

Avril 2015 – Septembre 2015

Littoral, Environnement, Télédétection, Géomatique  
Géomer - Plouzané



# Analyse spatiale dynamique par Simulation Multi-Agents, application à la viticulture en Anjou

*Céline Le Coq* –

*Licence Professionnelle SIG  
Université de LA ROCHELLE*

*Encadrement : Cyril Tissot, Mathias Rouan*

*Tuteur universitaire : Frédéric Rousseaux*



**ADVICLIM**





# SOMMAIRE

Remerciements .....	6
Remarques .....	6
<b>INTRODUCTION.....</b>	<b>7</b>
<b>I. MISE EN PLACE DU PROJET .....</b>	<b>8</b>
<b>I.1. Contexte du projet .....</b>	<b>8</b>
I.1.1. Historique .....	8
I.1.2. Acteurs et organismes.....	10
I.1.3. Objectifs .....	10
<b>I.2. Modélisation existante sous NetLogo .....</b>	<b>11</b>
I.2.1. Pourquoi modéliser la vigne ? .....	11
I.2.2. L'environnement NetLogo .....	11
<b>I.3. Objectifs du stage .....</b>	<b>12</b>
I.3.1. Similarités et différences entre GAMA et NetLogo .....	12
I.3.2. Etudes préalables : TERVICLIM et TERADCLIM .....	13
Elaboration d'une base de connaissances .....	13
Profil climatique .....	14
I.3.3. Enjeux et problématique .....	15
I.3.4. Fonctionnement général du modèle NetLogo .....	15
Un ensemble d'agents en interaction .....	15
A la base : le cycle de croissance de la vigne .....	15
Indice de degrés-jours : .....	17
Indice de Huglin : .....	17
Calendrier .....	17
Composante climatique.....	18
Gestion fine des paramètres environnementaux affectant la vigne.....	18
Résultats obtenus.....	19
I.3.5. Diagramme UML - modèle NetLogo .....	19
I.3.6. Limites et perspectives du modèle NetLogo.....	20
I.3.7. Apport de l'analyse multi-échelles pour ADVICLIM.....	20
I.3.8. Résultats attendus .....	20

I.4.	Zone d'étude.....	21
I.5.	Gestion de projet .....	22
I.5.1.	Logiciels .....	22
I.5.2.	Données.....	23
I.5.3.	Organisation au cours du stage.....	23
<b>II.</b>	<b>MODELISATION.....</b>	<b>24</b>
II.1.	Spécificité de la modélisation sur la plateforme GAMA.....	24
II.1.1.	Présentation de la plateforme .....	24
II.1.2.	Hiérarchisation et langage .....	25
II.1.3.	Schématisation du modèle .....	27
II.2.	Connexion à la base de données .....	27
II.2.1.	Interopérabilité logiciels : GAMA / PostGres-PostGIS .....	27
II.2.2.	Requêtes à la base de données .....	29
II.2.3.	Transfert des données par PostGres.....	29
II.2.4.	Utilisation de deux modèles .....	30
II.3.	La gestion de l'information géographique sous GAMA .....	31
II.3.1.	Définition de l'enveloppe de la couche.....	31
II.3.2.	De la couche d'informations à la formalisation agent.....	31
II.3.3.	Attribution des capteurs par parcelles .....	31
II.4.	Modèle de base : croissance de la vigne.....	33
II.4.1.	Gestion du calendrier sous GAMA.....	33
II.4.2.	Les indices bioclimatiques.....	34
	Une évolution dépendante des températures .....	34
	Deux compteurs opérationnels en parallèle.....	35
II.4.3.	Stades phénologiques de la vigne et évolution .....	36
II.5.	Modélisation multi-échelles.....	38
II.5.1.	Agrégation des parcelles à forte potentialité .....	38
<b>III.</b>	<b>ANALYSE CRITIQUE.....</b>	<b>41</b>
III.1.	Difficultés rencontrées.....	41
III.1.1.	Difficultés liées au domaine d'investigation .....	41
III.1.2.	Difficultés inhérentes aux limites du logiciels.....	41
III.2.	Limites.....	41
III.2.1.	Des données absentes .....	41
III.2.2.	Pondération de l'algorithme du plus proche voisin .....	43

Topographie.....	43
Environnement direct.....	43
III.2.3. Un climat méditerranéen à l'origine du modèle de disponibilité en eau WaLIS ....	44
III.2.4. Un temps de calcul qui rend peu exploitable le modèle auprès des viticulteurs ...	45
III.2.5. Des données brutes nécessitant un prétraitement chronophage.....	45
<b>III.3. Bilan et perspectives .....</b>	<b>45</b>
III.3.1. La disponibilité en eau, facteur limitant de la croissance .....	45
III.3.2. Agrégats spatiaux et niveaux d'échelles.....	47
Un paramétrage plus fin des conditions d'environnement.....	47
Une modélisation explorable à plusieurs niveaux .....	47
III.3.3. Utilité d'étendre le modèle à l'Anjou .....	48
III.3.4. Interactions parcelles – INAO : composante socio-économique .....	49
III.3.5. Simplifier pour mieux modéliser, des horizons temporels plus lointains.....	50
III.3.6. Bilan personnel .....	50
<b>III.4. Conclusion.....</b>	<b>51</b>
<b>RESSOURCES ET BIBLIOGRAPHIE.....</b>	<b>52</b>
<b>TABLE DES ILLUSTRATIONS.....</b>	<b>54</b>
Table des tableaux .....	54
Table des figures.....	54
Table des cartes.....	55
<b>ANNEXES.....</b>	<b>55</b>
Annexe I : Organisation du stage.....	56
Annexe II : Code complet.....	60
Annexe III : Exemple d'une année agronomique .....	69
Annexe IV : Commandes SQL .....	72
Boucle de récupération de la température.....	72
Capteur le plus proche.....	73
Agrégation en fonction d'un critère .....	73

## Remerciements

Je tiens dans un premier temps à remercier mes maîtres de stage Cyril Tissot et Mathias Rouan, et mon tuteur universitaire Frédéric Rousseaux, de m'avoir laissé l'opportunité de mener à bien ce projet et de m'avoir garanti un encadrement sans faille tout au long du stage. Je remercie également chaleureusement l'ensemble du LETG pour son accueil et notamment Hervé Quéno, initiateur du projet ADVICLIM.

L'équipe du LETG Géomer a su m'accompagner au cours de ce stage, un merci tout particulier à Annalisa Minelli et Yuji Kato sans qui les heures de développement auraient été beaucoup plus longues et également à l'ensemble du laboratoire qui a fait rimer ambiance de travail et convivialité.

La réactivité et l'engagement de l'équipe de GAMA reste un atout majeur dans la conduite de ce stage, merci encore à mes principaux interlocuteurs, Patrick Taillandier, Srirama Bhamidipati et Philippe Caillou.

Merci à mes graphistes en herbe Youenn et Malwenn pour leurs précieux conseils et à mes relectrices attentives Eléonore, Anaëlle et Mélinée.

## Remarques

La mise en page des documents cartographiques de ce rapport est adaptée à son format du rapport. Pour la plupart, les cartes ne comportent ni titre ni source, complètement illisibles et perturbant la lecture. Elles sont disponibles en annexes de la version PDF de ce document.

# INTRODUCTION

---

Si la communauté scientifique s'interroge massivement sur les effets du changement climatique à l'échelle planétaire, les initiatives de recherche abordant les conséquences des variations climatiques aux échelles fines restent marginales. Dans le contexte de la viticulture cette question est cruciale dans la mesure où la qualité du vin, le choix des cépages et la spécificité des terroirs dépendent de caractéristiques locales (Morlat *et al.*, 2001, Van Leeuwen *et al.*, 2008). Certaines études prédisent d'ores et déjà un glissement septentrional des zones de culture adaptées à la viticulture (Hannah *et al.*, 2013).

Il en résulte une forte demande de la profession viticole qui doit sans cesse s'adapter à des conditions locales modifiées. Fort de ce constat, il apparaît pertinent de développer une démarche de modélisation permettant de simuler l'impact des différents scénarios sur la dynamique de la vigne et les capacités d'adaptation des viticulteurs au changement climatique. Les travaux de stage exposés dans ce rapport s'inscrivent dans cette perspective en proposant une approche de modélisation permettant de simuler les activités viticoles dans un contexte de changement climatique.

La démarche proposée vise à lier les observations climatiques et agronomiques menées sur le terrain (Bonney *et al.*, 2009, Neethling *et al.*, 2012) et les questionnements liés à la recherche d'une adaptation optimale de la vigne à la spécificité des terroirs viticoles (Barbeau *et al.*, 2001 ; Asselin *et al.*, 2003). Elle mobilise des technologies relevant de la géomatique et des modèles multi-agents (Tissot, 2003, Tissot *et al.*, 2012).

La problématique de ce stage est alimentée par deux actions de recherche complémentaires :

- La première vise à proposer une méthodologie de modélisation capable d'intégrer les éléments complexes constituant le fonctionnement d'un terroir viticole dans un environnement de modélisation multi-agents
- La seconde vise à déterminer des zones d'aptitude agro-climatique au regard de l'évolution des conditions climatiques et des caractéristiques agro-pédologiques des terroirs actuels

Développé dans un contexte européen, le projet ADVICLIM sera tout d'abord explicité. Les réponses à la problématique en termes de développement dans un environnement de modélisation seront ensuite examinées puis critiquées. Pour conclure plusieurs perspectives de recherche seront proposées.

# I. MISE EN PLACE DU PROJET

## I.1. Contexte du projet

### I.1.1. Historique

Le projet dans sa continuité s'articule autour de trois entités de différents niveaux administratifs et différents stades temporels (cf. *Figure 1*). Dans un premier temps montés autour de financement nationaux, le projet actuel est désormais supporté par des fonds européens.

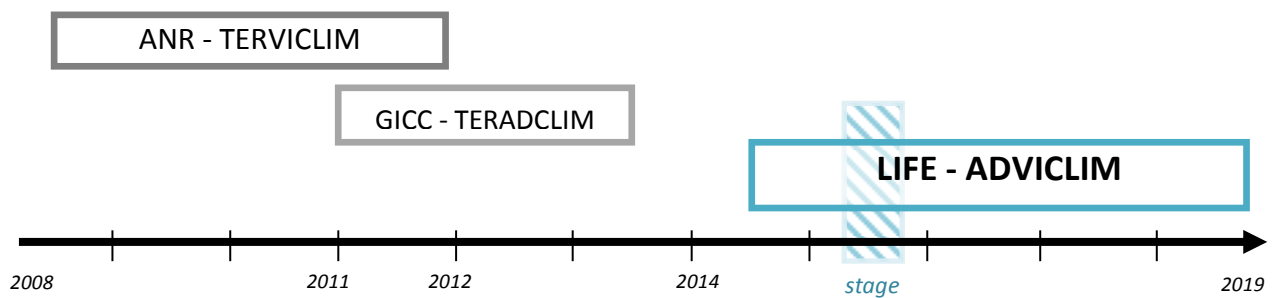
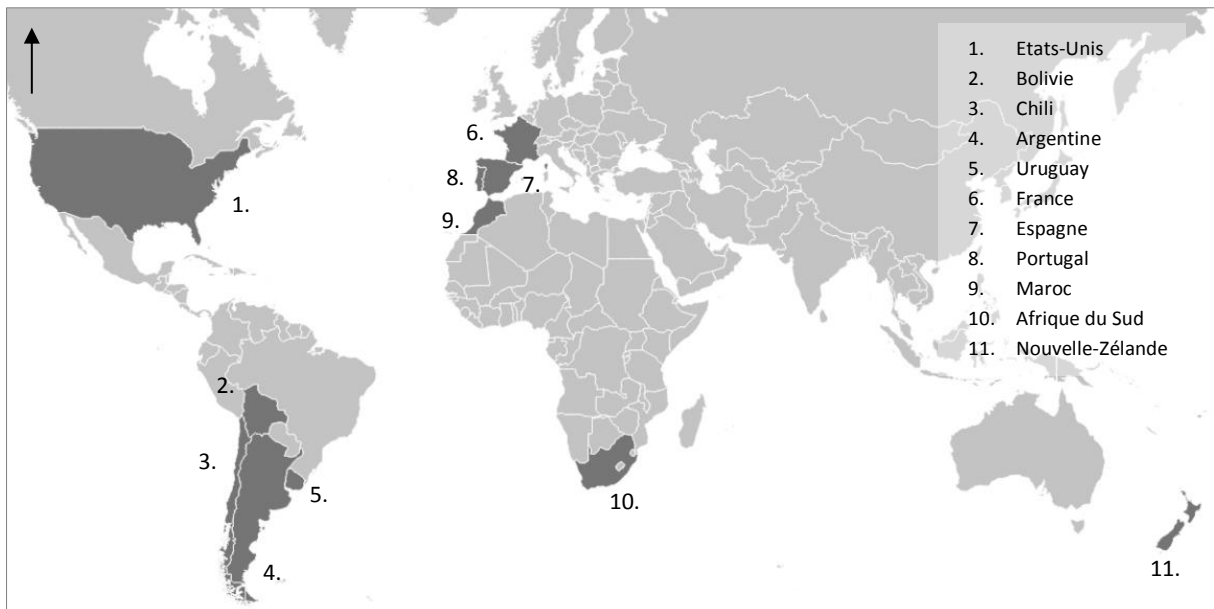


Figure 1: Evolution temporelle du projet et contexte du stage, de 2008 à 2019

De 2008 à 2012 le projet TERVICLIM<sup>1</sup> est ainsi supporté par une ANR (Agence Nationale de la Recherche), puis TERADCLIM<sup>2</sup> a pris le relais à travers un financement GICC (Gestion et Impact du Changement Climatique), de 2011 à 2013. Onze pays ont pris part à cette initiative (cf. *Carte 1*).



