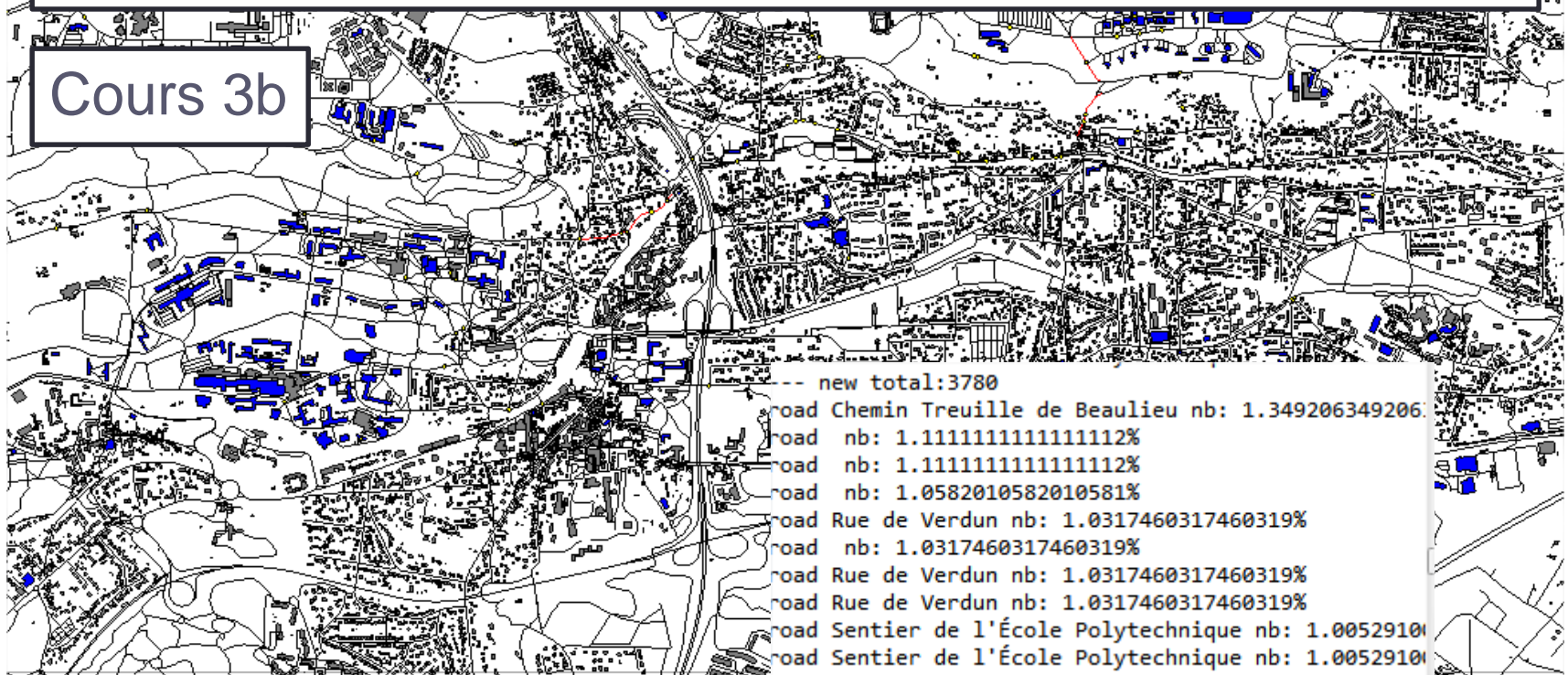


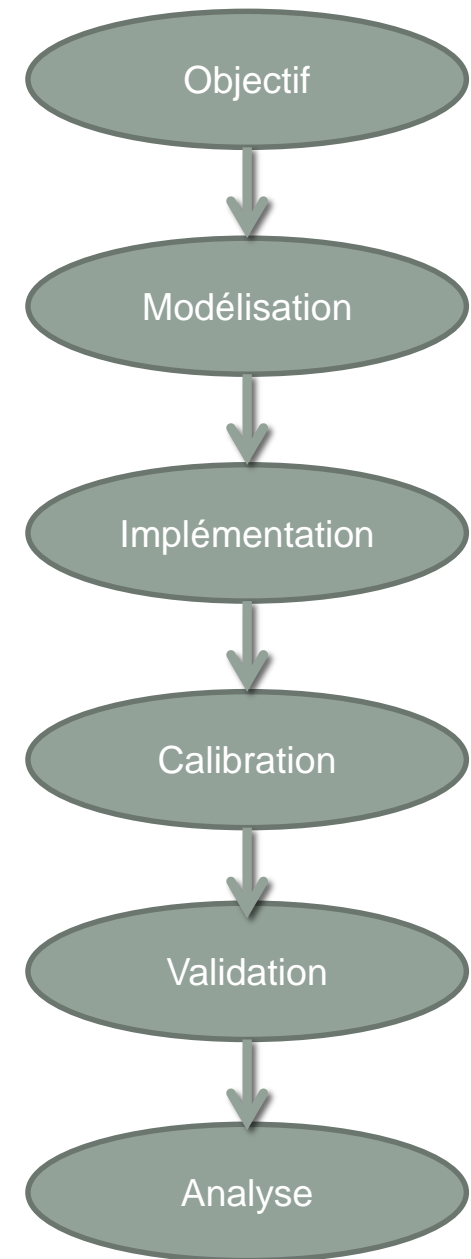
GAMA: EXEMPLE DE DEMARCHE

Cours 3b



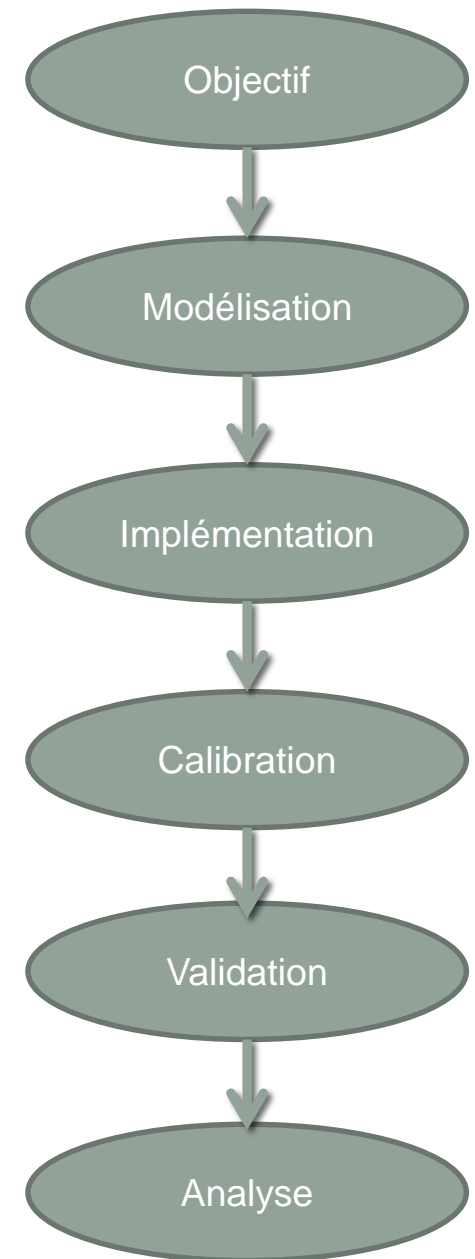
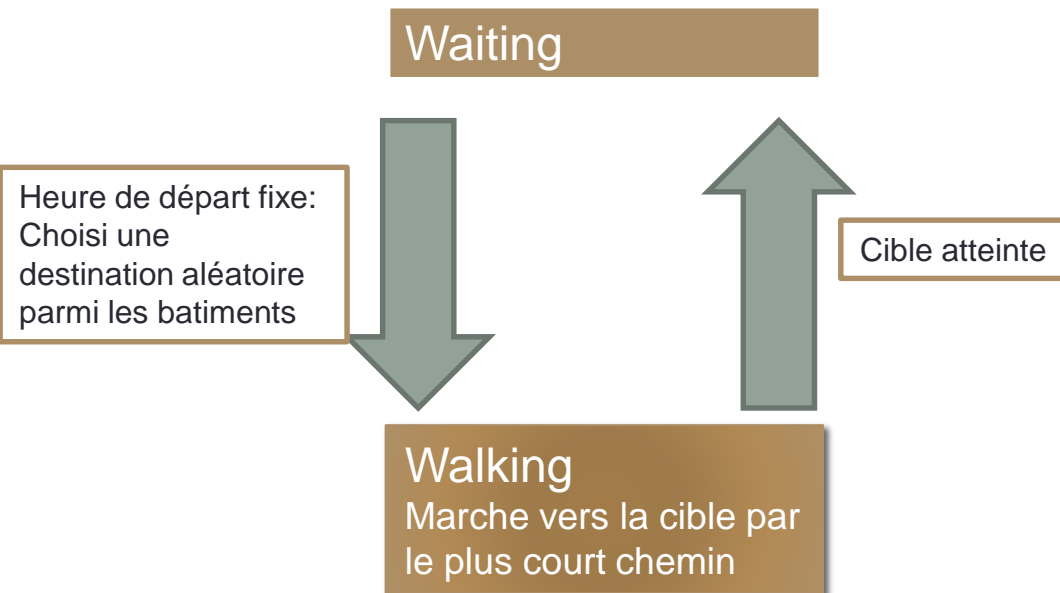
Objectif

- Ici
 - Etudier les voies les plus utilisées sur le campus pour mettre des panneaux d'information / développer la sécurité (trottoirs,...)
- Projet
 - Etudier les déplacements en bus
 - Etudier les propriété d'un réseau de neurone
 - Etudier les stratégies d'un jeu de combat
 - ...



Modélisation

- Environnement:
 - Ensemble de bâtiments
 - Graphe de routes permettant d'aller d'un bâtiment à l'autre
- Agents étudiant passant d'un bâtiment à l'autre
 - Modèle de comportement



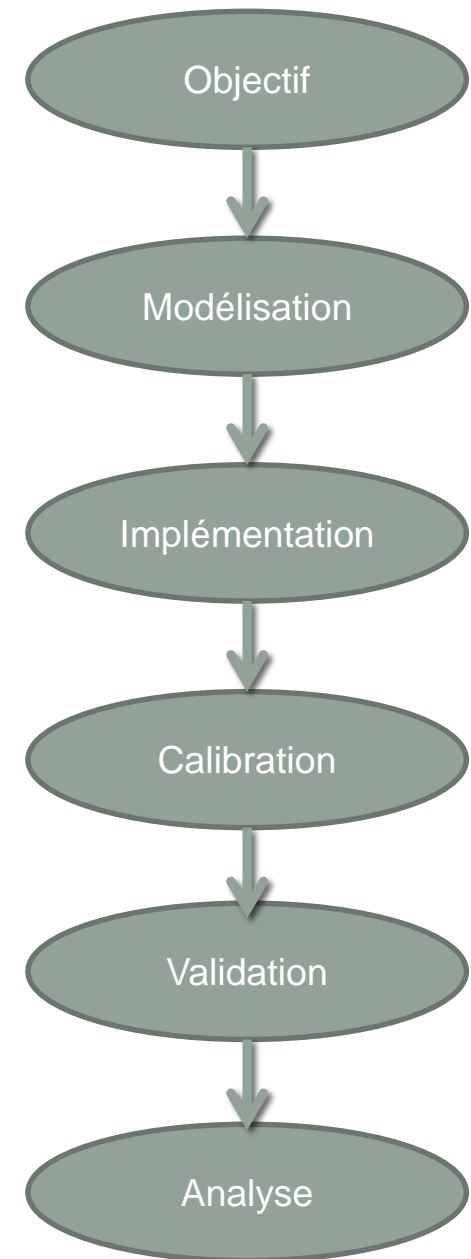
Implémentation

