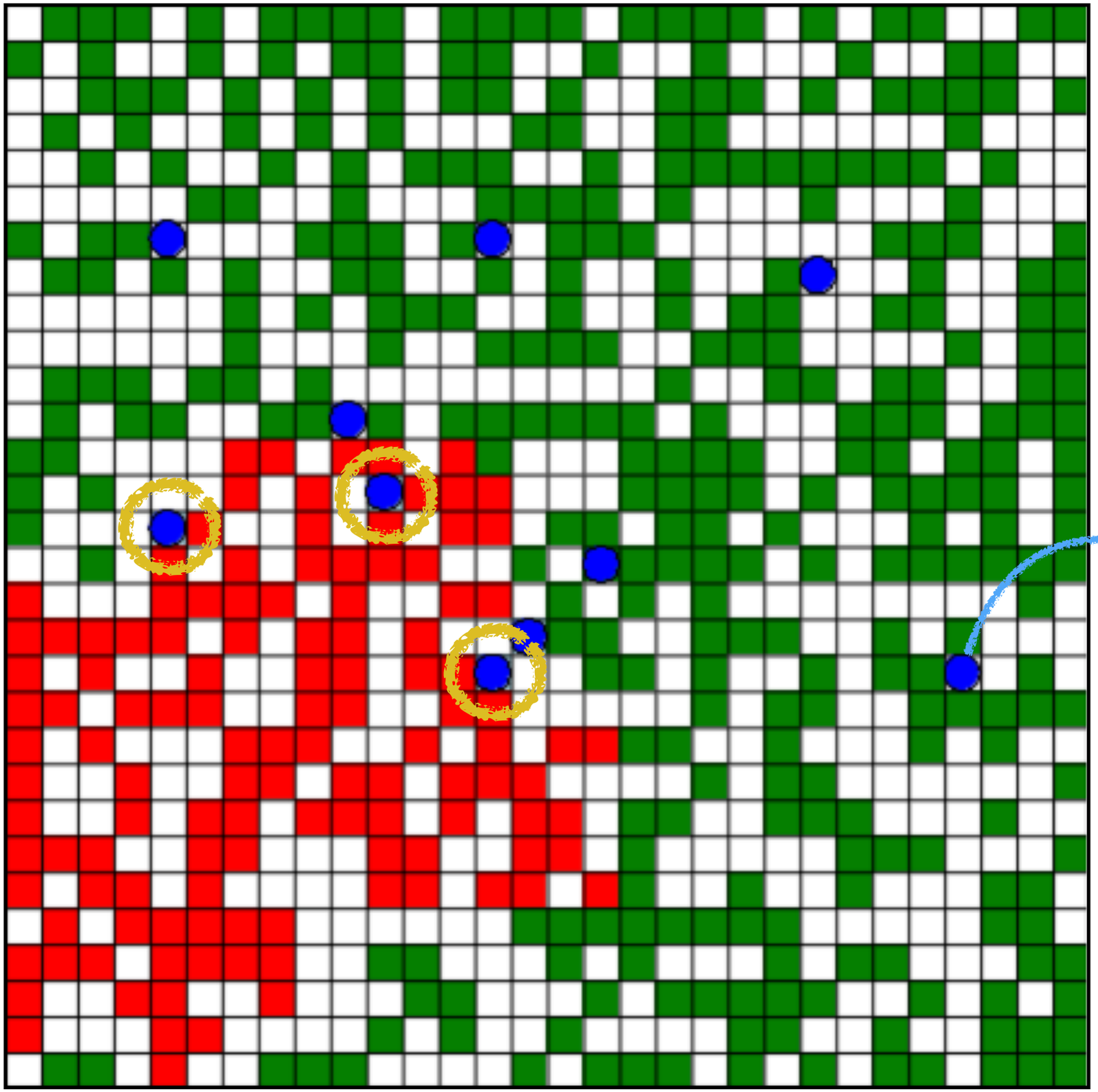




Choose a plot toward the closest busy firefighter

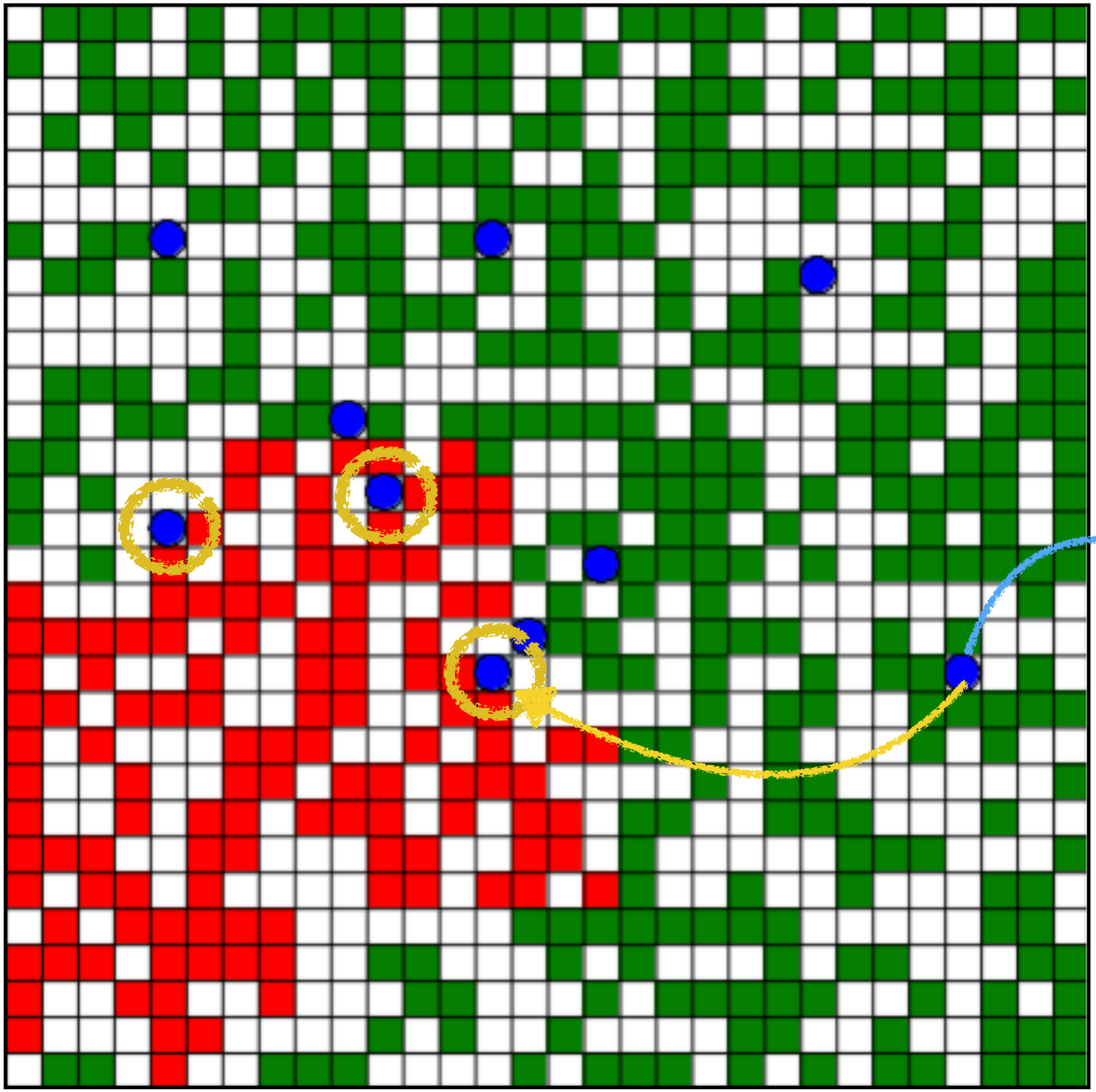