Carte 1: Répartition mondiale des sites expérimentaux, 15 vignobles au total, 11 pays représentés (D'après TERADCLIM, 2011)

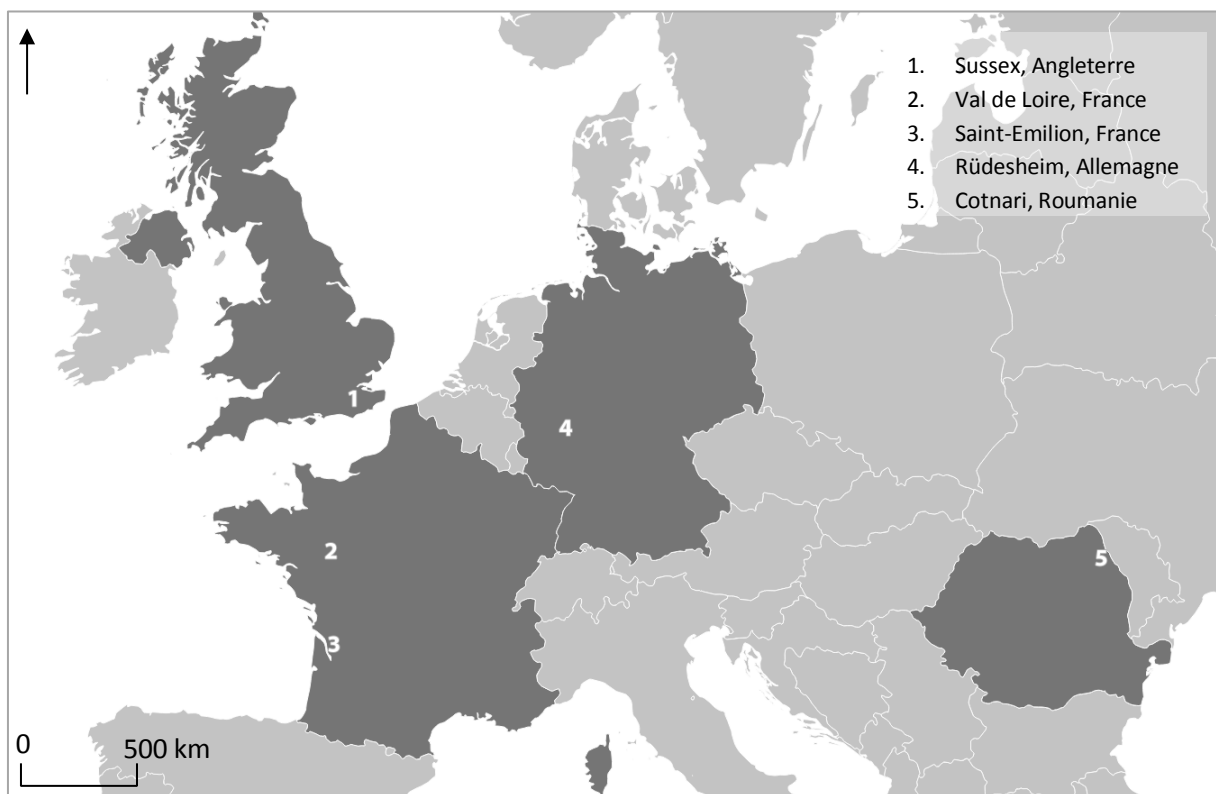
<sup>1</sup> « Observation et modélisation spatiale du climat à l'échelle des terroirs viticoles dans un contexte de changement climatique »

<sup>2</sup> « Adaptation au changement climatique à l'échelle des terroirs viticoles »



Il s'agit principalement d'obtenir dans un premier temps une modélisation spatiale du climat, notamment à une échelle plus fine que ce que le Modèle de Circulation Générale (MCG)<sup>3</sup> peut offrir, en intégrant les scénarios du GIEC<sup>4</sup>. De nombreuses stations météorologiques spécifiques ont ainsi été mises en place, en fonction des caractéristiques de surface des zones cibles (pente, exposition etc.). La modélisation des activités viticoles, en lien avec la variabilité climatique, est abordée dans le cadre du programme TERADCLIM avec deux premiers prototypes développés dans l'environnement NetLogo (sur le site de Mendoza en Argentine et dans la zone d'appellation Quart de Chaume en Anjou). Le programme LIFE<sup>5</sup> ADVICLIM (*Adaptation of Viticulture to CLIMate change : High resolution observations of adaptation scenario for viticulture*) mobilise ces méthodologies à l'échelle européenne avec un objectif de diffusion des résultats de la recherche auprès de la profession viticole.

ADVICLIM se compose de 3 volets : 1) climat et sa modélisation, 2) phénologie de la vigne et modélisation des activités viticoles (itinéraires agronomiques et stratégies d'adaptation), 3) Partage des données au niveau mondial et auprès des viticulteurs. Six sites pilotes ont été sélectionnés à cette occasion (cf. Carte 2).



Carte 2: Répartition des 5 sites pilotes au niveau européen, déployés sur 4 pays.

<sup>3</sup> Le MGC (ou GCM pour les anglophones) est un modèle numérique complexe mettant en jeu les interactions des fluides du système climatique (atmosphère, océan). Il fait intervenir les notions de mécanique et de thermodynamique régissant les fluides géophysiques dans les 3 dimensions et dans le temps.

<sup>4</sup> Groupe d'experts Intergouvernemental sur l'Evolution du Climat dont la mission principale est d' « établir régulièrement une expertise collective scientifique sur le changement climatique »

<sup>5</sup> LIFE : L'Instrument Financier pour l'Environnement, programme de soutien financier européen.

Ainsi, au vu du caractère interdisciplinaire et international du projet, de nombreux acteurs gravitent et interagissent autour d'ADVICLIM.

## I.1.2. Acteurs et organismes

Coordonné par le laboratoire LETG-Rennes COSTEL (Climat et Occupation du Sol par TELédétection) de l'Université Rennes 2, le projet ADVICLIM possède huit autres partenaires, cinq français : le CNRS, l'ECS ECOCLIMASOL, les antennes INRA de Bordeaux et Beaucauzé et l'Institut Français de la Vigne et du Vin (IFV) ; un britannique, l'université de Plumpton ; un allemand, l'université Hochschule Geisenheim ; et un roumain, représenté par l'USAMV Iasi (University of Agricultural Sciences and Veterinary Medicine).

L'UMR LETG (Littoral, Environnement, Télédétection, Géomatique) du CNRS, regroupe cinq laboratoires dans le grand ouest, dont Costel et Géomer impliqués dans le programme ADVICLIM. C'est dans le cadre de cette collaboration que ce stage a pu voir le jour, au sein des locaux de l'IUEM (Institut Universitaire Européen de la Mer), une des antennes de l'Université de Bretagne Occidentale.

Le projet scientifique du LETG réunit divers professionnels de manière transversale autour d'un thème de recherche générique commun, concernant l'analyse et la modélisation des systèmes complexes à l'interface entre nature et société.

L'un des supports en terme de visibilité pour le laboratoire d'accueil Géomer est constitué par le portail INDIGEO<sup>6</sup> (Rouan, 2015), qui pourrait à terme être intégré dans le projet ADVICLIM en tant qu'espace de visualisation des résultats, pour les partenaires aussi bien que pour les acteurs de fond du projet : les viticulteurs.

## I.1.3. Objectifs

ADVICLIM a pour objectif de contribuer à améliorer la gestion locale des vignobles face aux changements climatiques. Les objectifs sont ainsi de mesurer et modéliser, non seulement les changements induits par le réchauffement climatique mais aussi son impact en viticulture. Ce sera à partir de ces résultats que pourront être imaginées les meilleures réponses en termes d'adaptation et de mitigation au sein des vignobles soumis aux changements climatiques.

La mise en place du projet peut ainsi se diviser en cinq grandes parties :

- La modélisation climatique à échelle locale des vignobles étudiés.
- La modélisation des activités viticoles sous contraintes d'environnement et la simulation des stratégies d'adaptations.
- Le développement d'une application visant à aider les viticulteurs à mesurer et réduire leurs émissions de CO<sub>2</sub> au sein des vignobles.
- La mise en place expérimentale de mesures de mitigation et d'adaptation sur les sites pilotes, à fin de réduire d'au moins 20 % les émissions de gaz à effet de serre.

---

<sup>6</sup> <http://indigeo.fr>

- L'intégration d'un « manuel de bonnes pratiques » relatives aux mesures de mitigation et d'adaptation pour les viticulteurs.

## I.2. Modélisation existante sous NetLogo

### I.2.1. Pourquoi modéliser la vigne ?

Si le traitement des informations récoltées dans le cadre d'ADVCLIM peut être réalisé à l'aide d'outils SIG classiques, l'analyse des interactions entre les composantes climatiques, physiques, biologiques et humaines nécessite de modéliser des interactions complexes et récursives.

Dans ce contexte, les environnements de modélisation à base d'agents s'avèrent particulièrement bien adaptés. Les systèmes étudiés sont décomposés et modélisés en termes d'agents, dont les interactions permettent l'émergence d'une dynamique globale (Taillandier *et al.*, 2014, Shanmuganathan *et al.*, 2008, Drogoul *et al.*, 2003 ).

Il existe ainsi trois catégories de plates-formes, en fonction du langage utilisé (Taillandier *et al.*, 2014):

- langage de programmation générique (Java, C++, Python) adaptées au développement de modèles complexes : nombreux agents et processus (*SWARM, Repast Symphony*),
- langage de modélisation dédié (*NetLogo, GAMA*) plus simples à utiliser,
- définition des modèles à l'aide d'un langage de modélisation uniquement graphique, limitées en termes de richesse des modèles qu'elles permettent de construire (*StarLogo TNG, MAGéo, AgentSheets*).

Les deux plates-formes utilisées pour le projet font partie de la deuxième catégorie. GAMA est celle qui a été privilégiée comme environnement de développement au cours du stage.

### I.2.2. L'environnement NetLogo

NetLogo est un environnement de modélisation dédié à la simulation de phénomènes naturels ou sociologiques. Logiciel libre et open-source, il s'utilise via le langage de programmation *Logo*, dérivé d'un langage de programmation fonctionnel (Lisp). Initialement conçu dans un but éducatif, il est également répandu dans le domaine de la recherche, principalement pour sa facilité d'utilisation, sa prise en main étant possible pour les non-initiés à la programmation.

Les possibilités de modélisation sont basées sur les interactions entre « agents » : l'agent est assimilé à une entité informatique capable d'agir sur elle-même et sur son environnement, ayant ses propres caractéristiques et comportements, qui réagit à ses transformations et possède une représentation partielle de cet environnement<sup>7</sup> (Ferber, 1995). Les agents sont ici appelés des « *turtles* ».

NetLogo bénéficie de plusieurs APIs (extensions), dont une spécialement dédiée à l'utilisation des SIG, permettant de gérer les systèmes de coordonnées, les jeux de données vecteurs ou rasters,

---

<sup>7</sup> On considère ainsi que le concept se rapproche de l'intelligence artificielle.

d'effectuer des analyses spatiales vecteurs et des opérations sur les rasters et enfin de dessiner objets ou grilles.

Au niveau technique, NetLogo fonctionne à travers une machine virtuelle JAVA (ce qui est également le cas pour GAMA), permettant de travailler sur la plupart des plateformes (Windows, Mac, Linux...). Le logiciel a été écrit en Scala et en JAVA<sup>8</sup>. Cette caractéristique permet également d'exporter les modèles en tant qu'applets JAVA, utilisables avec un navigateur classique. La version utilisée pour développer les modèles Quart-de-Chaume et Mendoza était la 5.1 ainsi que l'extension SIG associée.

## 1.3. Objectifs du stage

La période de stage s'est divisée en plusieurs phases, la première s'est constituée en une analyse du modèle sous NetLogo, exposée dans les points suivants, la seconde a été dédiée au développement d'un prototype simplifié du fonctionnement d'un système viticole, pour finir le modèle a été optimisé, en y intégrant la notion de multi-échelles.

### 1.3.1. Similarités et différences entre GAMA et NetLogo

L'un des objectifs de ce stage était de développer un modèle de simulation sous un logiciel plus adapté à la gestion de données spatiales : la plateforme de simulation multi-agents GAMA (cf. II.1). Le développement de ce prototype permettra d'évaluer la prévalence de la plateforme pour ce type de projet. En effet, la modélisation sous NetLogo a soulevé certaines limites, que l'utilisation de GAMA pourrait potentiellement combler. GAMA est employé à travers un langage orienté objet, tout comme NetLogo. On y retrouve donc en partie les mêmes composantes, comme présenté dans le *Tableau 1*.