The screenshot shows the GAMA software interface for a road traffic simulation. The main window displays a map with roads and buildings. The left sidebar shows a project tree with folders like 'Models library', 'Shared models', and 'User models'. The bottom console shows the execution of a NetLogo-like script for creating roads and agents.

```

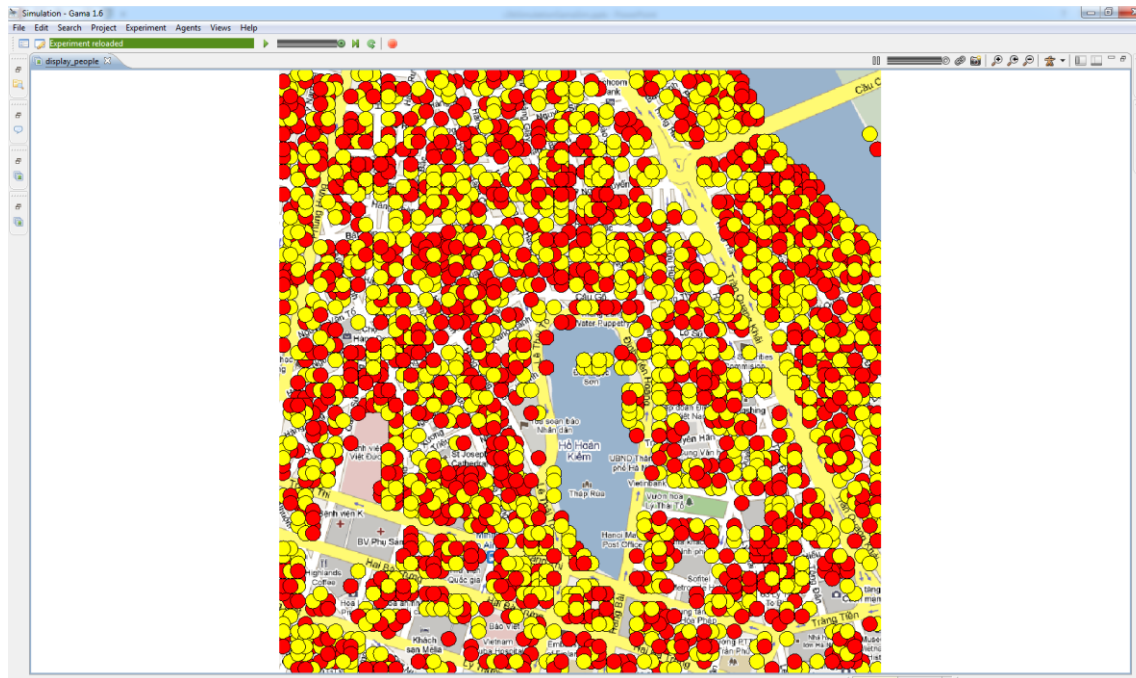
121 create road from: shape_file_roads with: [type::string(read (
122   color<-#blue;
123   if type=""
124   {
125     write("kill "+self.name);
126     do die;
127   }
128   if type="footway" {
129     color <- #green ;
130   }
131 }
132 write("roads :"+length(road));
133
134 the_graph <- as_edge_graph(road);
135 // bigcomponent <- connected_components_of (the_graph);
136
137 target<-one of(road where (each.name="Rue Nicolas Appert")):

