JSensor user Guide

Summary

JSensor without GUI

Creating the configuration file

After a model has been developed in JSensor, it's time to run it. For this we need to have installed JSensor. If you have not, see how to install in the **Installation Guide.**

To run the JSensor without GUI you need to create a configuration file containing the parameters for the simulation. We show here how to create this file and how to execute it.

The image below shows the settings file of flooding model.

JSensor.config

```
project_name = flooding

width = 18000

heigth = 18000

log = true

node_radio = 15

communication_radio = 300

number_of_processors = 4

rounds = 150

nodes = 10000

distribuition_model = RandomDistribution

node_implementation = FloodingNode

connectivity_model = FloodingNodeConectivity

mobility_model = NoMobility

seed_random = 1587632589
```

The field **project_name** should receive the name of the project you want to run, in this example we use the **flooding** model.

The fields **width** and **height** represents the size of the area being simulated.

If you wish to generate logs of the simulation should be assigned to the field **log** the value **true**. Otherwise, the field can be disregarded because by default the JSensor does not generate log.

The field **node_radio** represents the physical radius of the node, ie, its size.

The field **communication_radio** represents the communication radius size of the sensor.

The field **number_of_processors** tells how many cores are used in the simulation.

The field **rounds** represents the maximum number of rounds to be executed. If you want to execute the simulation until the end this field can be disregarded.

The field **seed_random** sets a seed for JSensor. If you do not want to use their own seed this field can be disregarded because the JSensor has a default.

The field **nodes** defines how many nodes are created in the simulation.

The fields **distribuition_model, node_implementation, connectivity_model e mobility_model** defines which implementations JSensor will use for the creation of the nodes that have been defined in the last field.

You can generate more than one type of node in the same simulation. To do this simply add to fields **distribuition_model, node_implementation, connectivity_model** and **mobility_model** the representation of vector, as shown in the figure below.

JSensor.config

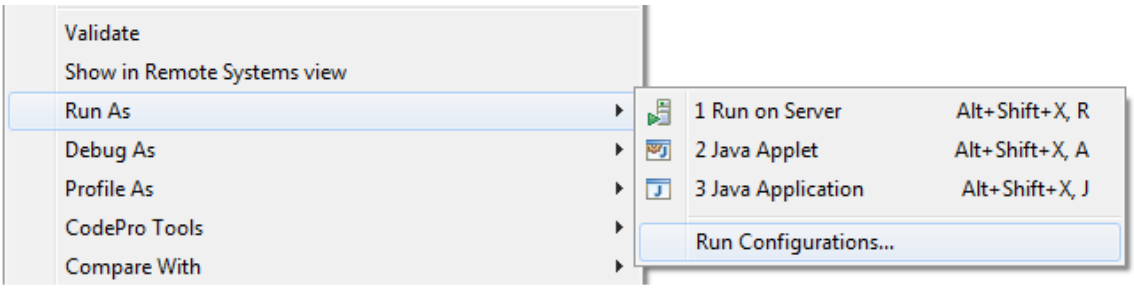
```
1  nodes[1] = 10000
2
3  distribuition_model[1] = RandomDistribution
4
5  node_implementation[1] = FloodingNode
6
7  connectivity_model[1] = FloodingNodeConectivity
8
9  mobility_model[1] = NoMobility
```

Execution by command line

To run the JSensor without GUI you need to call it via the command line (prompt or shell). How the executable JSensor is a .jar, to run you must use the command Java –jar JSensor.jar Flooding.config. Note that was passed a second argument beyond the .jar file. This argument is the configuration file where the settings for the application are defined. In this example we use the settings file Flooding.