*Tableau 1: Correspondance entre le vocabulaire de GAMA et celui de NetLogo (D'après Taillandier et al, 2014)*

GAML	NetLogo
Species	Breed
Micro-species	-
Parent-species	-
Child-species	-
Model	Model
Experiment	Observer
Agent	Turtle/observer
Attribute	'breed'-own
Action	Fonction globale appliquée sur un 'breed'
Behavior	Ensemble de fonctions globales appliquée sur un 'breed'
Aspect	Seulement un, associé au comportement
Skill	-
Statement	Primitive
Type	Type
Parametric type	-

<sup>8</sup> Les deux langages étant interoperables.

Au-delà des aspects techniques inhérents aux langages utilisés, on peut simplement remarquer que plusieurs fonctions n'existent pas dans NetLogo, ou ne sont que partiellement applicables.

Les « *species* » notamment, équivalent d'une classe en JAVA (ensemble de plus petites unités de bases ayant les mêmes caractéristiques globales), peuvent être organisées de manières beaucoup plus fines sous GAMA que sous NetLogo, avec notamment des micro-espèces, des parents, des enfants. Cette hiérarchisation plus complète, alliée à la mise en place de « *skills* » (capacités, compétences), pourraient à terme permettre de gérer plus facilement l'aspect multi-échelles du modèle ADVICLIM, en jouant sur les différents niveaux d'espèces.

En outre, la plateforme GAMA offre des performances accrues pour développer des modèles riches, intégrant de nombreux agents et données (Taillandier *et al.*, 2014). Le projet ayant pour ambition de passer de 54 parcelles viticoles à plus de 2000, cette perspective semble plus appropriée.

### I.3.2. Etudes préalables : TERVICLIM et TERADCLIM

#### *Elaboration d'une base de connaissances*

Les informations relatives aux pratiques des viticulteurs ne sont pas nécessairement récoltées de façons automatiques par les instituts de recherche agronomique ; ou dans une moindre mesure, pas à un tel niveau de précision. Une campagne d'enquêtes approfondies et indépendantes a donc été lancée auprès des viticulteurs afin de récolter un maximum de données. Le but étant de bien comprendre les interactions entre la variabilité climatique et les pratiques culturales, permettant de produire une description fine du déroulement des activités viticoles, dans un contexte de changement climatique (Tissot *et al.*, 2010, *Tableau 2*).

*Tableau 2: Caractéristiques des enquêtes approfondies et indépendantes réalisées auprès des viticulteurs (D'après Tissot et al, 2010)*

	Enquêtes 1	Enquêtes 2
<b>Objectifs</b>	Etude de la sensibilité des pratiques annuelles aux conditions climatiques	Etude de la capacité d'adaptation des viticulteurs
<b>Méthode</b>	Semi-dirigée	Semi-dirigée
<b>Questions et thèmes abordés</b>	<ul style="list-style-type: none"> <li>• Périodes de travail</li> <li>• Techniques et machines impliquées</li> <li>• Variables climatiques favorables et défavorables</li> </ul>	<ul style="list-style-type: none"> <li>• Evolution des pratiques</li> <li>• Facteurs menant aux changements</li> <li>• Conditions climatiques caractérisant « bons » et « mauvais » millésimes</li> <li>• Stratégies d'adaptations</li> </ul>

Au-delà du rapport des viticulteurs aux changements climatiques et de leurs techniques de travail, d'autres informations ont été récoltées via différentes sources, telles que le type d'entreprise, le profil de la production, le cépage<sup>9</sup> utilisé par parcelle, etc. L'ensemble des données est administré au sein d'une base de données PostGreSQL/PostGIS.

<sup>9</sup> Le cépage correspond au type de plants de vignes (ou cultivars), qui donneront différents types de vins. Les cépages les plus répandus sur notre zone d'étude sont le Chenin blanc, le Cabernet Franc et le Gamay.

## Profil climatique

L'intégration de simulations climatiques « classiques » (MGC, HadCM3<sup>10</sup> entre autres) s'avère complexe à l'échelle des terroirs viticoles. Cette difficulté résulte d'une incompatibilité des échelles de restitution de ces modélisations avec les spécificités des sites d'appellation comme le Quart de Chaume mais également de l'absence ou de l'imprécision de certaines données simulées (ETP, précipitations). Pour contourner cette limite, une méthode alternative a été mise en œuvre. Après avoir réalisé une classification des vingt dernières années en fonction de leurs profils climatiques (calcul des variables du climat et des indices bioclimatiques propres à la vigne), un ensemble de scénarii hypothétiques a été construit.

- Année chaude et sèche
- Année froide et sèche
- Année normale et sèche
- Année chaude et humide
- Année froide et humide
- Année normale et humide
- Année chaude et normale
- Année froide et normale
- Année normale

Les scénarios peuvent être appliqués de manière aléatoire, ou choisis par l'utilisateur. Le choix d'un scénario chaud et sec par exemple, se traduira au niveau de la modélisation par l'intégration majoritaire de données issues d'années chaudes et sèches (classées en fonction de la typologie) au cours de la période de simulation. En effet, les variations climatiques ont une forte incidence sur l'évolution et le cycle de vie de la vigne, qui peut être mis en évidence au regard de la durée et des dates de changement de stades de la vigne (cf.

Figure 2).

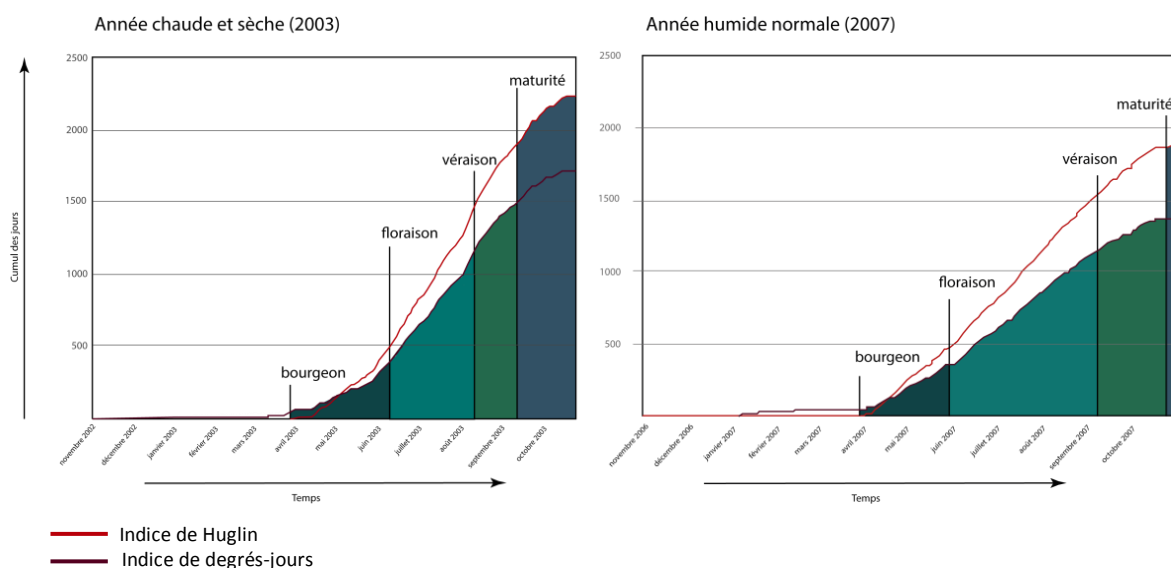


Figure 2: Exemples de courbes de profils climatiques sur deux années types et leurs incidences sur les principaux stades d'évolution de la vigne, en fonction des indices bioclimatiques de degrés-jours et de Huglin (D'après Tissot et al, 2015)

On peut remarquer que schématiquement, une année plutôt chaude et sèche (comme 2003) amènera plus vite la vigne à maturité; alors qu'une année plutôt humide (comme 2007) aura pour conséquence une maturation du raisin plus lente. Ces variations auront une influence au niveau des

<sup>10</sup> Le Hadley Centre Coupled Model, version 3 est une version plus récente de modèle général de circulation océan/atmosphère

actions agronomiques menées sur la vigne, notamment sur les vendanges. Le type d'actions réalisées sera également dépendant de la situation géographique de la vigne et de la typicité désirée.

### I.3.3. Enjeux et problématique

Afin d'évaluer l'impact de la variabilité des conditions climatiques sur les stratégies de production viticole, le modèle tend à mettre en relation les stades phénologiques de la vigne avec les conditions climatiques et les modes de conduites agronomiques, dans un contexte de changement global. La démarche repose sur la formalisation d'une relation cohérente entre un réseau d'agents réactifs (les viticulteurs, la vigne, l'INAO) et un terroir viticole, à travers une gamme de contraintes techniques, socio-économiques, réglementaires et environnementales.

La motivation était double :

- Restituer la dynamique de la vigne à partir des indices bioclimatiques
- Simuler les itinéraires agrotechniques dans un contexte de changement climatique

Tout l'enjeu de ce type de modélisation réside dans ses potentialités à intégrer des scénarii prospectifs. L'adaptation des itinéraires de conduites agronomiques et des stratégies de production constitue un élément clé pour la profession viticole : l'enjeu reste, en effet, de s'adapter au changement climatique, tout en préservant la typicité des terroirs et les caractéristiques des vins d'appellation, pierre angulaire de leurs succès économiques.

### I.3.4. Fonctionnement général du modèle NetLogo

Le modèle baptisé « SEVE » (*Simulating Environmental impact on Viticultural Ecosystems*) comporte plusieurs parties, interagissant les unes avec les autres.

#### *Un ensemble d'agents en interaction*

On dénombre dix agents en interaction les uns avec les autres :

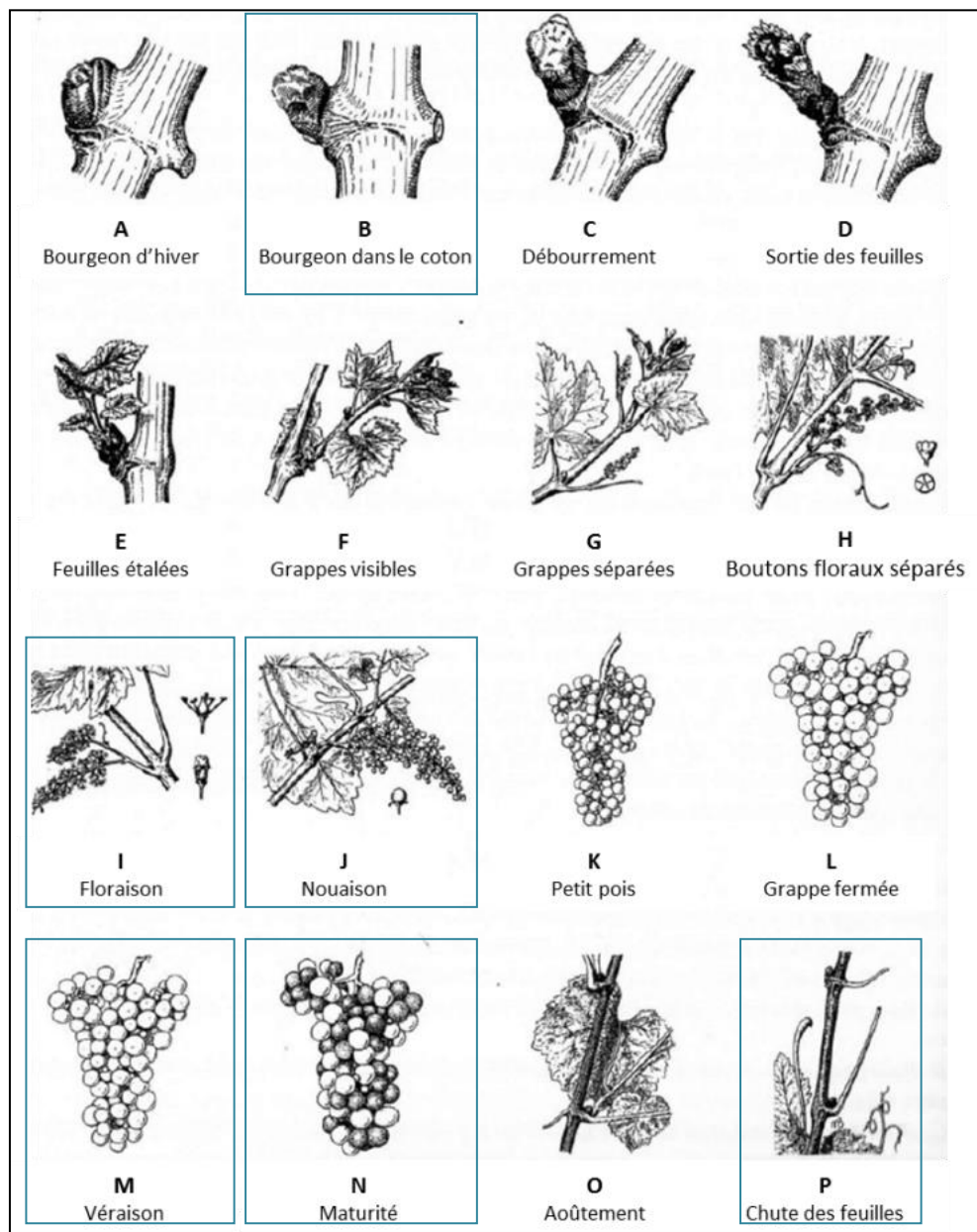
- Les viticulteurs (17 agents)
- Les parcelles (54 agents)
- Les vignes
- Les capteurs météorologiques (5 agents)
- Les actions (11 agents)
- Les outils d'adaptations (3 agents)
- Les ouvriers
- Les tracteurs
- Les années
- Les pathogènes (uniquement le mildiou pour le moment)

Les agents interagissent entre eux, amenant ou non d'autres agents à modifier leur comportement.