```



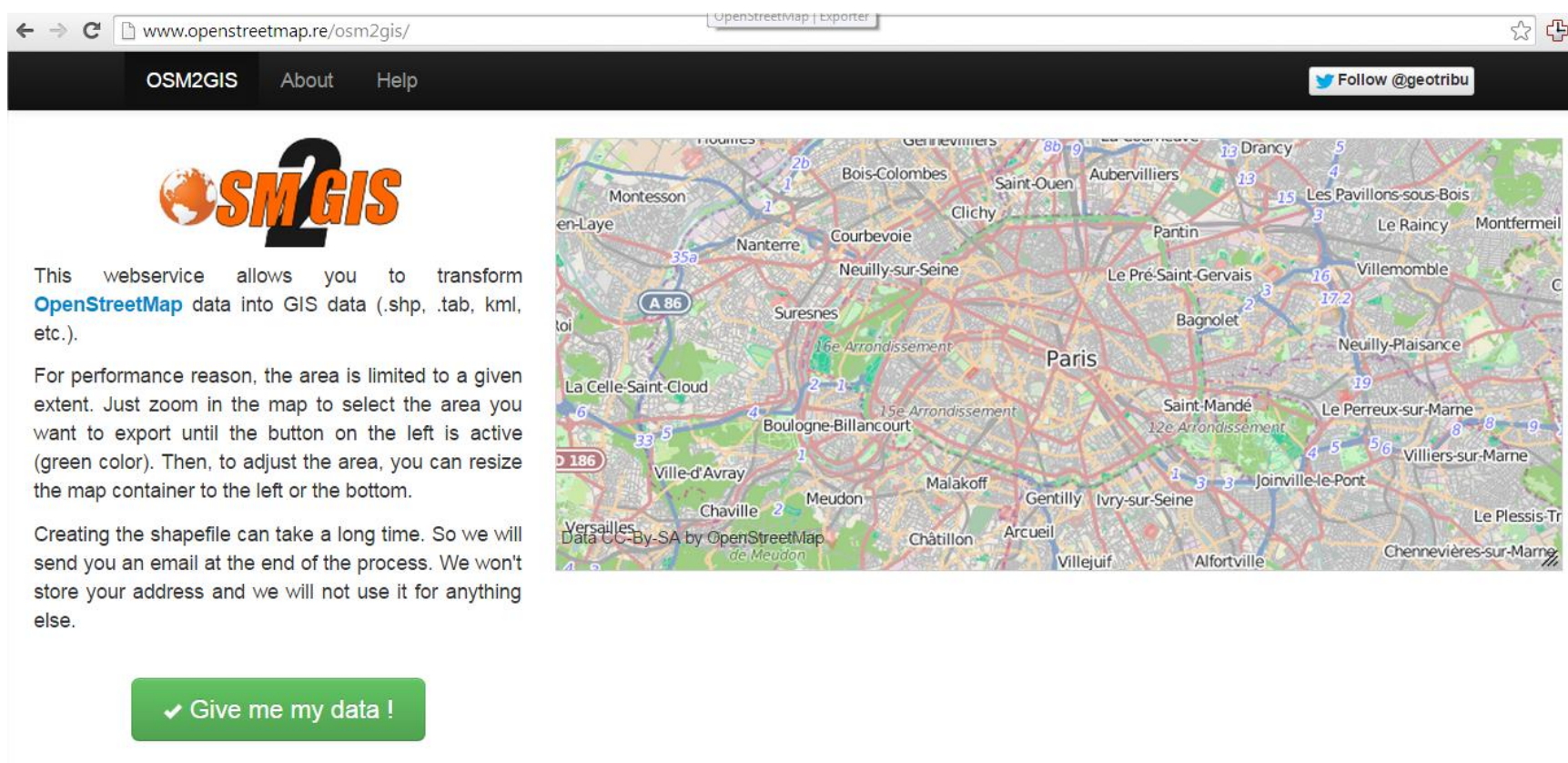
Etape 1: importation des données géographiques

- Première solution: utiliser une image Bitmap et utiliser les couleurs
 - voir modèle Toy Models / Segregation / Segregation (Google Map)



Deuxième solution: OpenStreetMap

- Etape 1: Exporter avec OSM2GIS



← → ↻ www.openstreetmap.re/osm2gis/ OpenStreetMap | Exporter ☆ +

OSM2GIS About Help [Follow @geotribu](#)

OSM2GIS

This webservice allows you to transform **OpenStreetMap** data into GIS data (.shp, .tab, kml, etc.).

For performance reason, the area is limited to a given extent. Just zoom in the map to select the area you want to export until the button on the left is active (green color). Then, to adjust the area, you can resize the map container to the left or the bottom.

Creating the shapefile can take a long time. So we will send you an email at the end of the process. We won't store your address and we will not use it for anything else.

✓ Give me my data !

- Etape 2: explorer les données avec Qgis
 - Ajouter les .shp dans « Ajouter couche vectorielle »
 - Utiliser la couche point pour centrer
 - Explorer les propriétés des polygones (batiments) et routes (line) avec les tables d'attributs des couches

The screenshot shows the QGIS 2.0.1-Dufour interface. The main map area displays a green background with red points and blue lines representing OpenStreetMap data. The 'Couches' (Layers) panel on the left shows three layers: 'osm_point', 'osm_line', and 'osm_polygon'. The 'Table attributaire - osm_line' window is open, showing a table with 20 rows and 10 columns. The table contains attributes for various road types, including 'leisure', 'lock', 'man_made', 'military', 'motorcar', 'name', 'natural', 'office', and 'on'. The 'name' column contains street names like 'RER C Tronçon ...', 'Rue de Verdun', 'L'Yvette', 'L'Orge', 'TGV Atlantique ...', 'RER C Tronçon ...', 'Grande Ceinture', 'Rue Noetzin', 'Chemin du Pet...', 'Rue Louise Weiss', 'Rue d'Arsonval', 'Rue Louise Weiss', 'Rue du Général ...', 'Rue Noetzin', 'Rue Louis de Br...', 'Rue du Doyen ...', 'Rue Noetzin', and 'Bus 7 : Orsay-Vi...'. The 'office' column has 'yes' for some entries, while others are 'NULL'.