Choose a plot toward the closest busy firefighter



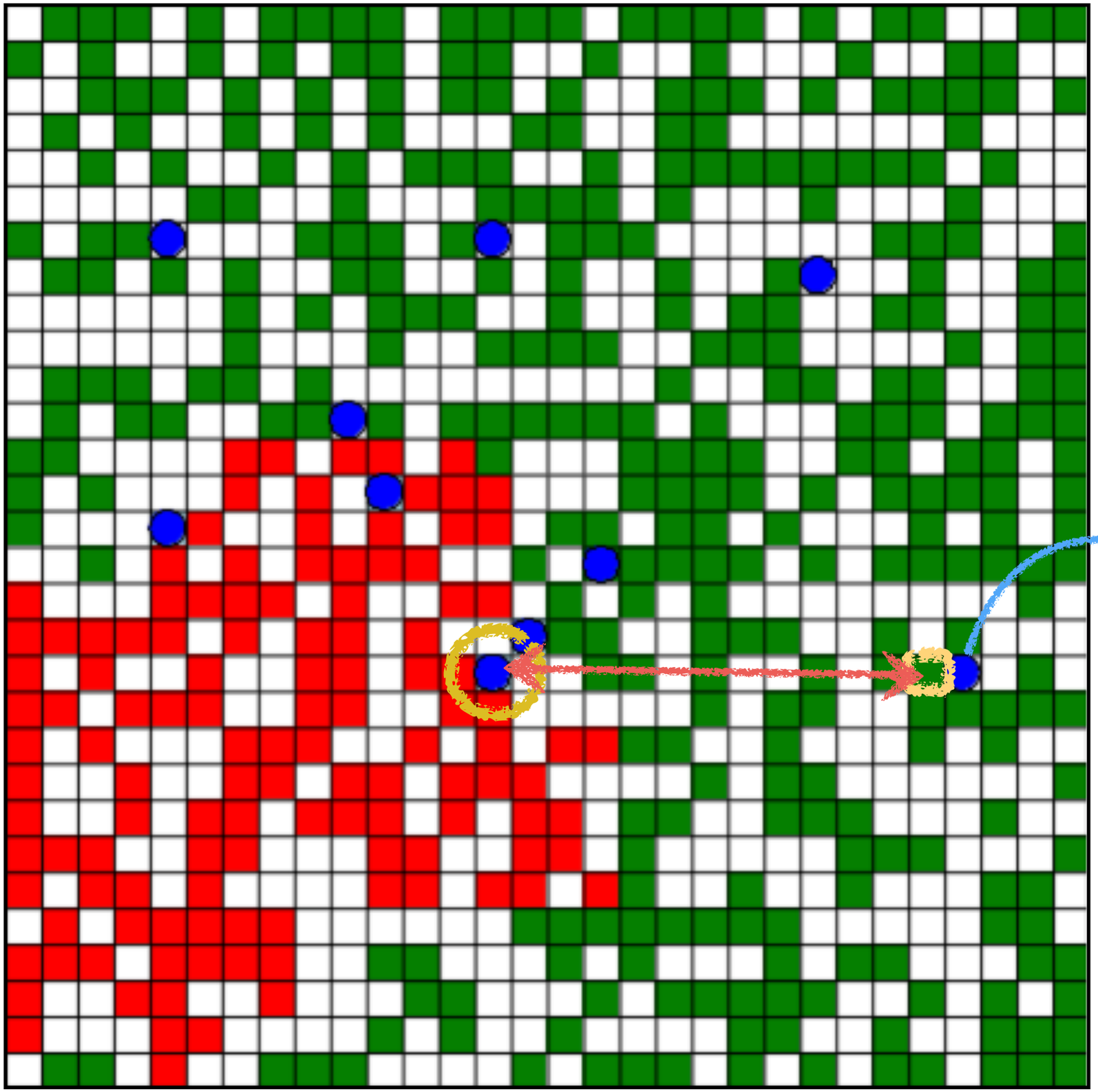
How will it choose its next location ?