#### *A la base : le cycle de croissance de la vigne*

La base du système repose sur la simulation de la croissance de la vigne. La classification utilisée est celle de Baggiolini (Baggiolini, 1952) divisée en seize stades, nommés de A à P. Ils comportent tous des caractéristiques spécifiques (cf. *Figure 3*).

Leur intégration et leur utilisation au sein du modèle reposent sur leur relation avec deux indices bioclimatiques, l'indice de degrés-jours et l'indice de Huglin.



Stade clé, correspondant à un changement de couleur au niveau graphique dans le modèle

Figure 3: Les stades repères de la vigne (D'après Baggiolini, 1952)

Ces deux indices permettent d'évaluer les besoins en chaleur de la vigne pour le développement des différents stades phénologiques. Ils sont tous deux calculés différemment (cf. Figure 4), à des périodes de l'année différentes. L'indice de degrés-jours est calculé sur la période végétative du 1<sup>er</sup> novembre / 31 octobre et l'indice de Huglin débute lui au 1<sup>er</sup> avril. Comme on peut le remarquer dans le Tableau 3, l'indice de Huglin permet principalement de déterminer le passage au stade de maturité (code N).



L'indice de degrés-jours correspond plus précisément à la somme des températures moyennes journalières à partir de la base de 10°C. Cette température correspond effectivement au seuil à partir duquel la vigne est physiologiquement active et est donc capable d'entrer en phase de croissance. L'indice de Huglin, ou indice héliothermique, est lui basé sur la moyenne des températures et la température maximale, auxquelles est associé un coefficient de longueur du jour (k) qui varie en fonction de l'hémisphère (Huglin, 1978).

Les indices étant liés aux exigences thermiques des cépages et aux taux de sucre potentiels du raisin pour celui d'Huglin, on retrouvera des seuils spécifiques pour chacun d'entre eux. Ces valeurs théoriques sont également très dépendantes de la zone géographique, en France par exemple, la Syrah, en opposition au Chenin qui atteint une maturité théorique à 1 500, atteint elle son potentiel à partir d'un indice de Huglin de 2 100.

Au sein du modèle, le calcul est donc réalisé chaque « jour » (chaque pas de temps), pour chaque parcelle de vigne, afin de déterminer la croissance de la vigne. Deux compteurs sont en place, un pour chaque indice bioclimatique. A chaque passage de seuil, le stade évolue également, tant de manière théorique que graphique.

Tableau 3: Relations entre stades et indices bioclimatiques en fonction d'un cépage exemple, le Chenin, le plus répandu en Anjou

Cépage	Code du stade	Indice de degré-jours	Indice de Huglin
Chenin	A	0	-
	B	23	-
	C	45	-
	D	97	-
	E	149	-
	F	201	-
	G	252	-
	H	304	-
	I	356	-
	J	549	-
	K	742	-
	L	934	-
	M	1 127	-
	N	-	1 500
	O	-	-
P	-	-	

Indice de degrés-jours :

$$= \sum \frac{T \text{ max} + T \text{ min}}{2} - 10$$

a.

Indice de Huglin :

$$= \sum \frac{(T \text{ moy} - 10) + (T \text{ max} - 10)}{2} \times k$$

b.

Figure 4: a. Formule permettant d'obtenir le cumul de degrés-jours b. Formule permettant d'obtenir l'indice de Huglin. Avec T max : température maximale, T min : température minimale, T moy : température moyenne k : coefficient de longueur du jour.

### Calendrier

Il n'existait pas, au moment du développement du modèle SEVE, d'extension<sup>11</sup> permettant de gérer les données en intégrant une composante temporelle définie, type calendrier. Les informations transmises par Météo France et l'INRA étant journalières et datées, il était pourtant nécessaire d'intégrer cette composante au code. L'option choisie a été de « recréer » un calendrier sous NetLogo, en utilisant le principe des jours juliens. Il s'agit en fait d'un compteur, lancé à chaque début d'année de simulation, permettant d'associer un nombre à chaque date. C'est ce nombre qui est alors utilisé lors des traitements et des calculs.

<sup>11</sup> Cette extension a désormais été développée par NetLogo et est actuellement en phase d'intégration dans le modèle.

### *Composante climatique*

Comme abordé en 1.3.2, les données météorologiques connues ont préalablement été traitées et classées, aboutissant à neuf profils climatiques représentatifs de neuf évolutions potentielles futures.

L'intégration de ces profils au sein du modèle de croissance de la vigne modifie ainsi les dates auxquelles les stades sont atteints pour chaque parcelle de vigne, les indices étant basés sur des variables thermiques.

Les données de températures et de précipitations sont utilisées pour calculer des jours agronomiquement disponibles pour la réalisation des tâches de travail de la vigne (Barbeau, 2011). Les agents viticulteurs sont informés des conditions météorologiques à + 4 jours. En fonction de ces données, de son profil de production et des caractéristiques agro-climatiques de ses parcelles (nature du sol, pente, exposition, bilan hydrique...), chaque Agent Viticulteur réalise des choix de conduite agronomique. A titre d'exemple, certains traitements phytosanitaires sont contraints par la pluviométrie le jour de l'application (produits systémiques), voir les jours suivants (produits de contacts) ; il en est de même pour le vent, qui, s'il est supérieur à 19 km/h, contraint l'utilisation de pesticides (interdiction réglementaire).

### *Gestion fine des paramètres environnementaux affectant la vigne*

Outre la composante climatique, d'autres variables peuvent affecter le comportement de la vigne. Les réponses à ces perturbations peuvent varier en fonctions de plusieurs paramètres :

- **Le type d'exploitation (conventionnelle, raisonnée ou biologique)**, influencera sur le mode de conduite agronomique du viticulteur, la présence de pathogènes (agent à proprement parlé) ne sera par exemple pas traitée de la même façon (produit de contact en exploitation « bio », produit systémique en conventionnel).
- **Le contexte socio-économique (nombre d'ouvriers, de tracteurs, surface à exploiter)**, pourra pondérer ou non la faisabilité d'une action : si le nombre d'ouvriers est insuffisant pour pouvoir effectuer une action, celle-ci pourra être repoussée, etc.
- **Les priorités fixées sur les vignes**, dépendent de l'avancement de la phénologie (retard ou précocité de la croissance dans un stade donné) et de la stratégie du viticulteur (mode de production).
- **Le type d'actions agronomiques**, dépendra des conditions climatiques, des priorités et du contexte socio-économique.
- **La nature du sol**, aura une influence sur la fraction d'eau transpirable du sol (FTSW), l'état hydrique du sol. Ce paramètre est géré par un modèle indépendant sous R<sup>12</sup>, calculant un bilan hydrique quotidien pour chaque parcelle.

Tous ces paramètres sont interdépendants : lors du lancement d'une action agronomique, le signal passera par une chaîne de booléens, déterminants si les conditions sont favorables ou non et donc si l'action agronomique pourra être entreprise ou non. C'est au final l'agent viticulteur qui transmettra la décision à l'agent vigne. L'interaction entre les deux agents est ainsi permanente.

---

<sup>12</sup> R est un logiciel libre de traitement de données et d'analyses statistiques. Le modèle utilisé par TERADCLIM permet d'obtenir des données réalistes de FTSW pour chaque parcelle en fonction des conditions environnementales. Il est couplé à NetLogo et effectue un calcul à chaque cycle du modèle. Il a été développé par Gabriel Daudin et Sébastien Roux de l'INRA Montpellier (Celette et al, 2010) et est basé sur la simulation WaLIS (Water balAnced for Intercropped Systems).



### I.3.6. Limites et perspectives du modèle NetLogo

Dans certains vignobles, le modèle a été utilisé afin de reconstituer des itinéraires agronomiques pour des années connues ou des séquences prospectives.

En termes de gestion des informations géographiques en tant que telle, l'unique possibilité offerte par NetLogo pour le type de données utilisées est l'affichage en grille. Cette solution est satisfaisante pour le modèle Quart-de-Chaume qui ne comporte que 54 parcelles, mais il est difficilement envisageable d'adopter le même fonctionnement à l'échelle du Layon (2 000 parcelles). En effet, chaque parcelle étant représentée par un ensemble de cellules, l'affichage à haute résolution est chronophage. Les tests effectués sans affichage graphique réduisent ainsi le temps de simulation de moitié (prototype Quart-de-Chaume). La gestion du multi-échelles, qu'elle soit graphique ou procédurale n'étant pas possible sous NetLogo, le développement d'un prototype dans l'environnement GAMA a été imaginé.

### I.3.7. Apport de l'analyse multi-échelles pour ADVICLIM

Le projet ADVICLIM dispose de différents jeux de données, à petites et grandes échelles (des observations de terrains aux simulations de modèles climatiques régionalisés). L'unité « parcelle » reste fondamentale, mais ne peut être exploitée à des niveaux d'agrégation supérieurs. A l'inverse, le réseau de stations météorologiques national peut difficilement mettre en évidence les variations microclimatiques à l'échelle d'un vignoble. Cette dualité entre les niveaux scalaires des données sources et les niveaux de représentations des données de simulation nécessite d'intégrer et de restituer les informations à plusieurs échelles. Cette démarche doit s'appuyer sur des règles d'agrégation multi-critères en fonction du type d'objet spatial à représenter.

### I.3.8. Résultats attendus

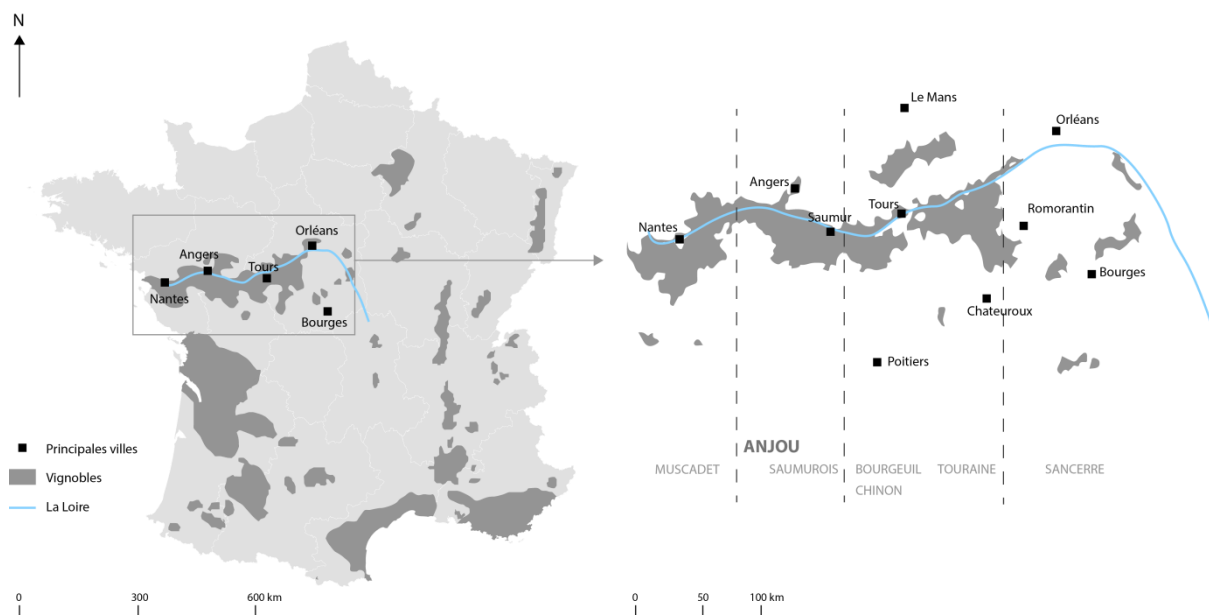
L'objectif est double :

- il s'agit d'une part de répondre à la demande croissante des professionnels de la viticulture qui doivent adapter leurs méthodes de production à la variabilité des conditions climatiques sans connaître l'horizon temporel de ces évolutions. Le développement de technologies permettant d'aborder dynamiquement et dans une dimension prospective cette problématique représente donc un fort enjeu à l'échelle nationale (maintien de la spécificité des terroirs viticoles) et internationale (identification de zones d'implantation en adéquation avec un cépage et des pratiques viticoles associées) ;
- il s'agit d'autre part de proposer une méthodologie de modélisation capable d'aborder une problématique complexe intégrant des contraintes spatio-temporelles multi-échelles et des méthodes de production basées sur des savoir-faire, parfois jalousement gardés, difficiles à mettre en équation.