	leisure	lock	man_made	military	motorcar	name	natural	office	on
0	NULL	NULL	NULL	NULL	NULL	RER C Tronçon ...	NULL	NULL	NULL
1	NULL	NULL	NULL	NULL	NULL	Rue de Verdun	NULL	NULL	NULL
2	NULL	NULL	NULL	NULL	NULL	L'Yvette	NULL	NULL	NULL
3	NULL	NULL	NULL	NULL	NULL	L'Orge	NULL	NULL	NULL
4	NULL	NULL	NULL	NULL	NULL	TGV Atlantique ...	NULL	NULL	NULL
5	NULL	NULL	NULL	NULL	NULL	RER C Tronçon ...	NULL	NULL	NULL
6	NULL	NULL	NULL	NULL	NULL	Grande Ceinture	NULL	NULL	NULL
7	NULL	NULL	NULL	NULL	NULL	Rue Noetzin	NULL	NULL	NULL
8	NULL	NULL	NULL	NULL	NULL	Chemin du Pet...	NULL	NULL	NULL
9	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
10	NULL	NULL	NULL	NULL	NULL	Rue Louise Weiss	NULL	NULL	NULL
11	NULL	NULL	NULL	NULL	NULL	Rue d'Arsonval	NULL	NULL	NULL
12	NULL	NULL	NULL	NULL	NULL	Rue Louise Weiss	NULL	NULL	yes
13	NULL	NULL	NULL	NULL	NULL	Rue du Général ...	NULL	NULL	NULL
14	NULL	NULL	NULL	NULL	NULL	Rue Noetzin	NULL	NULL	NULL
15	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
16	NULL	NULL	NULL	NULL	NULL	Rue Louis de Br...	NULL	NULL	NULL
17	NULL	NULL	NULL	NULL	NULL	Rue du Doyen ...	NULL	NULL	yes
18	NULL	NULL	NULL	NULL	NULL	Rue Noetzin	NULL	NULL	yes
19	NULL	NULL	NULL	NULL	NULL	Bus 7 : Orsay-Vi...	NULL	NULL	NULL

- Etape 3: importer dans GAMA

```

file shape_file_buildings <- file("../includes/osm_polygon.shp");
file shape_file_roads <- file("../includes/osm_line.shp");
file shape_file_bounds <- file("../includes/osm_point.shp");
geometry shape <- envelope(shape_file_bounds);
float step <- 10 #mn;

init {
  create building from: shape_file_buildings with: [type::string(read ("building")), amenity::string(read ("amenity"))] {
    if (type!="yes" and type!="station") {
//      write("kill "+self.name);
      do die;
    }
    if name contains "Bâtiment"
    {
      color<-#blue;
    }
    if name!=""
    {
      color<-#blue;
    }
  }
  create road from: shape_file_roads with: [type::string(read ("highway"))] {
    color<-#blue;
    if type=""
    {
      write("kill "+self.name);
      do die;
    }
    if type="footway" {
      color <- #green ;
    }
  }
}

```

- Exercice

- Changer la couleur des voies plus importantes
 - (Highway=« secondary »)

Etape 2: créer les agents people

```
bat_univ <- building where (each.color=#blue);  
  
create people number: nb_people {  
  location <- any_location_in (one_of (bat_univ));  
}
```

```
species people {  
  rgb color <- #yellow ;  
  
  aspect base {  
    draw circle(10) color: color;  
  }  
}
```

Etape 3: nettoyer le graphe et ajouter le comportement des agents people

- Création du graphe...

```
the_graph <- as_edge_graph(road);  
target<-one_of(road where (each.name="Rue Nicolas Appert"));  
do clean_roads;
```

- Clean_roads pour nettoyer un peu les données!

Le comportement de l'agent people

```
species people skills:[moving] control:fsm{
  rgb color <- #yellow ;
  building gare_place <- nil ;
  building working_place <- nil ;
  int start_work ;
  int end_work ;
  point the_target <- nil ;
  path currentpath;

  aspect base {
    draw circle(10) color: color;
  }

  state waiting initial:true{
    transition to:moving when:(current_hour = start_work)
    {
      building bt<-one_of(bat_univ);
      the_target <- any_location_in(bt);
      currentpath<-path_between(the_graph,location,the_target);
      if (currentpath=nil)
      {
        write "no path to"+bt.name;
        location<-the_target;
      }
    }
  }

  state moving {
    do goto target: the_target on: the_graph ;

    transition to:waiting when:the_target.location = location
    {
      the_target <- nil ;
    }
  }
}
```

Etape 4: variables observées

```

state waiting initial:true{
  transition to:moving when:(current_hour = start_work)
  {
    building bt<-one_of(bat_univ);
    the_target <- any_location_in(bt);
    currentpath<-path_between(the_graph,location,the_target);
    if (currentpath=nil)
    {
      write "no path to"+bt.name;
      location<-the_target;
    }
    else
    {
      loop ro over:currentpath.edges
      {
        if (species(ro)=road)
        {
          (ro as road).nbpeople<-(ro as road).nbpeople+1;
        }
      }
    }
  }
}

```

```

Console
road nb: 2.511370377694285%
--- new total:5057
road Rue du Château nb: 3.8560411311053984%
road Rue de la Guyonnerie nb: 3.6187462922681433%
road Rue du Château nb: 3.6187462922681433%
road nb: 3.2430294641091555%
road nb: 3.2430294641091555%
road nb: 3.2430294641091555%
road Chemin Rural de Petit Saclay à la Pacaterie nb
road nb: 2.570694087403599%
road Allée des Découvertes nb: 2.570694087403599%
road nb: 2.511370377694285%
--- new total:5057

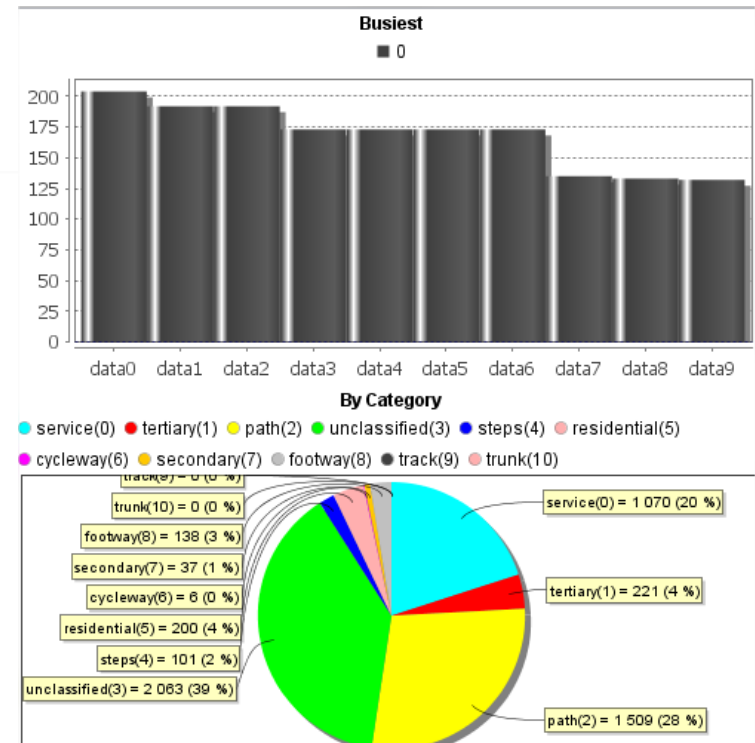
```



```

reflex updategraph
{
  busyroads<-10 last (connected_roads sort_by each.nbpeople);
  int total<-sum(connected_roads collect each.nbpeople)+1;
  if ((time mod 10)=0)
  {
    write("--- new total:"+total);
    loop ro over:busyroads
    {
      write("road "+ro.name+" nb: "+(ro.nbpeople*100/total)+"%");
    }
  }
  sumnb<-[];
  loop ty over:names
  {
    list<road> ro<-(connected_roads where (each.type contains ty));
    if (length(ro)=0)
    {
      add 0 to:sumnb;
    }
    else
    {
      add sum(ro collect each.nbpeople) to:s
    }
  }
}