Choose a plot toward the closest busy firefighter



How will it choose its next location ?
- Find the closest busy firefighter

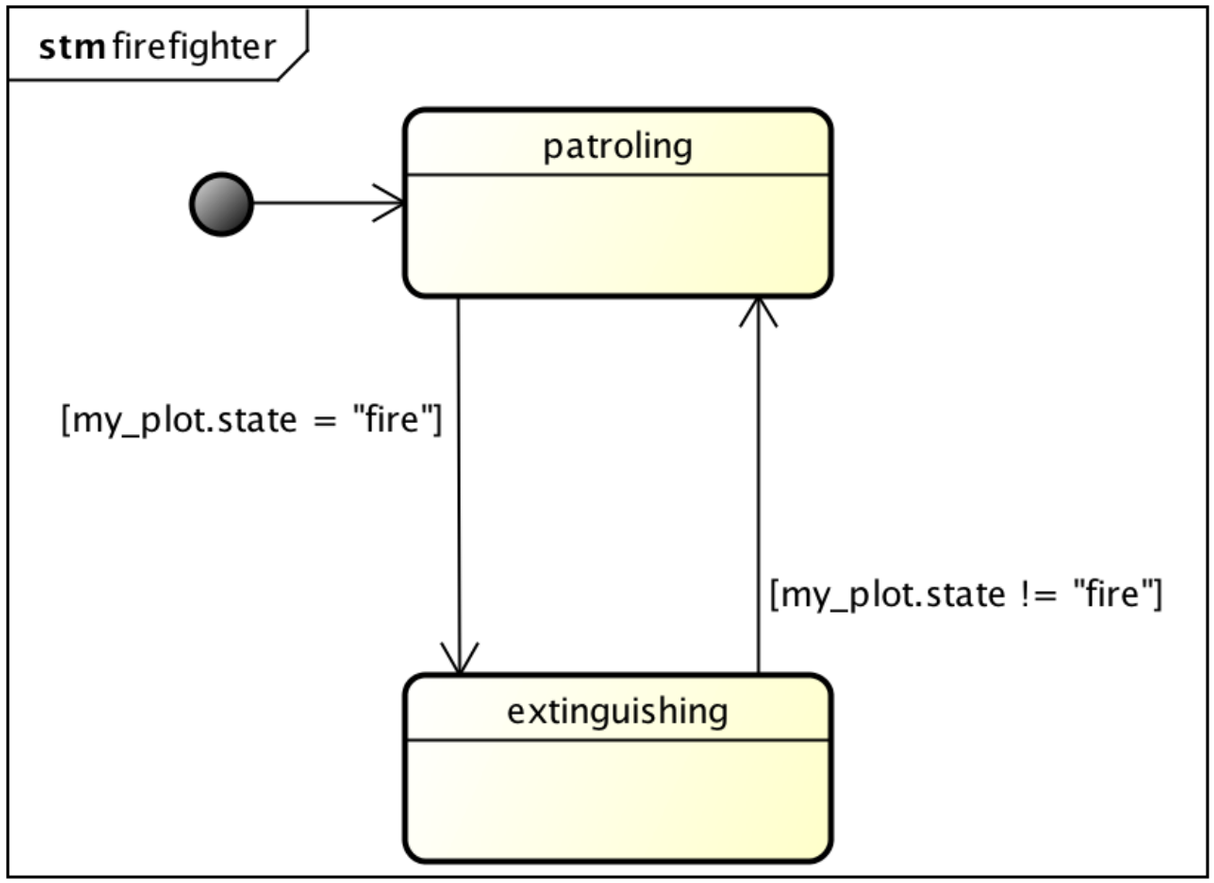
Choose a plot toward the closest busy firefighter