En termes de livrables, le modèle en lui-même, rédigé en GAML, reste l'enjeu principal du stage. Celui-ci est commenté de manière précise afin de pouvoir être réutilisé sans difficulté. L'accent a également été mis sur son caractère généraliste, de nouvelles données devant pouvoir être intégrées en modifiant le moins possibles le script. Le rapport constitue également une trace importante de la stratégie adoptée, indispensable à la poursuite du projet.

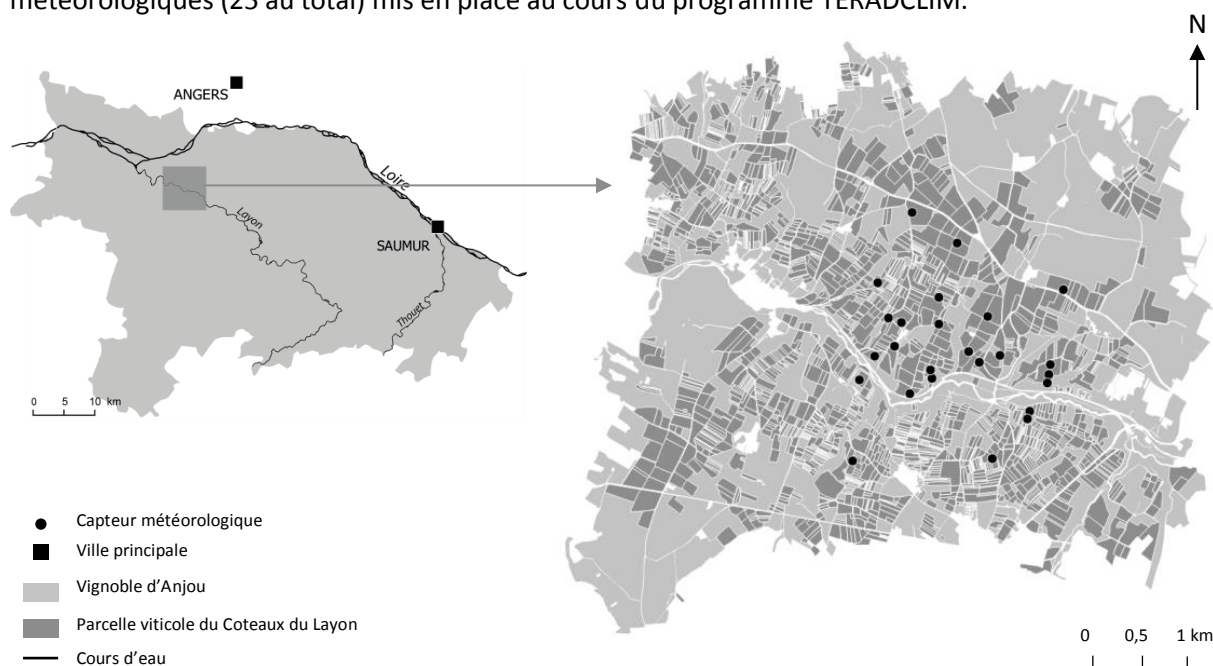
## I.4. Zone d'étude

Les parcelles tests se situent dans le département du Maine et Loire, au sein des vignobles du Val de Loire, constituant la limite septentrionale de la culture de la vigne (cf. *Carte 3*). Elles connaissent une importante variabilité climatique, dues à l'influence océanique à l'Ouest, qui s'atténue à l'Est, à travers la rencontre avec les bas plateaux du sud du bassin Parisien (Bonnefoy, 2013).



*Carte 3: Situation et caractéristiques des vignobles du Val de Loire au sein des vignobles français*

Ce sont plus particulièrement les zones de la moyenne vallée de Loire qui sont étudiées, caractérisées au point de vue climatique comme « océanique altérées » à « océanique dégradées » (Bonnefoy, 2013). Les manipulations de données ont dans un premier temps été effectuées sur un échantillon de la région viticole du Layon (cf. *Carte 4*), disposant d'un grand nombre de capteurs météorologiques (25 au total) mis en place au cours du programme TERADCLIM.



*Carte 4: Zone d'intérêt du Coteaux du Layon*

## I.5. Gestion de projet

### I.5.1. Logiciels

Tous les logiciels utilisés sont libres, pour certains, également open source (cf. Figure 7).

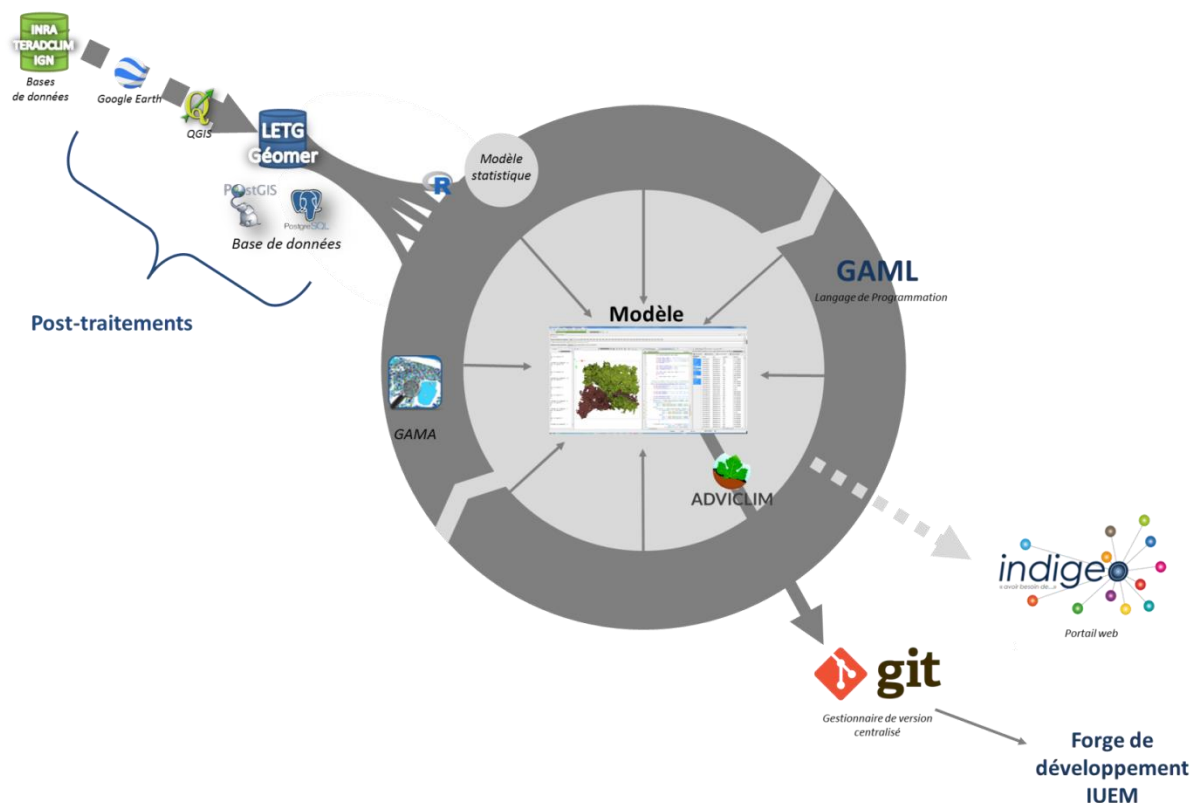


Figure 7: Représentation schématique de l'utilisation des différents logiciels mis en jeu et de leurs interactions

Les données sont transmises sous plusieurs formes, en tableurs ou en couches d'informations géographiques (.shp ou .kml). Il est donc nécessaire d'effectuer un certain nombre de pré-traitements, ou simplement de vérifier l'intégrité des données, via un panel de logiciels dédiés, allant de la gestion des fichiers de formes à l'intégration en base de données. Les informations ainsi stockées alimentent un couple de plateformes de modélisation tournant en parallèle, R, pour la partie bilan hydrique, et GAMA pour l'ensemble du modèle. A chaque étape de développement, le script du modèle ADVCLIM est envoyé à la forge de développement de l'IUEM (tucuxi) à travers le logiciel git. Grâce à ce gestionnaire de versions, il est possible à tout moment de retrouver une copie fonctionnelle du modèle précédemment développé. Comme évoqué en 1.1.2, l'export des résultats vers le portail IDG indigeo de Géomer est envisagé, même si ce dispositif n'est pas encore mis en place.

## 1.5.2. Données

Les données utilisées pour la réalisation du modèle proviennent de cinq sources principales :

- Des **programmes précédents** (TERVICLIM et TERADCLIM) ; il s'agit de données météorologiques indépendantes enregistrées par des capteurs *Tinytag*<sup>13</sup> et des stations météorologiques situées en Anjou. On y retrouve la date d'enregistrement, les températures minimales et maximales et le code du capteur.
- De **Météo France**; lorsque les données *Tinytag* sont indisponibles, ce sont les données générales qui sont utilisées. Pour la zone d'étude, il existe deux stations utilisables : celle de Beaulieu-sur-Layon et celle de Martigné-Briand (également site INRA).
- De l'**IGN**, pour le parcellaire, en format shape. A ce stade d'avancement, 6 163 parcelles sont traitées.
- Du **CTV** (Cellule « Terroirs Viticoles » soutenu par l'IFV) pour les données relatives à la nature du sol et au bilan hydrique.
- De l'**INRA** (Institut National de la Recherche Agronomique), qui fournit les données (viticulteurs, nature du sol, réserve en eau, potentiel de vigueur etc.) associées aux fichiers de l'IGN, mais aussi les couches d'information des zones d'appellation sur le territoire étudié, ainsi qu'une partie des données météorologiques.

Tableau 4 : Données utilisées pour la mise en place du modèle

Désignation	Source	Unité	Projection	Utilisation
<b>Températures</b> (minimales et maximales)	INRA Météo France	°C	-	Calcul des indices bioclimatiques
<b>Parcellaires viticoles</b>	BD TOPO® IGN	m	Lambert 93	Affichage graphique
<b>Capteurs</b>	INRA Météo France	m	Lambert 93	Affichage graphique, plus proche voisin
<b>Humidité</b>	INRA Météo France	%	-	Calcul des potentiels hydriques des parcelles
<b>Pluviométrie</b>	INRA Météo France	mm	-	
<b>Nature des sols</b>	CTV	Classe	-	
<b>Potentiel de vigueur</b>	INRA	% d'enherbement	-	
<b>Type d'entreprise</b>	TERADCLIM	Classe	-	Pondération des actions sur la vigne
<b>Type de cépages</b>	INRA	Classe	-	Pondération des indices bioclimatiques

## 1.5.3. Organisation au cours du stage

Le modèle SEVE étant déjà en place sous NetLogo, il était nécessaire dans un premier temps de le prendre en main et d'assimiler les différents éléments mis en jeu, afin de les transférer sur la plateforme GAMA. En outre, les objectifs et les données ayant évolués entre TERADCLIM et

<sup>13</sup> Résultant d'un abus de langage, *Tinytag* est en fait une marque de capteurs dédiés aux applications environnementales. Ils sont caractérisés par leur taille minimale et leur robustesse.

ADVICLIM, seules les données et les processus clés ont été implémentés sous GAMA. Cette simplification a permis de réduire le temps de développement du modèle tout comme le temps nécessaire aux différents tests. Ainsi, ce sont les données ayant trait au développement de la vigne et à leur environnement qui sont privilégiées. La phase de construction des agrégats spatiaux a ainsi pu être mise en exécution, malgré le laps de temps réduit. La complexité du modèle sera augmentée à la suite du stage, dans le but d'intégrer l'ensemble des paramètres agronomiques et de se rapprocher des besoins de la profession viticole.

Les bases de données TERADCLIM et ADVICLIM sont hébergées au sein du laboratoire LETG-Brest Géomer sur un serveur commun de l'IUEM. Une copie des bases a été créée en local, afin d'accélérer les vitesses de calcul lors des tests mais également de préserver l'intégrité des bases de données « sources ».

La structure et la hiérarchisation des tâches sont représentées à travers la *Figure 21* de l'*Annexe I*. Les diagrammes prévisionnel et effectif ainsi que la répartition horaire sont eux exposés respectivement en *Figure 22* et *Figure 23* de cette même annexe.

## II. MODELISATION

---

Après une phase de prise en main du modèle développé précédemment et de familiarisation avec les enjeux du projet, un prototype de modèle fonctionnant sous GAMA a été proposé.

### II.1. Spécificité de la modélisation sur la plateforme GAMA

#### II.1.1. Présentation de la plateforme

La plateforme GAMA (GIS & Agent-based Modelling Architecture<sup>14</sup>) est un environnement de développement et de modélisation open-source émanant principalement d'un collectif universitaire franco-vietnamien. Tout utilisateur peut proposer améliorations et extensions, qu'il soumet à la communauté<sup>15</sup>. Malgré une documentation parfois lacunaire, la plateforme est ainsi dotée de groupes d'utilisateurs très réactifs.

Entièrement développée en JAVA, elle s'appuie sur la machine virtuelle associée. L'utilisation des opérateurs spatiaux passe ainsi par la librairie JTS (Java Topology Suite) et respecte les standards de

---

<sup>14</sup> SIG et Architecture de Modélisation Multi-agent.