```



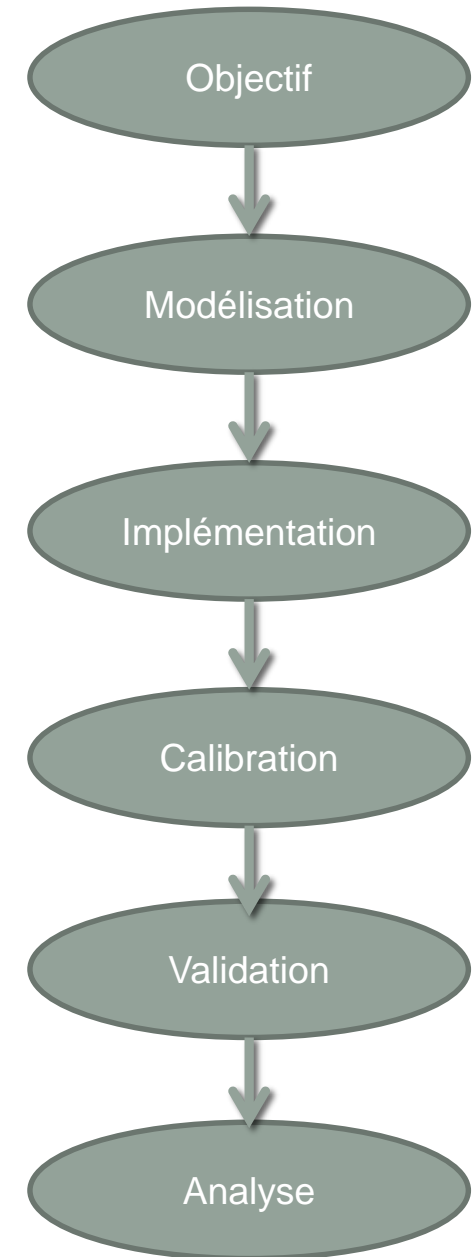
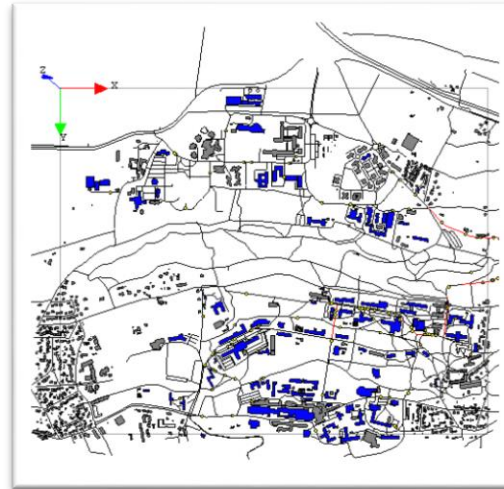
```

output {
  display city_display type:opengl {
    species building aspect: base ;
    species road aspect: base ;
    species people aspect: base ;
  }
  display chart_display refresh_every: 10 {
    chart "Busiest" type: histogram size: {1, 0.5} position: {0, 0} {
      datalist value: (busyroads collect each.nbpeople) style:bar;
    }
    chart "By Category" type: pie size: {1, 0.5} position: {0, 0.5} {
      datalist legend: names value: sumnb;
    }
  }
}

```


Calibration

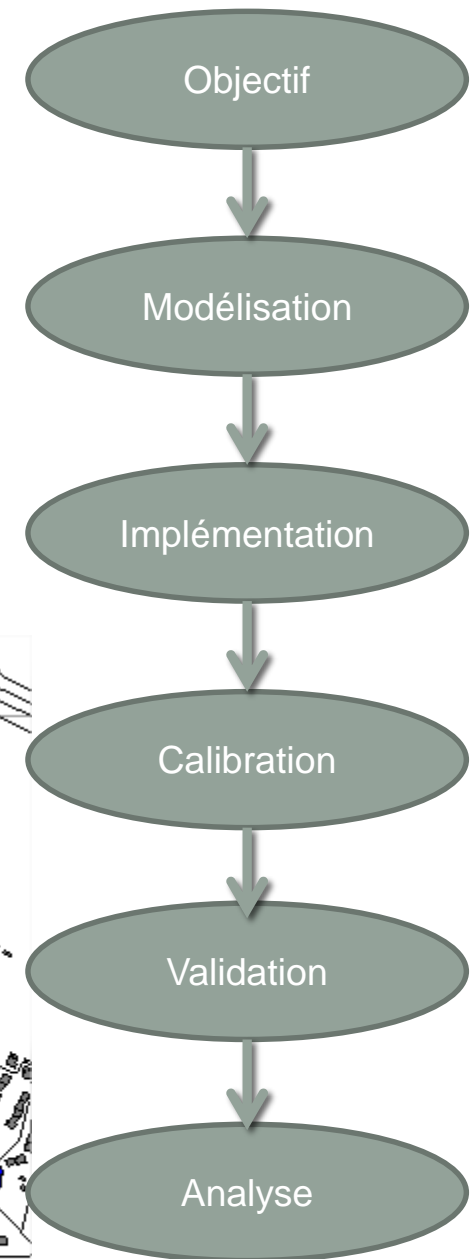
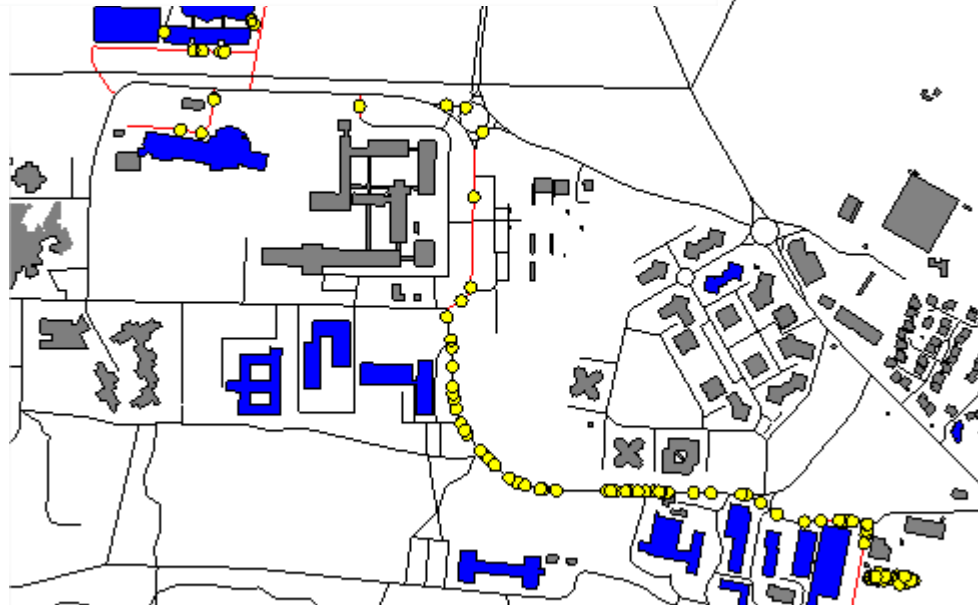
- 4 jeux de données OpenStreetMap:
 - Normal (université)
 - Medium (ajout de Polytechnique)
 - Large
 - Very Large
- + 1 configuration de validation
- Calibration des paramètres des agents-étudiants
 - Par l'expert...
 - Pas de calibration automatique nécessaire par optimisation