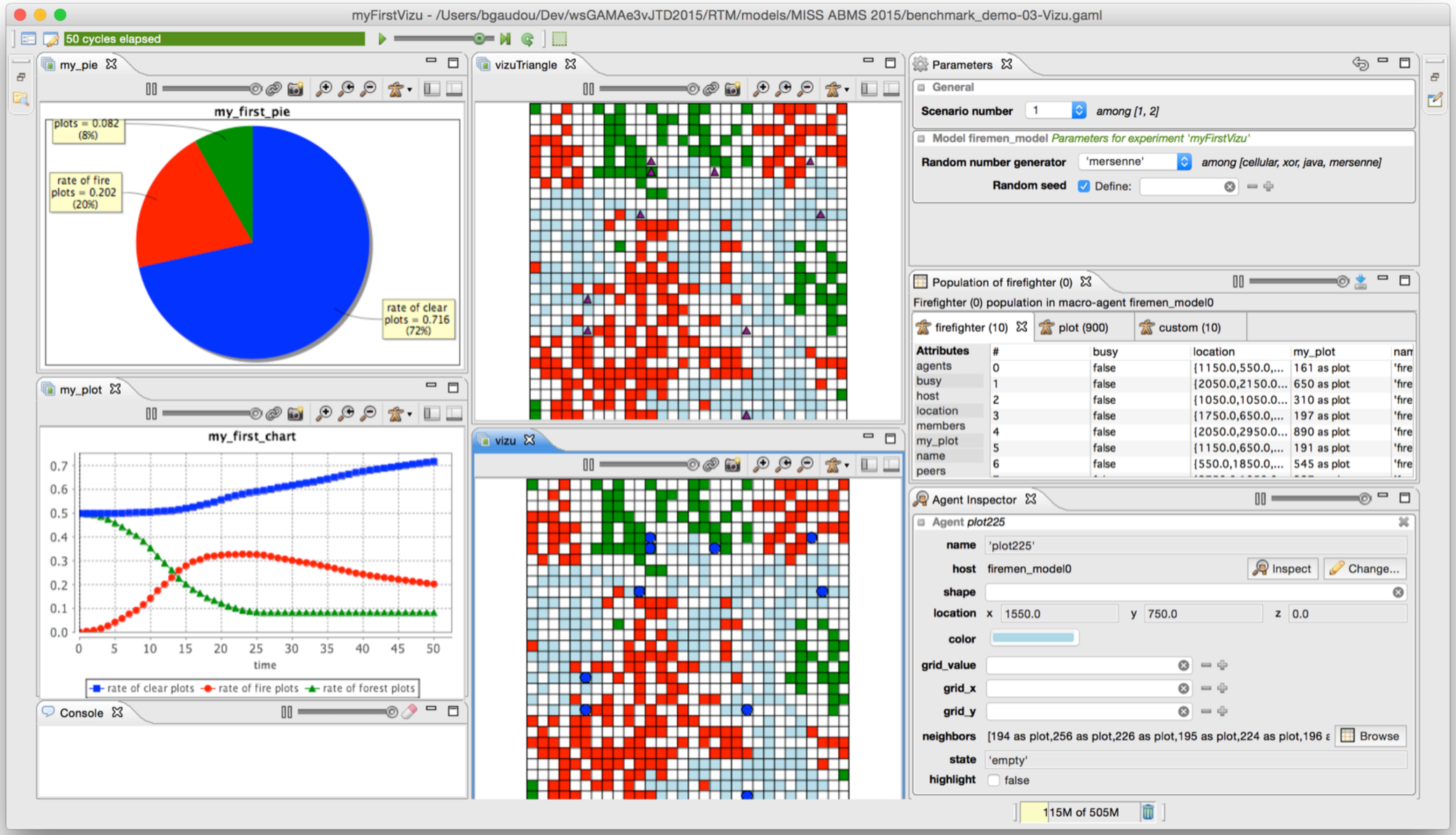
How will it choose its next location ?
- Find the closest busy firefighter
- Find the closest cell to this firefighter

Extensions

- Add a action do_fire to plot
- Stop simulation when no more fire
- Finite State Machine architecture: patrolling and extinguishing are now 2 states (and not 2 reflexes anymore)