<sup>15</sup> Depuis juin 2015 la plateforme est passée sur GitHub, leader du service web d'hébergement et de gestion de développement de logiciels de façon collaborative.



l'OGC<sup>16</sup> au regard des spécifications SQL (Simple Features Specification for SQL) permettant d'interroger les bases de données, qui peuvent être géographiques.

D'un point de vue ergonomie la plateforme est divisée en plusieurs fenêtres, qui permettent de contrôler les différents paramètres du modèle (cf. *Figure 8*).

GAMA permet, entre autre, la rédaction de modèles en GAML, langage orienté objet dérivé du JAVA, dédié à la manipulation des objets de la plateforme. La définition d'un modèle s'organise en trois blocs : **1**) le bloc global, qui décrit les caractéristiques et les dynamiques du « monde » créé pendant la simulation, incluant la création d'autres agents ; **2**) le bloc comprenant les variables et les processus relatifs aux espèces, les archétypes d'agents avec leurs caractéristiques, leurs comportements et leurs représentations visuelles ; **3**) et enfin le bloc d'expérimentations, définis à travers les variables caractérisant l'état interne des agents. Ce dernier bloc correspond au cadre d'exécution du modèle.

## II.1.2. Hiérarchisation et langage

Comme évoqué dans le 1.3.1, GAMA possède l'avantage d'offrir un système de hiérarchisation élaboré. Les agents sont structurés en *species*, détenant les mêmes caractéristiques de base ; elles peuvent être spécialisées en *sub-species* évitant les redondances pour des *species* qui simuleraient des comportements similaires. En plus des *skills* associés (bouger, « dessiner », ou comme dans le modèle ADVICLIM, communiquer avec la base de données), des *reflex* ou des *actions* peuvent contrôler des comportements nécessaires à l'évolution du modèle. Les *reflex* sont constitués d'une suite de séquences de déclarations.

Par défaut un *reflex* se produira à chaque tour mais peut également être conditionné. Tout comme l'*action*, il peut concerner le « monde » global (par exemple, le *reflex* « calendrier » qui concerne l'ensemble du modèle ADVICLIM) ou une seule *species* (comme le *reflex* « evolve » qui ne s'appliquera qu'aux parcelles de vigne). Une *action* peut de son côté être déclarée de façon indifférente et appelée dans toute autre partie du modèle.

Les interactions entre agents se formalisent par l'intermédiaire de « questions » à travers l'opérateur *ask*. L'agent demandeur peut ainsi obtenir d'un autre un retour de données, un changement d'état, l'exécution d'une *action*, etc.

GAMA offre de nombreux opérateurs<sup>17</sup>, permettant d'exécuter des opérations complexes. En plus des opérateurs classiques (arithmétiques, manipulation des caractères etc.), le maniement des objets spatiaux est rendu possible grâce à des instructions spécifiques.

---

<sup>16</sup> Open Geospatial Consortium, garantissant l'interopérabilité des contenus, services et échanges liés à l'information géographique

<sup>17</sup> Description des opérateurs par catégories : [https://code.google.com/p/gama-platform/wiki/G\\_OperatorsAK](https://code.google.com/p/gama-platform/wiki/G_OperatorsAK)

### Contrôle du modèle

Contrôle de la vitesse de visualisation du modèle, rechargement et affichage d'informations relatives aux cycles.

### Paramétrages

Les paramètres définis par l'utilisateur peuvent être contrôlés à partir de cette fenêtre. Ici on peut choisir un type de scenario climatique

### Fenêtre graphique

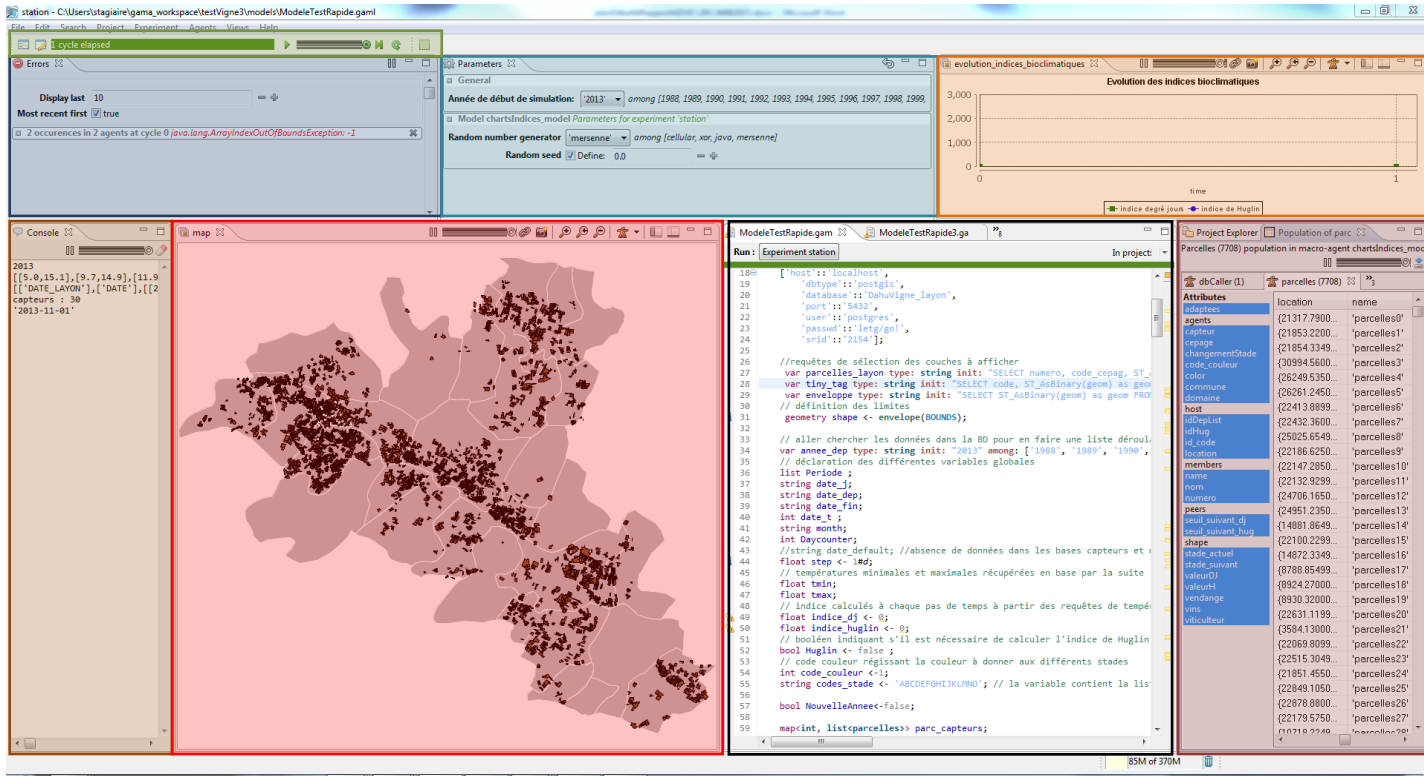
L'utilisateur a la possibilité d'ajouter un graphique relatif aux variables qui l'intéressent.

### Fenêtre d'erreurs

La plateforme reconnaît un certain nombre d'erreurs types qu'elle transmet à l'utilisateur. Les erreurs peuvent être critiques ou modérées (le modèle peut quand même tourner)

### Console

L'utilisateur a la possibilité de retourner certaines informations sous forme de texte à partir de l'instruction « write » dans le code.



### Fenêtre de visualisation des résultats

Elle évolue en fonction des interactions entre les agents. Plusieurs fenêtres peuvent être créées en fonction des besoins dans le bloc *experiment*.

### Fenêtre d'édition du code du modèle

Elle permet de définir les interactions entre objets, en GAML. Le modèle ne peut être lancé que si le code est construit correctement : pour cela un code couleur et des signets sont utilisés, indiquant la nature probable de l'erreur.

Figure 8: Ensemble des éléments composant la plateforme GAMA

### Inspecteur

Permet de visualiser en direct l'évolution des différentes variables dans une table.

### II.1.3. Schématisation du modèle

La *Figure 9* illustre de façon simplifiée les interactions entre les différents *reflex*. Après la phase d'initialisation (permettant de récupérer les informations dépendantes des choix utilisateurs à travers la base de données), puis l'instanciation des agents, l'affichage peut être généré. L'ensemble des listes de données est utilisé pour calculer les variables associées aux agents *capteurs* et *parcelles*. Les interactions entre agents sont régies par une chaîne de booléens, déclenchant ou non, le calcul d'indice de Huglin, le passage de la vigne au stade suivant, ou en dernier lieu l'export de couche(s) d'informations obtenue(s).

## II.2. Connexion à la base de données

La modélisation de la croissance de la vigne repose sur un système complexe d'interactions entre la température, la nature du sol, la disponibilité en eau et les actions agronomiques. Toutes ces données, qu'elles soient brutes ou calculées, proviennent d'une base de données PostGreSQL. Un accès rapide et optimisé à ce SGBD est donc fondamental pour pouvoir mettre en place l'ensemble des fonctions de calcul du modèle.

Malgré la faible performance du couplage entre GAMA et PostGreSQL (résultant d'une mauvaise gestion des requêtes SQL dans GAMA) cette solution technique a été retenue. En effet, les sources de données étant nombreuses et disparates, il paraissait plus judicieux de les stocker au sein d'une base bien administrée. De plus, l'objectif est de disposer d'un modèle générique permettant d'intégrer, sans modification profonde du script GAML, de nouveaux paramètres issus de la base de données.

### II.2.1. Interopérabilité logiciels : GAMA / PostGres-PostGIS

Du propre aveu des développeurs de GAMA, les connexions aux bases de données PostGresql/PostGIS sont un des points faibles de la plateforme en termes de performances. En effet, elles impliquent une ouverture puis une fermeture de connexion pour chaque requête envoyée et ce, sans mutualiser l'ouverture créée pour l'ensemble des données demandées.

Certains traitements faisant intervenir la totalité des quelques 2 000 parcelles de façon récurrente, le temps de calcul en est extrêmement allongé. Afin de contourner ce problème, une simplification de l'accès aux données a été mise en place, en réduisant au maximum le nombre de connexions. Ainsi, à chaque initialisation, en fonction des paramètres choisis par l'utilisateur, une requête sélectionne la totalité des données sur une année agronomique entière, c'est-à-dire du 1<sup>er</sup> novembre au 31 octobre de l'année suivante.

Par ailleurs, dans certaines circonstances, l'exploitation directe des fonctions offertes par la base de données PostGIS s'avère plus performante que l'utilisation des ressources logicielles de GAMA. Cette remarque vaut tout particulièrement pour des opérations spatiales, notamment lors du calcul des capteurs les plus proches (cf. II.3.3), et de l'agrégation des zones de potentialités (cf. II.5.1).

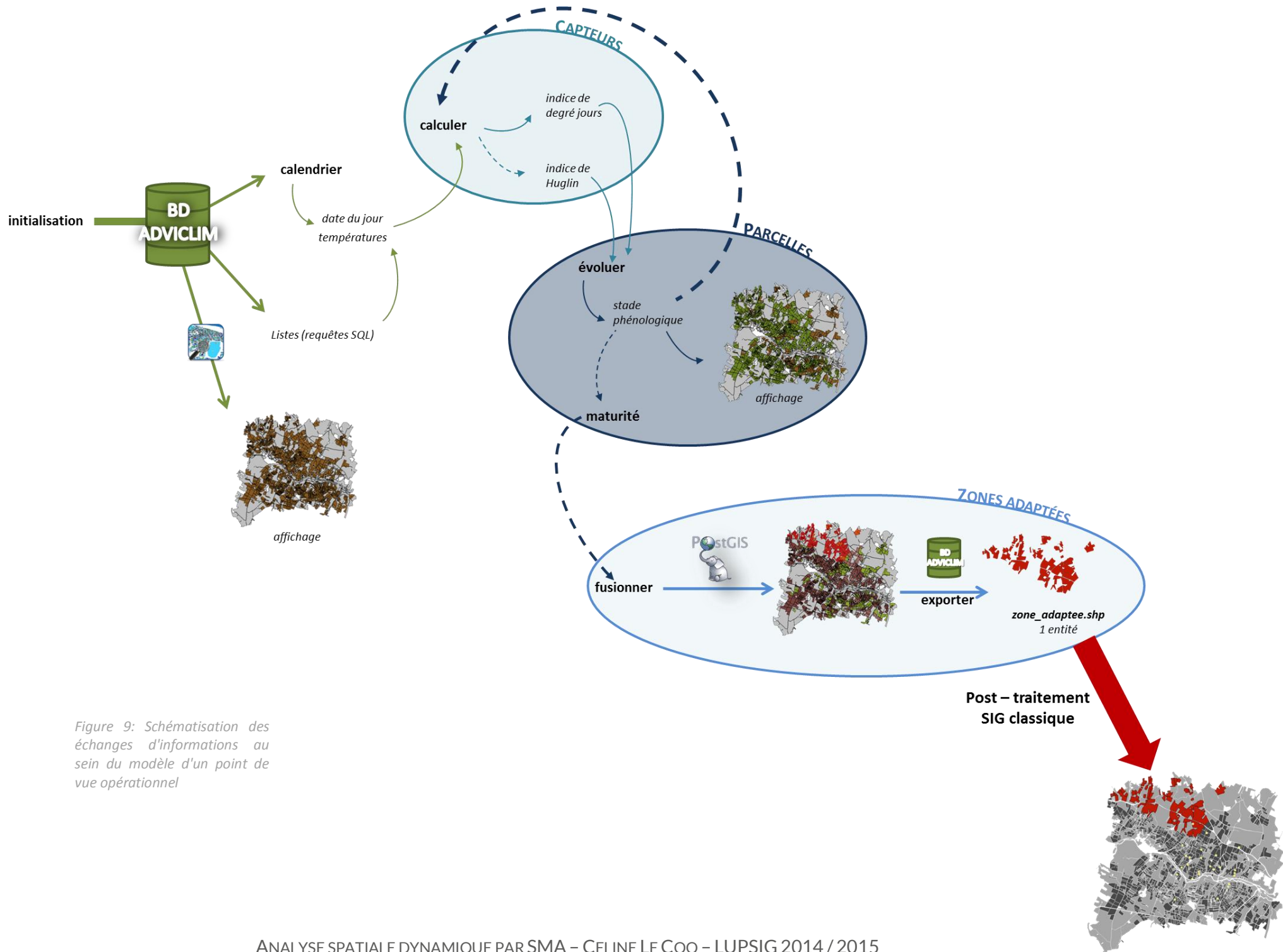


Figure 9: Schématisation des échanges d'informations au sein du modèle d'un point de vue opérationnel