Etape 5: Validation interne

- Vérification de cohérence sur le trajet AdaLovelace-CESFO

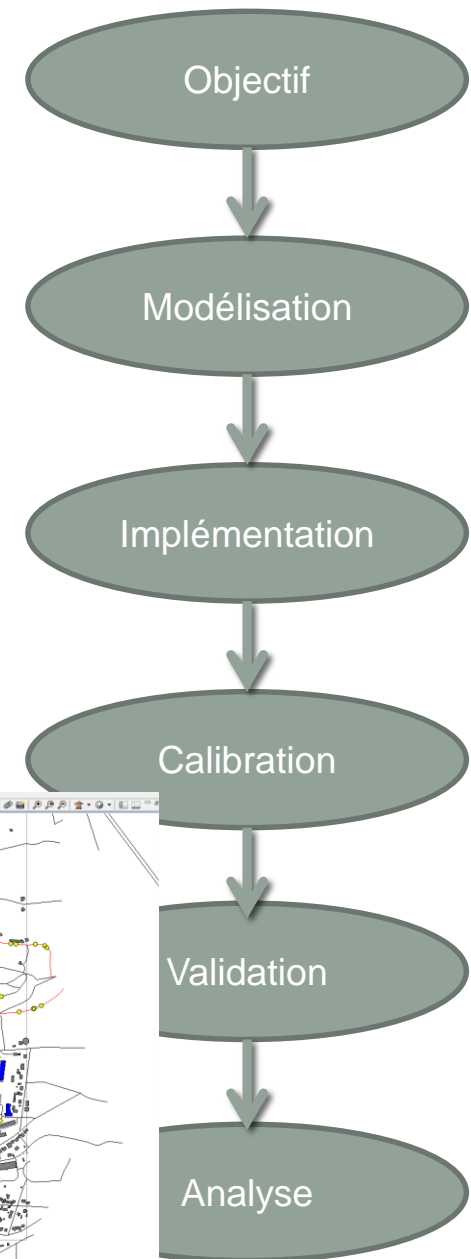
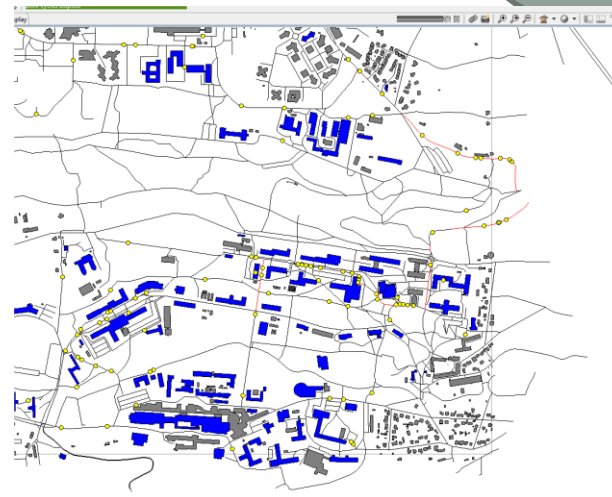
```
state waiting initial:true{  
  transition to:moving when:(current_hour = start_work)  
  {  
    building bt<-one_of(bat_univ where (each.name contains "CESFO"));  
    the_target <- any_location_in(bt);  
    location<-the_target;  
    building bt<-one_of(bat_univ where (each.name contains "Ada Lovelace"));  
    the_target <- any_location_in(bt);  
    currentpath<-path_between(the_graph,location,the_target);  
    if (currentpath=nil)  
    {  
      write "no path to"+bt.name;  
      location<-the_target;  
    }  
  }  
}
```

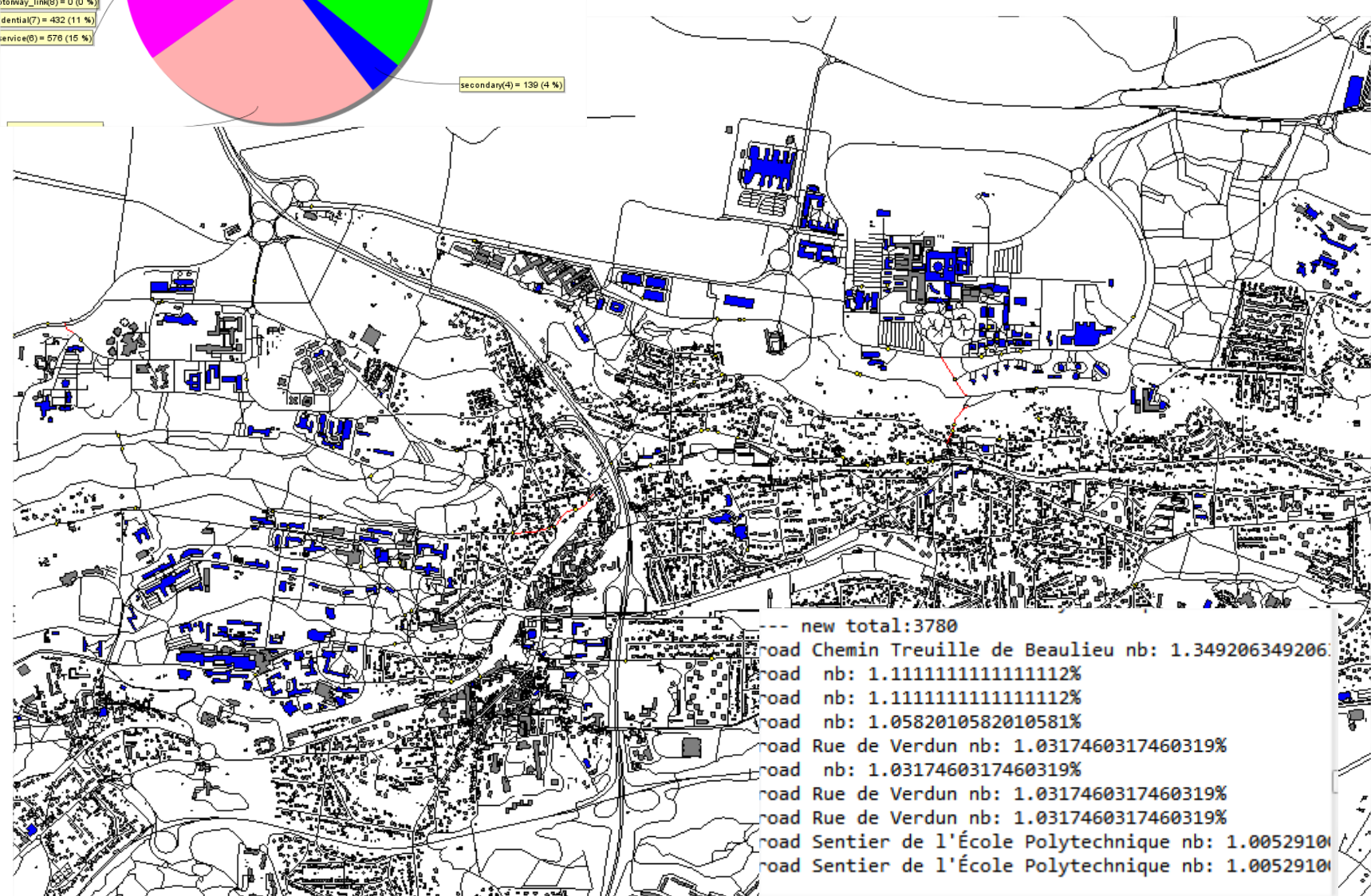
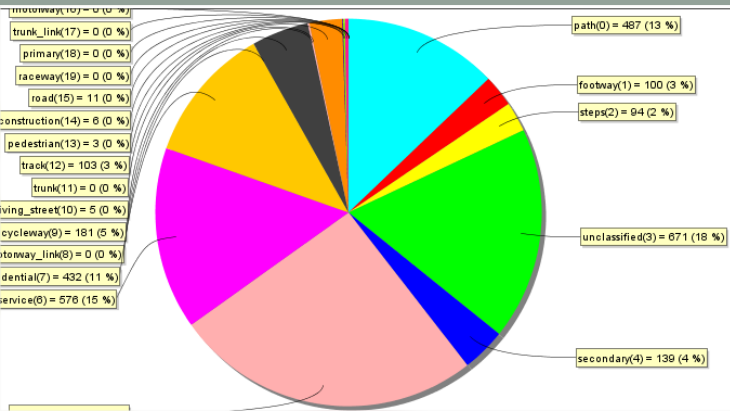