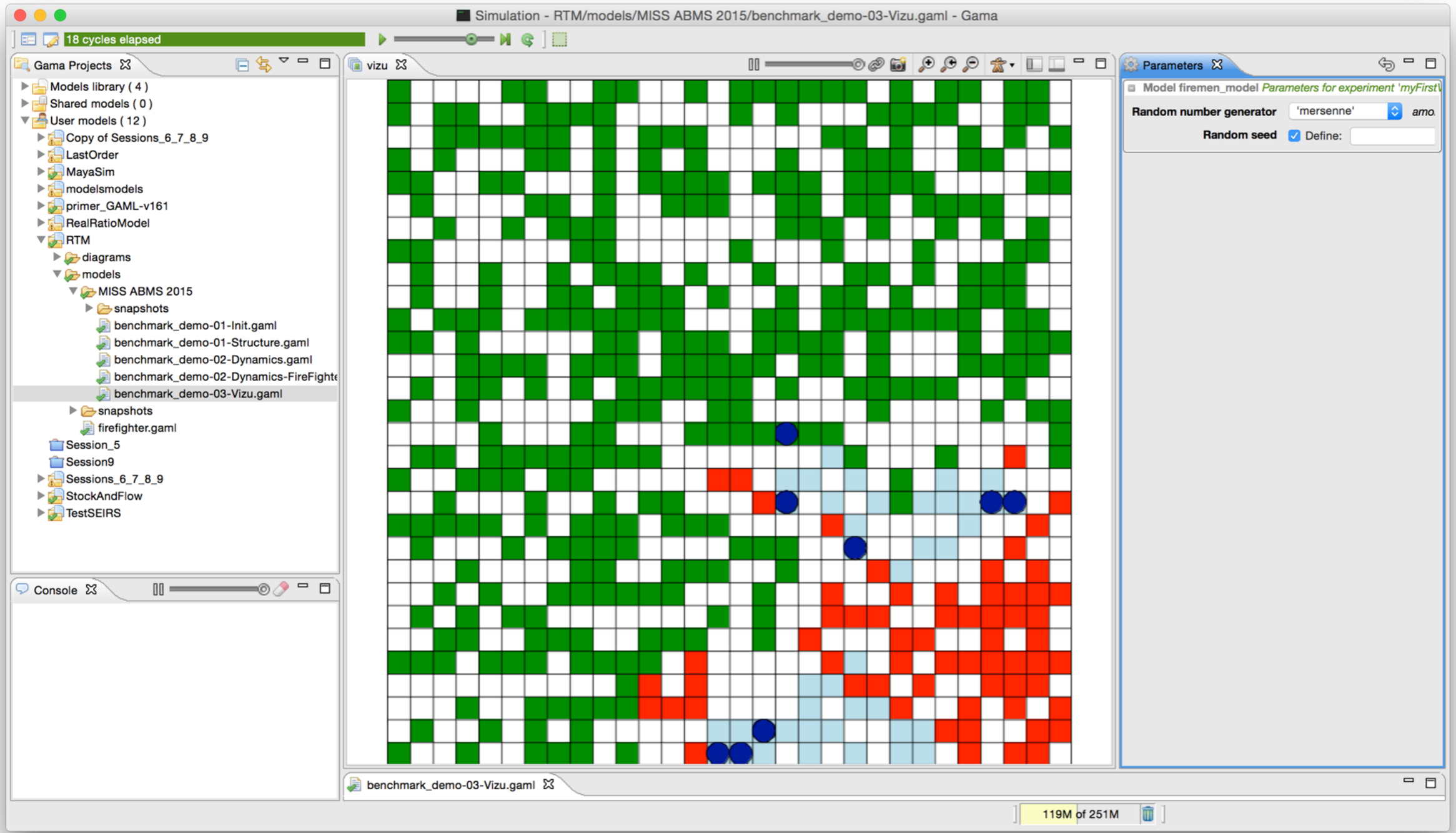


Firefighter model: Visualisation, monitoring and model exploration



Benoit Gaudou (Univ. Toulouse 1)
Patrick Taillandier (INRAE)

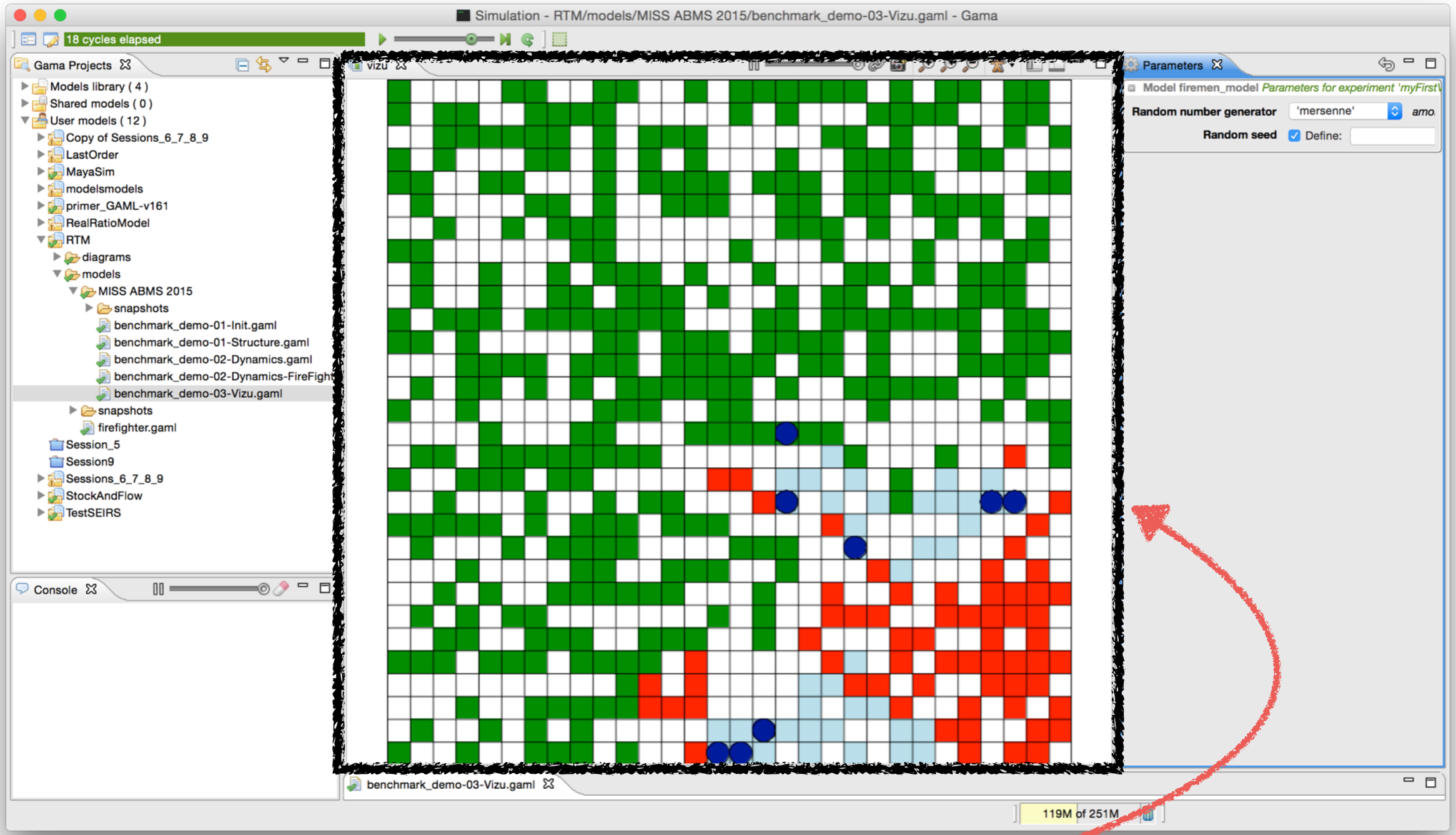
Benchmark model - First visualisation



In GAMA, there are 2 kinds of experiments:

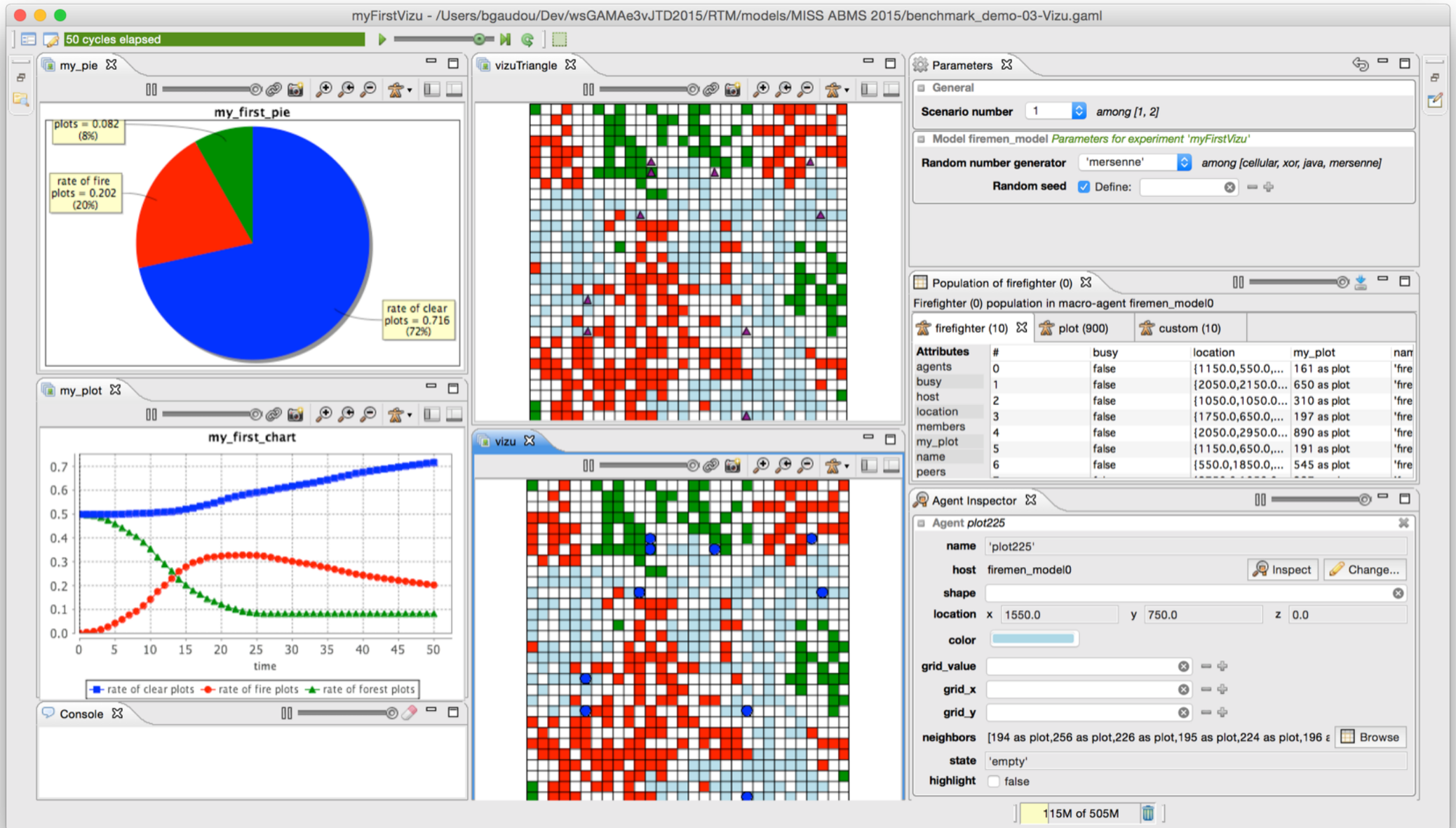
- gui
- batch

Displays in GAMA

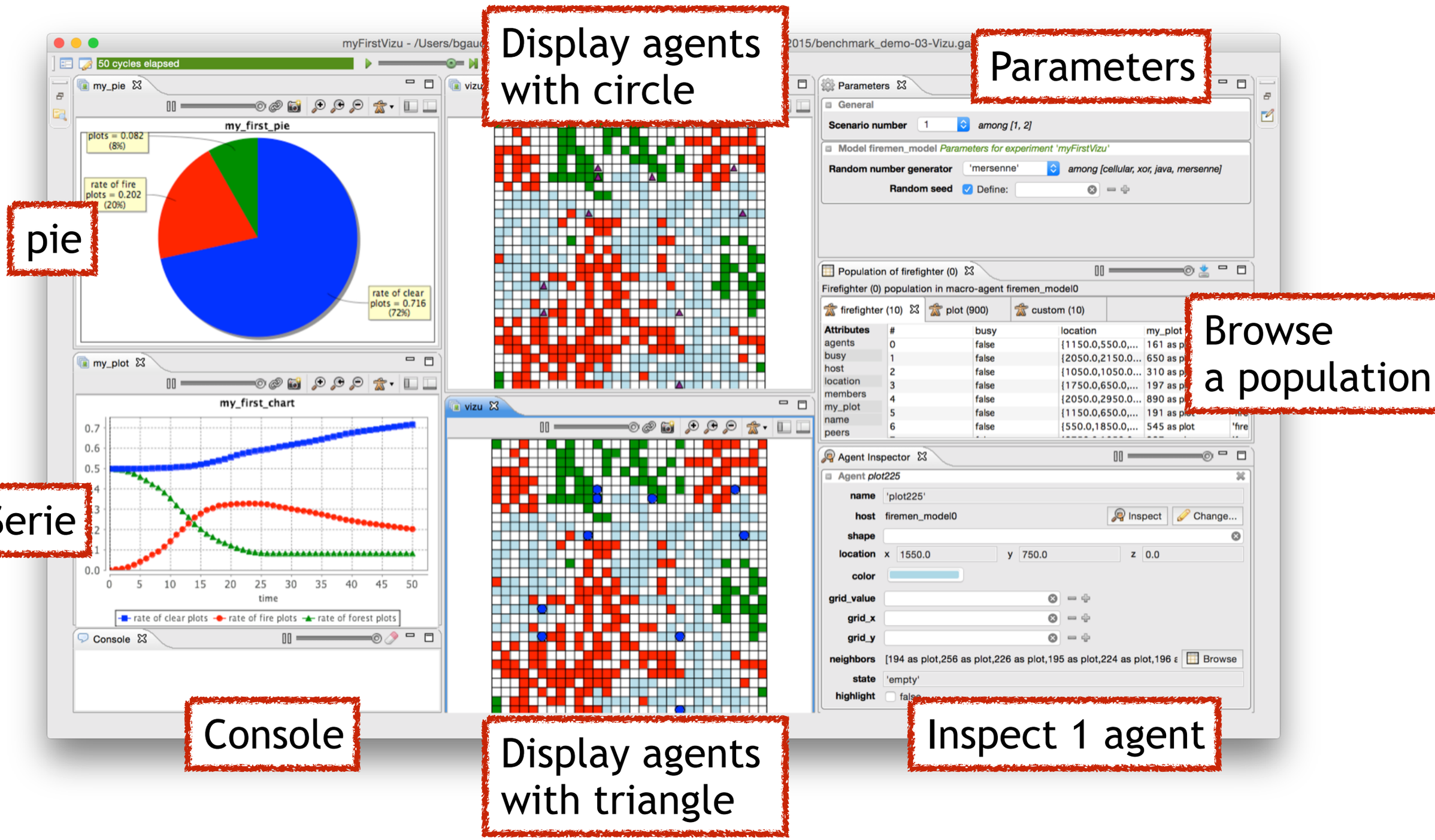


- In GAMA, we can define as many **displays** as needed, each of them represent a point of view on the simulation.