## II.2.2. Requêtes à la base de données

Les paramètres de connexion sont définis comme une variable (PARAMS) dans le bloc global à l'initialisation (cf. Figure 10).

```
global {  
  
  var PARAMS type:map  
  //définition des paramètres de connexion à la base de données  
  init:  
    ['host'::'localhost',  
     'dbtype'::'postgis',  
     'database'::'DahuVigne_layon',  
     'port'::'5432',  
     'user'::'postgres',  
     'passwd'::'*****',  
     'srid'::'2154'];
```

Figure 10: Initialisation des paramètres des connexions à la base de données dans le modèle ADVICLIM sous GAMA

La base de données est ensuite définie comme une *species*, donc comme un agent à part entière, dans le *reflex init* du bloc global. Afin de pouvoir interagir avec la base de données, la *species* (qui répond ici au nom de *dbCaller*) est dotée de capacités spécifiques permettant d'intégrer des requêtes SQL (*SQLSKILL*). Cela permet par la suite de l'appeler à chaque requête. Ces appels ne peuvent se faire qu'à travers un *reflex* ou une *action*. Une fois la connexion générale établie, chaque requête est envoyée en définissant un certain nombre de paramètres. La Figure 11 illustre la traduction de l'envoi et la réception d'une requête en GAML.

Le texte de la requête SQL est déclaré comme une chaîne de caractères (1), la base de données est ensuite questionnée à travers la commande *ask* (2), comme n'importe quel autre agent. Les *SQLSKILL* lui permettent de traiter directement la requête en tant que telle (3). On retrouve ainsi les paramètres de connexion et le texte de la requête (4) et (1). Ici un seul élément est renvoyé, le nombre de capteurs utilisés ; l'information est donc directement extraite de la liste de listes transmise à GAMA et stockée dans une variable, *nbCapteurs* (5). Le procédé est semblable pour l'ensemble des requêtes.

```
  action requetesSQLgenerales {  
(1) string queryCapteurs <- "SELECT count(code)  
                             FROM (SELECT DISTINCT code FROM temp_layon)  
                             AS nbCapteurs ;" ;  
  
(2) ask dbCaller{  
      nbCapteurs <- (self select( (3) params: (4) PARAMS, (1) select: queryCapteurs )) (5) [2][0][0];
```

Figure 11: Exemple de requêtes faisant appel à une base PostGres en GAML

## II.2.3. Transfert des données par PostGres

Les requêtes sont transmises sous formes de listes de listes, comprenant trois listes principales (cf. Figure 12) :

```
[[noms des colonnes cibles], [natures des colonnes], [ [données ligne 1 ], [ données ligne 2], [...] ]]
```

```

Console
(1) 2013
(2) [['TMIN', 'TMAX'], ['FLOAT8', 'FLOAT8'], [[5.0, 15.1], [9.7, 14.9], [11.9, 16.0], [6.2, 1
(3) [['DATE_LAYON'], ['DATE'], [[2013-10-31], [2013-11-01], [2013-11-02], [2013-11-03],
(4) capteurs : 30
(5) '2013-11-01'

```

Figure 12: Exemple des informations retournées par la console à l'utilisateur lors du premier tour du modèle sous GAMA. (1) Année cible (2) Liste de listes des températures pour la période (3) Liste de listes des dates pour la période (4) Nombre de capteurs sur la zone (5) Date du premier jour de l'année agronomique (ici le 1<sup>er</sup> novembre 2013).

A chaque requête, il est donc nécessaire de descendre dans la chaîne ainsi créée, afin de sélectionner le bon maillon.

Plusieurs solutions sont possibles afin de garder ces informations en mémoire. Il est possible de conserver les données telles qu'elles sont transmises par PostgreSQL. Ce moyen est envisageable lorsque la requête est simple et n'implique qu'une colonne (cf. Figure 11 (5)). Pour accéder plus simplement aux données comportant plusieurs informations et ainsi alléger le code, il peut également être judicieux d'extraire uniquement la liste des données utilisées. Ce dispositif est employé pour manier les informations liées aux températures journalières (cf. II.4.3). Lorsque la requête est plus complexe et implique des relations spécifiques entre données, le choix a été fait de créer des *maps*, c'est-à-dire un ensemble de données (listes, matrices, d'autres maps...) liées par des clés ou index. Il est alors possible de rechercher une donnée particulière à partir de sa clé. C'est ce procédé qui est utilisé lors de la création des relations entre cépages, stades et indices bioclimatiques (cf. II.4.2).

## II.2.4. Utilisation de deux modèles

En raison de la faible vélocité d'un modèle faisant constamment appel à la base de données PostgreSQL, deux modèles ont été développés. L'un destiné à une utilisation pendant la phase de développement, l'autre constituant le modèle final. Le premier ne fait appel à la base de données qu'à l'initialisation, toutes les données étant prétraitées en amont via une table intermédiaire. Il supprime également un autre processus chronophage, le calcul du capteur le plus proche par parcelle, lui aussi implanté au préalable en base. Le deuxième modèle prend en considération toutes les modifications survenues en base de données, à laquelle il se connecte directement à chaque début d'année agronomique.

Tableau 5: Exemples comparatifs du temps nécessaire au fonctionnement du modèle ADVICLIM avec ou sans connexion à la base de données PostgreSQL, en moyenne sur une année simulée

Fonctionnalités	Avec connexion à la BD (ms)	Sans Connexion à la BD (ms)
<b>Calendrier (seul)</b>	125	11
<b>Changement de stades</b>	10 000	4
<b>Capteur le plus proche</b>	10	50 000
<b>Agrégation</b>	10	De 10 000 à + 60 000
<b>Ensemble du modèle</b>	<b>2 500</b>	<b>130</b>

## II.3. La gestion de l'information géographique sous GAMA

Pour plus de commodités, l'ensemble des données spatialisées proviennent de la base de données. La possibilité pour GAMA d'intégrer directement les couches d'informations sous forme de *shapefile* est donc ici totalement écartée.

Certaines couches ont été composées ou modifiées pour l'occasion, afin d'améliorer l'affichage graphique. Ainsi le fond de cartes permettant de visualiser les informations dans leur contexte géographique est composé des limites communales concernées par les activités viticoles. La couche de points des capteurs a été convertie en couche de polygones au vue de l'impossibilité de modifier l'aspect des symboles ponctuels (taille) à partir de GAMA.

### II.3.1. Définition de l'enveloppe de la couche

Il s'agit ici de donner une étendue d'affichage de la couche d'information en définissant son emprise totale. L'instruction passe par l'opérateur GAML *envelope*. La déclaration de la variable est similaire à celle des paramètres de connexion à la base de données (cf. II.2.2), exceptée la sélection de la géométrie de la couche qui servira de référence (ici le fond de carte).

### II.3.2. De la couche d'informations à la formalisation agent

Chaque couche d'information à afficher doit être définie en tant que *species* sous GAMA à l'initialisation. Chaque parcelle, chaque capteur, devient ainsi un agent autonome, capable d'interagir avec son environnement. Les attributs des couches peuvent être déclarés directement lors de cette étape en tant que variables. Les attributs sont ainsi lus et récupérés dans le modèle à partir des tables PostGreSQL. Un certain nombre d'attributs restera statique (viticulteur, commune, surface etc.) d'autres pourront avoir un caractère dynamique et être recalculés à chaque cycle, comme les indices bioclimatiques, les stades phénologiques, etc.

En plus de leur représentation géographique émanant de la géométrie des tables PostGIS les agents issus des *shapefiles* pourront présenter différents aspects, en rapport ou non avec leurs attributs ou leurs géométries. La géométrie de chaque agent est considérée comme une variable par GAMA, les projections spatiales des données est également gérer de façon interne. C'est ainsi que la représentation visuelle de l'état phénologique des parcelles peut évoluer au cours de la simulation.

Bien que GAMA soit capable de gérer trois types de topologies (vectorielle, grille et graphe), seul le mode vectoriel est utilisé. Ce choix se traduit par un affichage graphique plus fluide et facilite la création et l'export de nouveaux fichiers de formes représentant les futures « zones adaptées ». La vitesse d'affichage est grandement améliorée par rapport au modèle sous NetLogo qui s'appuie sur un automate cellulaire pour représenter les données.

### II.3.3. Attribution des capteurs par parcelles

Afin de prendre en compte les variabilités climatiques à échelles fines c'est le réseau de mesure mis en place dans le cadre du programme TERADCLIM et d'études de l'INRA qui est utilisé. En effet, le

maillage du réseau de Météo France ne permet pas de mettre en évidence les variations microclimatiques, les stations étant souvent trop espacées. Malgré cela, les stations Météo France sont utilisées comme références en cas d'absence de données sur certaines périodes.

A chaque parcelle est attribué le capteur météo le plus proche. Trois méthodes permettent de calculer cette relation de voisinage :

- En prétraitement, via QGIS, avec la fonction « *Distance to nearest Hub* » (cf. Figure 13)
- Directement sous GAMA, en utilisant l'opérateur « *closest\_to* »
- Sous GAMA mais en passant par la fonction PostGIS « *st\_distance* »

Le prétraitement via QGIS a été effectué afin d'améliorer la réactivité du modèle en phase de test.

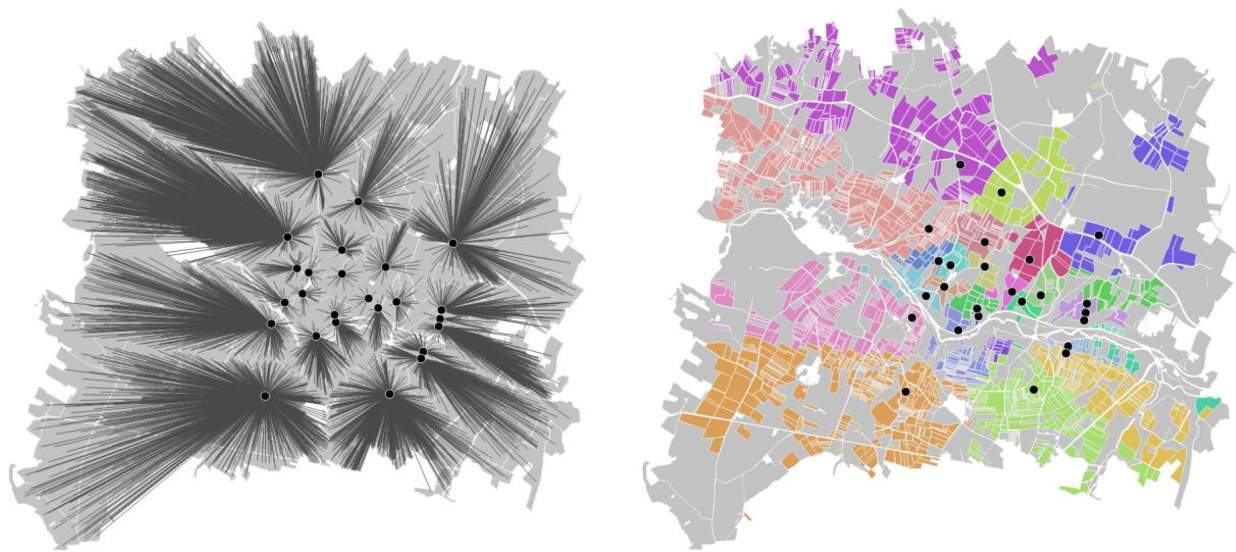


Figure 13: Attribution des capteurs par parcelles en fonction des relations de voisinages. a. Capture d'écran du résultat de l'algorithme « *Distance to the nearest hub* » de QGIS b. Résultat de l'analyse des relations capteurs/parcelles sous QGIS

Sous QGIS, c'est donc la fonction *Distance to nearest hub* qui est utilisée (*Boîte à outils de traitements* → *QGIS geotools* → *Vector analysis tools*). On obtient ainsi, en sélectionnant un fichier de forme « ligne », les plus courtes distances entre les capteurs et les parcelles (cf. Figure 13, a). Les codes de capteurs font ensuite l'objet d'une jointure attributaire, permettant d'associer parcelles et capteurs (cf. Figure 13, b). Les données sont donc directement accessibles pour une exploitation dans GAMA.