Validation externe

- Données observées empirique:
 - Je vais a l'angle de deux rues et je compte les passant à une heure de changement de cours. Je calcule le ratio et je vérifie que c'est le même dans la simulation.
- Expert: les routes sont-elles cohérentes?

```
Console X
road nb: 2.511370377694285%
--- new total:5057
road Rue du Château nb: 3.8560411311053984%
road Rue de la Guyonnerie nb: 3.6187462922681433%
road Rue du Château nb: 3.6187462922681433%
road nb: 3.2430294641091555%
road nb: 3.2430294641091555%
road nb: 3.2430294641091555%
road Chemin Rural de Petit Saclay à la Pacaterie nb
road nb: 2.570694087403599%
road Allée des Découvertes nb: 2.570694087403599%
road nb: 2.511370377694285%
--- new total:5057
```



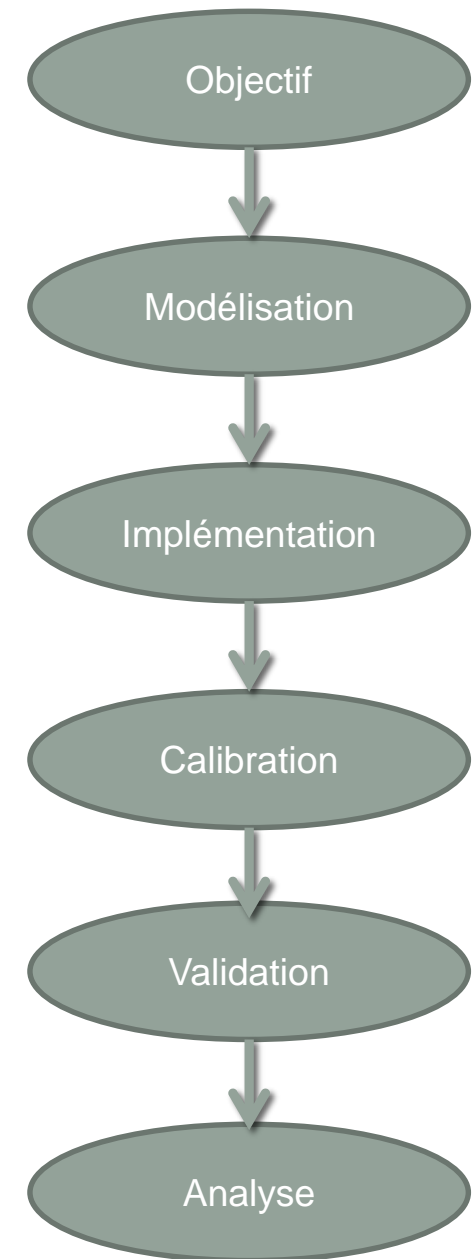


```

--- new total:3780
road Chemin Treuille de Beaulieu nb: 1.349206349206
road nb: 1.111111111111112%
road nb: 1.111111111111112%
road nb: 1.0582010582010581%
road Rue de Verdun nb: 1.0317460317460319%
road nb: 1.0317460317460319%
road Rue de Verdun nb: 1.0317460317460319%
road Rue de Verdun nb: 1.0317460317460319%
road Sentier de l'École Polytechnique nb: 1.0052910
road Sentier de l'École Polytechnique nb: 1.0052910
    
```

Etape 6: Analyse

- Analyse de stabilité: suppression aléatoire de 50% des batiments
- Effet d'une suppression de route?
- Effet de l'ajout d'un bâtiment avec une forte probabilité de présence (gare)
- Trajet vers une gare?



Analyse normative

- Quel emplacement choisir comme gare?

```

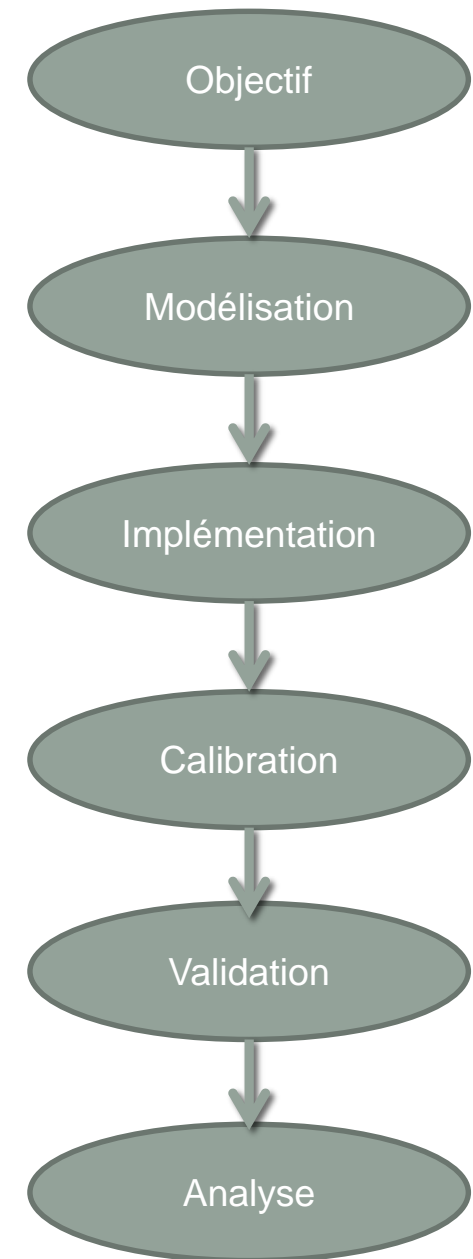
}
experiment Optimization type: batch repeat: 1 keep_seed: true until: ( time > 200 )
{
  parameter "GareX:" var: garex min: 0 max: 6000 step: 100;
  parameter "GareY:" var: garey min: 0 max: 6000 step: 100;
  method tabu minimize: totalnb iter_max: 30 tabu_list_size: 3;
  permanent {
    display Batchgraph background: rgb('white') refresh_every: 1 {
      chart "batchpos" type: scatter {
        data "posgare" value: {garex,garey};
      }
    }
    display batchval background: rgb('white') refresh_every: 1 {
      chart "batchval" type: series {
        data "posgare" value: totalnb;
      }
    }
  }
}
}

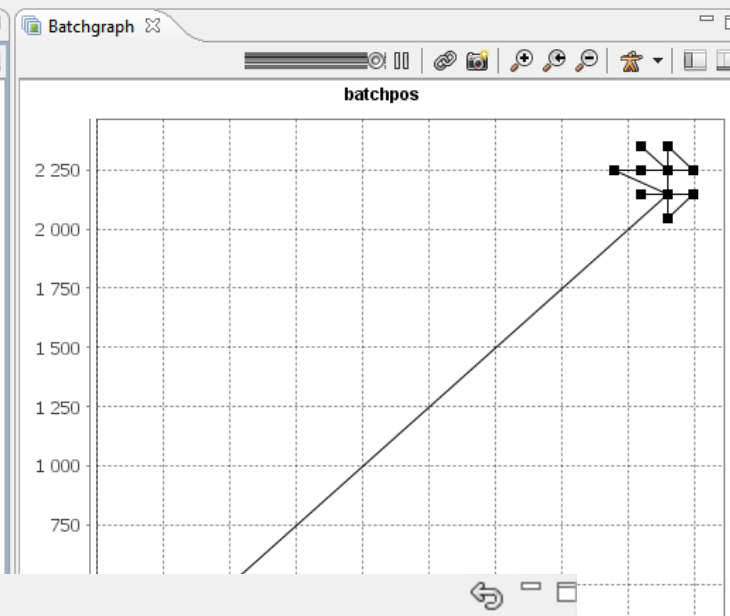
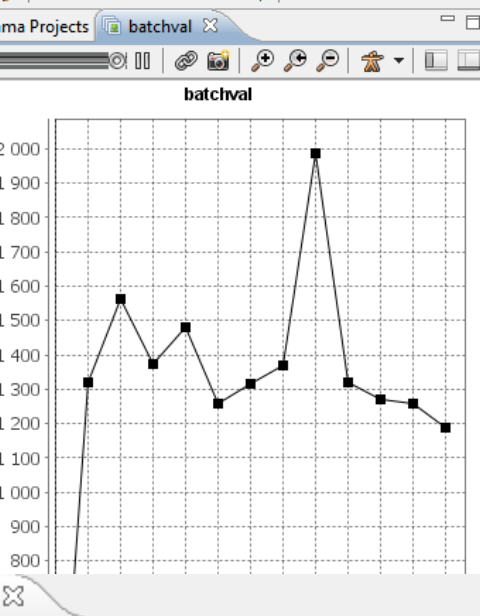
```

```

state waiting initial: true
{
  transition to: moving when: (current_hour = start_work)
  {
    building bt <- one_of(bat_univ);
    the_target <- any_location_in(bt);
    location <- the_target;
    the_target <- {garex,garey,0};
    currentpath <- path_between(the_graph, location, the_target);
  }
}