Let’s build our virtual laboratory



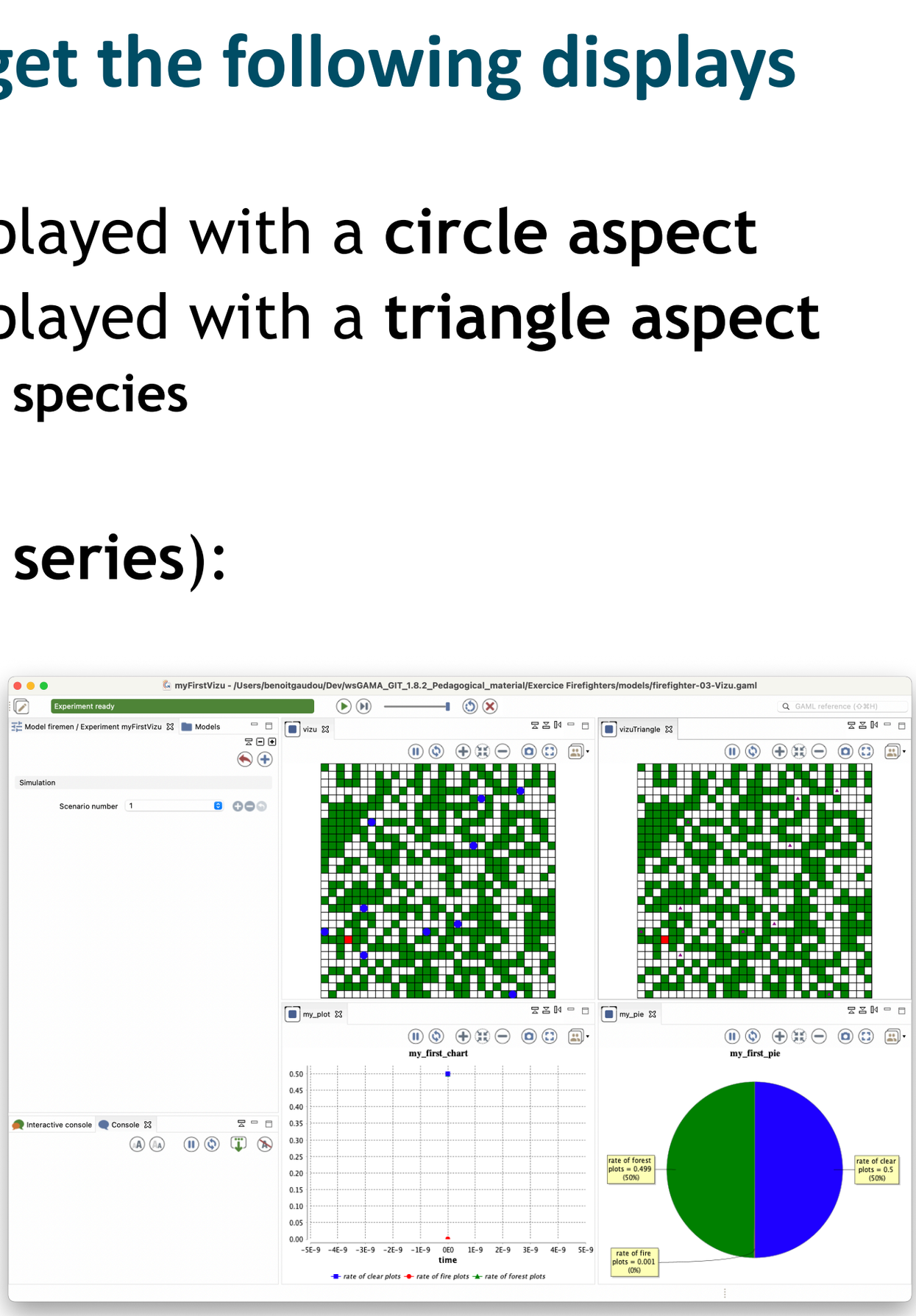
Let’s build our virtual laboratory



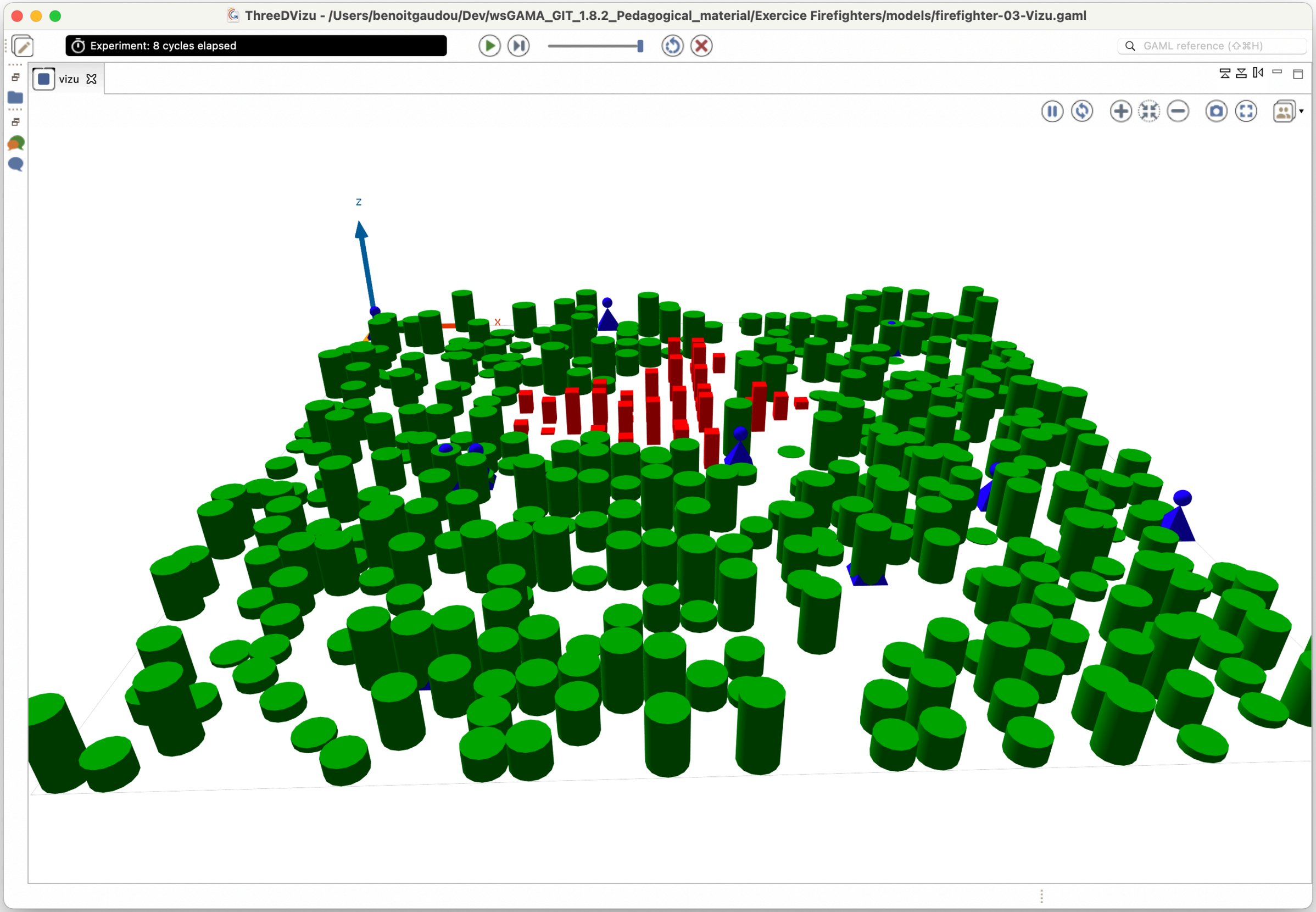
Update the experiment to get the following displays

- 1 display with firefighters displayed with a **circle aspect**
- 1 display with firefighters displayed with a **triangle aspect**
  - **Add corresponding aspects to the species**

- A display plotting (**as a time series**):
  - The **number of forest plots**
  - The **number of on fire plots**
  - The **number of empty plots.**
- A display plotting the same information **as a pie**



Create a new experiment to have a 3D display



Exploration of the model

- **Create a batch experiment to explore the model**
  - Parameter to explore: the scenario
  - Stop condition: when there is no more plot on fire
  - Replication: 10
  - Exploration method: exhaustive method

- **Outputs - Displays (cf. permanent statement)**
  - The rate of remaining forest plots (at the end of each bunch of replications)
  - The end cycle (at the end of each bunch of replications)

- **Outputs - Save in file**
  - Save the scenario number, the id of the replication, and the 2 indicators (remaining forests and end cycle).

firefighter-03-Vizu.gaml | firefighter.csv

Search

scenario	replication	rateForest	End cycle
1	0	0.00444444444444444	347
1	1	0.00444444444444444	347
1	2	0.00444444444444444	347
1	3	0.00444444444444444	347
1	4	0.00444444444444444	347
1	5	0.00444444444444444	347
1	6	0.00444444444444444	347
1	7	0.00444444444444444	347
2	8	0.03333333333333333	725
2	9	0.03333333333333333	725
2	10	0.03333333333333333	725
2	11	0.03333333333333333	725
2	12	0.03333333333333333	725
2	13	0.03333333333333333	725
2	14	0.03333333333333333	725
2	15	0.03333333333333333	725

Exploration of the model