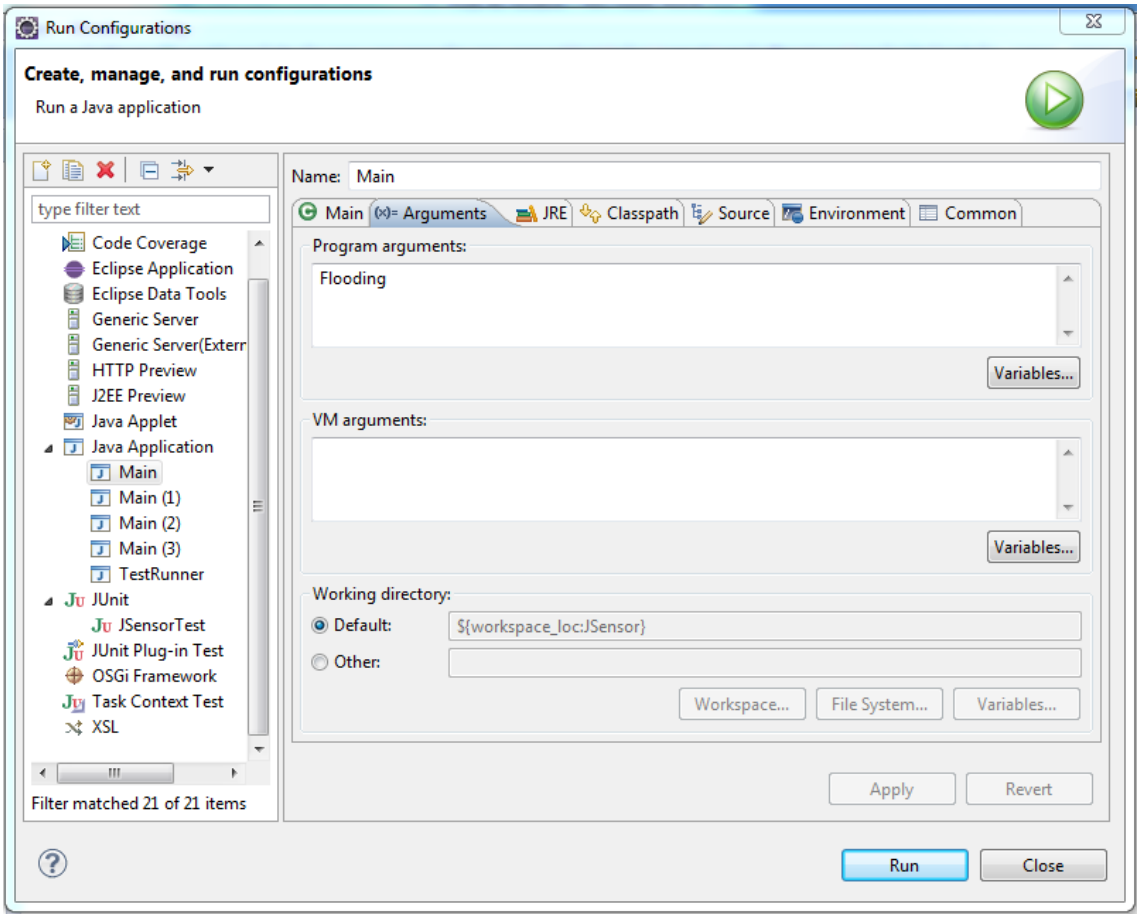
Execution by the Eclipse IDE

To run JSensor using **Eclipse IDE** just pass the settings file as parameter, in this case we use the Flooding model. To do this click on the project with the right button and access the menu **Run As -> Run Configurations.**



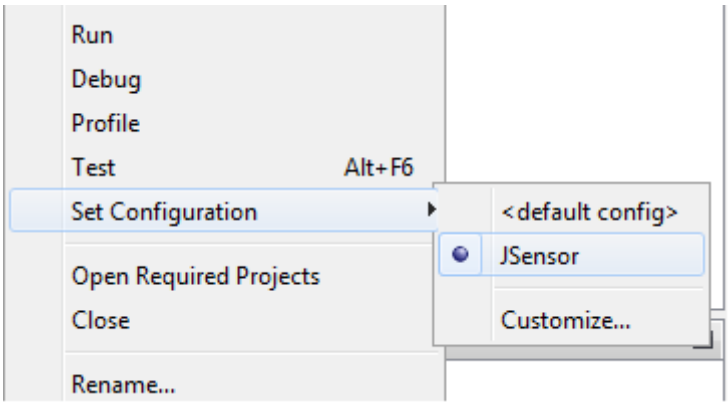
Thus the Run Configurations window opens. Then, go to the **Java Application** menu and select the project to be executed. After choosing the project, go to the **Arguments** tab and in **Param Arguments** put the settings file of the model you want to run. In this example we are using the flooding model.