```





Errors Parameters

Model saclay_model Parameters for experiment 'Optimization'

Random number generator 'mersenne' among [cellular, xor, java, mersenne]

Default random seed Not defined 0.211345298100153

Exploration method

Stop condition time > 200

Best fitness 1259.0 with {garey=2247.48365, garex=1800.0}

Last fitness 1188.0 with {garey=2347.48365, garex=1800.0}

Parameter space garex (61) * ... (61) = 3721

Exploration method Method tabu | fitness = ...lations for each solution

Tabu list size 3

Maximum number of iterations 30.0

Parameters to explore

GareX: 2047.48365 between 0 and 6000 every 100

GareY: 2347.48365 between 0 and 6000 every 100

Model Parameters for experiment 'Optimization_hc'

Random number generator 'mersenne' among [cellular, xor, java, mersenne]

Default random seed Not defined 0.803559101801504

Exploration method

Stop condition time > 200

Best fitness 1241.0 with {garey=600.0, garex=1800.0}

Last fitness 1491.0 with {garey=400.0, garex=1800.0}

Parameter space garex (31) * garey (31) = 961

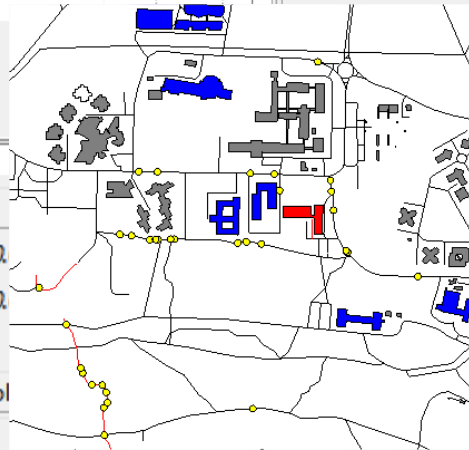
Exploration method Method hill_climbing | fitness = mini...age of 1 simulations for each solution

Maximum number of iterations 50

Parameters to explore

GareX: between 0 and 6000 every 200

GareY: between 0 and 6000 every 200



Shapefile for the buildings: C:\dev\GAMA\WP8\te

Shapefile for the roads: C:\dev\GAMA\WP8\te

Shapefile for the bounds: C:\dev\GAMA\WP8\te

People

Number of people agents 100

Model saclay_model Parameters for experiment 'saclay'

Random number generator 'mersenne' among [cellular, xor, java, mersenne]

Default random seed Not defined 0.660472665618968

Agent Inspector

Agent 'Maison de l'Ingénieur'

name 'Maison de l'Ing' (u00E9)nieur'

host saclay_model0

shape polygon ([[1655.8100000000268, 678.4200000036508, 0.0], [1655.8100000000268, 678.4200000036508, 0.0], [1714.86000000000151, 684.0150000011993], [1655.8100000000268, 678.4200000036508, 0.0], [1655.8100000000268, 678.4200000036508, 0.0]])

location x 1714.86000000000151 y 684.0150000011993

type 'yes'

- ❖ Le choix de la méthode d'exploration se fait à travers l'instruction **method**
- ❖ Par défaut, si aucune instruction **method** n'est définie, la méthode utilisée est une exploration exhaustive (test de toutes les combinaisons de valeurs de paramètres possibles)
- ❖ Utilisation de l'instruction **method** : **method** nom_methode +
 - **maximize** : optionnelle, expression à maximiser (*fitness*)
 - **minimize** : optionnelle, expression à minimiser (*fitness*)
 } L'un ou l'autre
 - **aggregation** : optionnelle, façon dans sont agrégées les résultats de plusieurs simulations pour le calcul de la *fitness* globale d'un jeu de paramètres (*min* ou *max*). Si cette facette n'est pas définie, utilisation de la moyenne.
 - Paramètres de la méthode d'exploration (dépend du type de méthode)
- ❖ Méthodes d'exploration existantes (pour plus de détails sur ces méthodes et leurs paramètres, voir : <http://code.google.com/p/gama-platform/wiki/GAMA?tm=6>)
 - **exhaustive** : exploration exhaustive de l'espace
 - **hill_climbing** : recherche locale gloutonne
 - **annealing** : recuit simulé
 - **tabu** : Recherche locale taboue
 - **reactive_tabu** : Recherche locale réactive
 - **genetic** : Algorithme génétique

```

experiment mon_experimentation type: batch {
  parameter 'Valeur de ma_variable1' var: ma_variable1 min: 10 max: 30 step: 1;
  parameter 'Valeur de ma_variable2' var: ma_variable2 among: [1, 5, 10];
  method tabu minimize: ma_fitness iter_max: 10 tabu_list_size: 3;
}

```

Exporter les résultats

- `reflex write_results {`
- `save [time,nbpeopletot, garex,garey] type: "csv" to: "result.csv";`
- `}`

Des valeurs peuvent être exportées en CSV

L'exportation est surtout utile a la fin d'un plan d'experience