Now just click on **Run** to execute the JSensor using the flooding model.

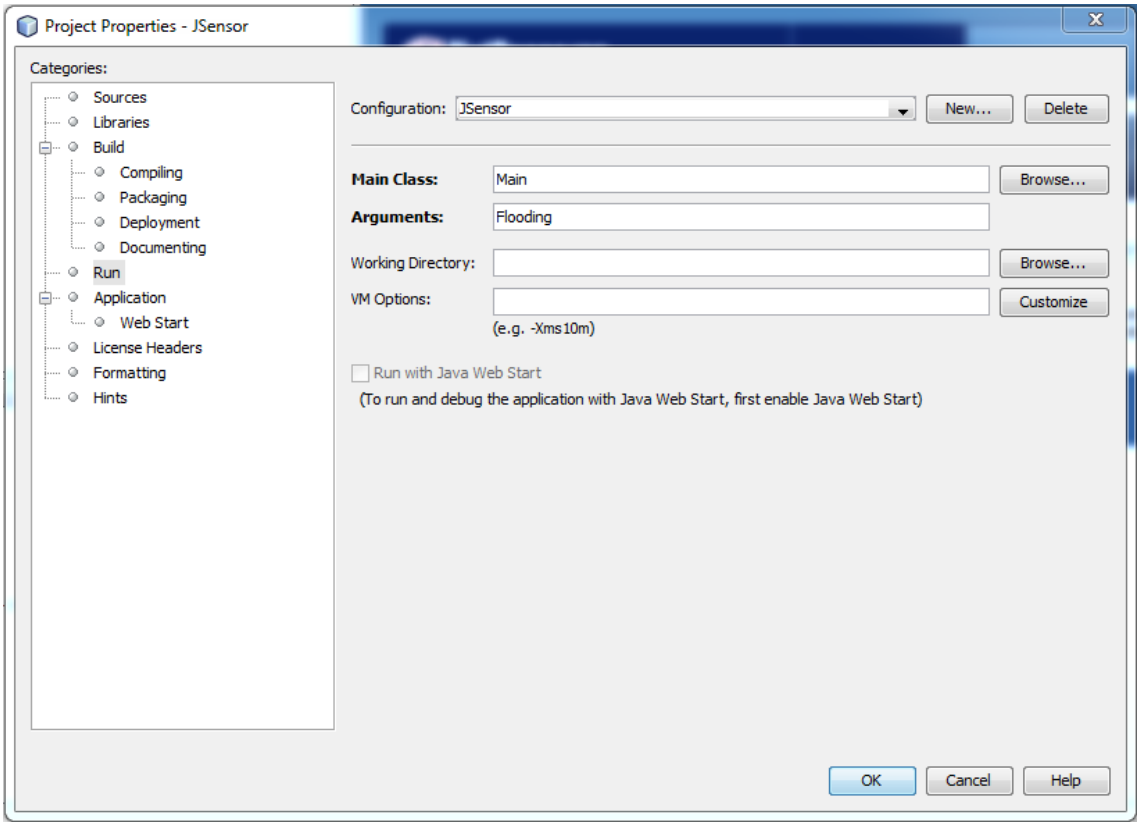


Execution by the NetBeans IDE

To run JSensor using **NetBeans** IDE just pass the settings file as parameter, in this case we use the Flooding model. To do this click on the project with the right button and access the menu **Set Configuration -> Customize.**



Thus the Project Properties window opens. Then, go to the **Run** menu In the field **Arguments** and put the settings file you want to run the model. In this example we are using the **flooding** model.



Now just click on **ok,** so the sensor is configured to use the flooding model.

Common Problems

- **Excess Threads**

  If the message "The number of threads must be less than or equal to the number of processing cores" appears, it means that you are trying to run a simulation in which the number of threads passed as a parameter in the configuration file of the simulation exceeds the number of cores in the computer. The JSensor does not allow the number of threads exceeds the number of cores, since it would result in more context switches (competition for CPU's) decreasing the performance of the simulator.

  Thus, if this error occurs simply change the number of threads passed as a parameter in the configuration file to a value that is compatible with the number of cores in the computer.