En effet, à la différence d'une utilisation directe des variables de chaque agent, l'usage d'un opérateur spatial GAML sera plus chronophage. En définitive, GAMA « interroge » chaque agent à chaque initialisation du modèle, la requête est donc exécutée plus de 2 000 fois, pour chacun des 25 capteurs, soit environ 80s au total.

En utilisant l'interopérabilité GAMA / PostGIS l'opération gagne en rapidité (seulement 400ms) mais les résultats doivent ensuite être retraités, alourdissant le code. La commande SQL utilisée classe les distances entre capteurs et parcelles, en ne retenant que les classements de rang = 1 (*Annexe SQL*).



Cette opération est réalisée entre les centroïdes des objets afin d'éviter les correspondances multiples, lorsque des parcelles ont plusieurs capteurs dans leur voisinage.

Ces trois procédés diffèrent par leurs temps d'exécution et leurs modalités de mise en œuvre. Bien que très pratique, à la fois en terme de récupération de données (donc de codage) et de rapidité, le prétraitement sous QGIS suppose, en revanche, une intervention sur les fichiers de formes à chaque modification du site d'étude. L'un des objectifs étant de pouvoir intégrer le plus facilement possible de nouveaux éléments, cette solution est peu viable sur le long terme. Par ailleurs, bien que l'opérateur GAML remplisse cette mission, il est assez peu performant en termes de rapidité d'exécution. Le passage par les fonctions spatiales de PostGIS paraît ainsi constituer le meilleur compromis.

## II.4. Modèle de base : croissance de la vigne

Après la phase d'initialisation permettant l'affichage des données géographiques et la mise en place de l'ensemble des requêtes correspondant aux choix utilisateurs, les données relatives à l'évolution de la vigne peuvent être exploitées. Les notions de temporalités reliées à leurs variables climatiques forment le cœur du modèle, dont le reste des variables est dépendant.

### II.4.1. Gestion du calendrier sous GAMA

Le temps et sa gestion tient une place prépondérante au sein du modèle, toutes les données climatiques étant datées et le profil climatique de l'année ayant son importance.

Contrairement à ce qui était supposé au début du stage, il n'y a pas de façon directe de gérer un calendrier sous GAMA. Une extension existe bien, mais pour en bénéficier il est nécessaire d'utiliser la version « développement » de GAMA, pour le moment peu stable et plus difficile à prendre en main pour les non-initiés.

Plusieurs solutions ont été envisagées afin de palier le problème :

- L'utilisation du calendrier PostGres
- L'utilisation du système Epoch<sup>18</sup>
- L'utilisation du principe des jours julien (cf. I.3.4)

C'est finalement un système hybride entre l'utilisation du calendrier offert par PostGreSQL et son adaptation au principe des jours julien qui a été privilégié. En effet, un parallèle entre la date Epoch et la fonction *date\_time* de GAMA avait dans un premier temps été envisagé. La manipulation des données s'est en définitive révélée plus problématique puisque demandant un post-traitement.

Le calendrier PostGreSQL est appelé via une fonction SQL. La date formatée de façon spécifique « *année-mois-jour* » est ainsi reconnaissable et utilisable. Cette modalité est exploitée dans le code

---

<sup>18</sup> Le système Epoch repose sur la comptabilisation des secondes écoulées depuis le 1<sup>er</sup> janvier 1970 à minuit (UTC/UGC). Chaque jour (et chaque seconde de chaque jour par conséquent) possède donc un identifiant unique. A titre d'exemple, le 15 septembre 2015 à minuit correspondrait à Epoch = 1442275200.

avec la possibilité de rajouter un entier à une date. Ainsi, à chaque cycle, la date du jour est incrémenté de un, simulant l'avancée dans l'année. Cette procédure a ensuite été associée à la récupération des données ayant trait aux températures, comme développé en II.4.2.

La version 1.7 de la plateforme GAMA, qui sortira à la fin de l'année 2015 offrira vraisemblablement la possibilité d'intégrer un calendrier aux données, modélisées de façon plus harmonieuse et logique.

## II.4.2. Les indices bioclimatiques

### *Une évolution dépendante des températures*

Les données clés régissant l'évolution des indices bioclimatiques sont constituées par les températures, maximales et minimales journalières. Afin d'outrepasser à la fois les problèmes liés à la faible performance du couplage GAMA/PostGres et au caractère lacunaire de la base de données, une requête SQL permettant de récupérer l'ensemble des données sur une année est envoyée à l'initialisation du modèle. En effet, deux écueils entravent la rédaction d'une requête simple :

- Certaines dates ne possèdent pas de données de température pour le capteur
- Certaines dates n'existent pas pour le capteur

Ces difficultés sont dues à la nature des appareils de mesure et au traitement qui en a été fait par la suite. Parfois fragilisés par les conditions climatiques, certains capteurs *Tinytag* ont subi des dysfonctionnements pendant plusieurs mois ; les données ont été stockées sous forme de tableur et celles concernées par les dysfonctionnements parfois complètement omises.

Dans le cas où les données de température ne sont pas disponibles pour un capteur, ce sont les données de la station Météo France la plus proche qui sont utilisées. Cette solution permet d'accéder à une série temporelle complète de 1988 à 2014.

La fonction SQL est construite en deux boucles imbriquées (cf. *Figure 14*), la première incrémentant le code des capteurs (1), la seconde incrémentant le calendrier (2). Elle repose sur la vérification des deux critères de validation de l'intégrité de la série temporelle pour le capteur, les « CASE » de la fonction SQL (3). Les variables dépendantes des choix utilisateurs sont intégrées en GAML dans le code ("**+variable+**"). Les résultats sont exportés dans une table temporaire (4), retraitée en *liste de listes* dans GAMA pour être réutilisés dans le modèle (5).

```

ask dbCaller{
// Mise à jour des données de la table temporaire sur toute la période sélectionnée.
//Boucles imbriquées sur le nombre de capteurs puis 365 jours
do executeUpdate(params: PARAMS,
updateComm: "DELETE FROM tempe_temp;");
do executeUpdate(params: PARAMS,
updateComm: // Sélection des tmin tmax en fonction de l'état des données. Si les données sont absentes des tinyTag,
//sélection du capteur MétéoFrance
"CREATE OR REPLACE FUNCTION table_correct() RETURNS SETOF tempe_temp AS
$BODY$

DECLARE i tempe_temp%rowtype;

BEGIN

FOR j IN 1..'+nbCapteurs+' LOOP
FOR i IN 0..365 LOOP
RETURN QUERY SELECT
Sélection
températures
minimales
(3)
CASE
WHEN (SELECT EXISTS (SELECT tmin FROM temp_layon WHERE date_layon = ('+date_dep+' ::date + i) AND code = j))
AND (SELECT tmin NOTNULL FROM temp_layon WHERE date_layon = ('+date_dep+' ::date + i) AND code = j)
THEN (SELECT tmin FROM temp_layon WHERE date_layon = ('+date_dep+' ::date + i) AND code = j)
ELSE (SELECT tmin FROM meteo WHERE date = ('+date_dep+' ::date + i))
END,
Sélection
températures
maximales
CASE
WHEN (SELECT EXISTS (SELECT tmax FROM temp_layon WHERE date_layon = ('+date_dep+' ::date + i) AND code = j))
AND (SELECT tmax NOTNULL FROM temp_layon WHERE date_layon = ('+date_dep+' ::date + i) AND code = j)
THEN (SELECT tmax FROM temp_layon WHERE date_layon = ('+date_dep+' ::date + i) AND code = j)
ELSE (SELECT tmax FROM meteo WHERE date = ('+date_dep+' ::date + i))
END;
END LOOP;
END LOOP;
RETURN;
END
$BODY$
language plpgsql;
(4) INSERT INTO tempe_temp SELECT * FROM table_correct();");
(5) tminmax <- (self.select(params:: PARAMS, select:: queryTminTmax))[2];
write tminmax;
}

```

Figure 14: Fonction SQL intégrée au script en GAML régissant la création d'une table source intègre pour les températures relevées par chaque capteur.

La liste ainsi obtenue est segmentée en période annuelle puis associée en tant que variable à chacun des agents capteurs.

### Deux compteurs opérationnels en parallèle

Comme il a pu être décrit dans la partie 1.3.4, deux indices bioclimatiques conditionnent l'évolution de la vigne : l'indice de degrés-jours et l'indice de Huglin. Ceux-ci ne se déclenchant pas au même moment, deux compteurs ont été mis en place, un pour chaque indice.

Ainsi, dès que la date du 1<sup>er</sup> novembre est atteinte, la variable « compteur de degrés-jours » est remis à zéro puis initialisée; il en est de même pour la variable « compteur de Huglin », son initialisation débute elle le 1<sup>er</sup> avril.

GAMA offre la potentialité de pouvoir suivre cette évolution à travers une représentation graphique (cf. Figure 15). L'axe des abscisses représente le temps qui s'écoule et est mesuré en cycles. Il est pour l'instant impossible d'intégrer des références temporelles comme les mois, qui ont été rajoutés manuellement pour plus de lisibilité.

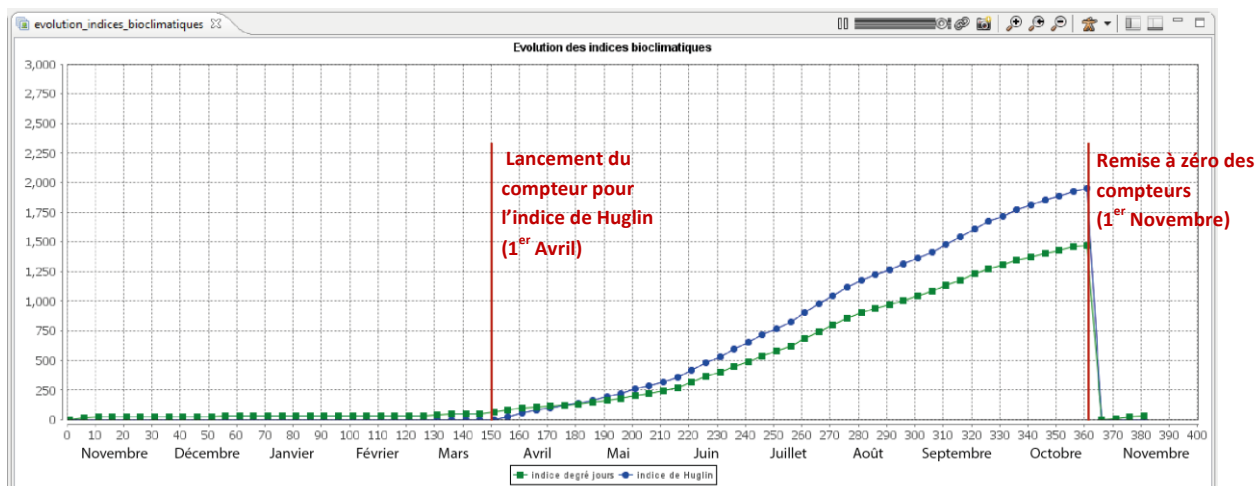


Figure 15: Evolution des indices bioclimatiques au cours d'une année agricole - Graphique généré sous GAMA

Le *reflex* de calcul des indices est réalisé à chaque pas de temps, pour tous les agents capteurs, à partir de la liste des températures (cf. point précédent). Les valeurs obtenues ne sont ajoutées aux compteurs que si elles sont positives. Si la valeur du compteur est différente de celle du tour précédent, elle est transmise aux parcelles concernées, qui détermineront si cela implique un changement de stade phénologique ou non.

Afin de faciliter leur exploitation par la suite, chaque valeur d'indice est appariée avec le code de son capteur. Puis l'ensemble des paires est collecté à chaque tour au sein d'une même entité : une *map*, dont les clés sont constituées par les codes capteurs. C'est cette *map* qui est envoyée aux parcelles. Les données correspondant à la fois à la clé et au capteur le plus proche de la parcelle sont ainsi associées.

### II.4.3. Stades phénologiques de la vigne et évolution

La gestion du changement de stade de la vigne demande une phase d'initialisation plus importante que pour les autres fonctions du modèle. Les traitements sont effectués à travers des requêtes à la base de données puis sont retraités dans GAMA. La *Figure 16* reprend schématiquement les échanges de variables entre le monde (global) et les agents capteurs et parcelles, permettant de simuler ce processus. Pour plus de clarté, seules les parcelles concernées par une évolution sont représentées, associées à leurs capteurs, en traits pleins.

Les variations d'évolution sont entre autre dépendantes du type de cépage. Si quinze cépages sont répertoriés dans la base de données, seulement sept sont présents sur la zone du Layon et huit au total sur l'Anjou. Il convient donc, afin de minimiser la quantité de données stockées par GAMA, de ne sélectionner que les cépages existants (1). Les requêtes sont ensuite faites uniquement sur cet échantillon.

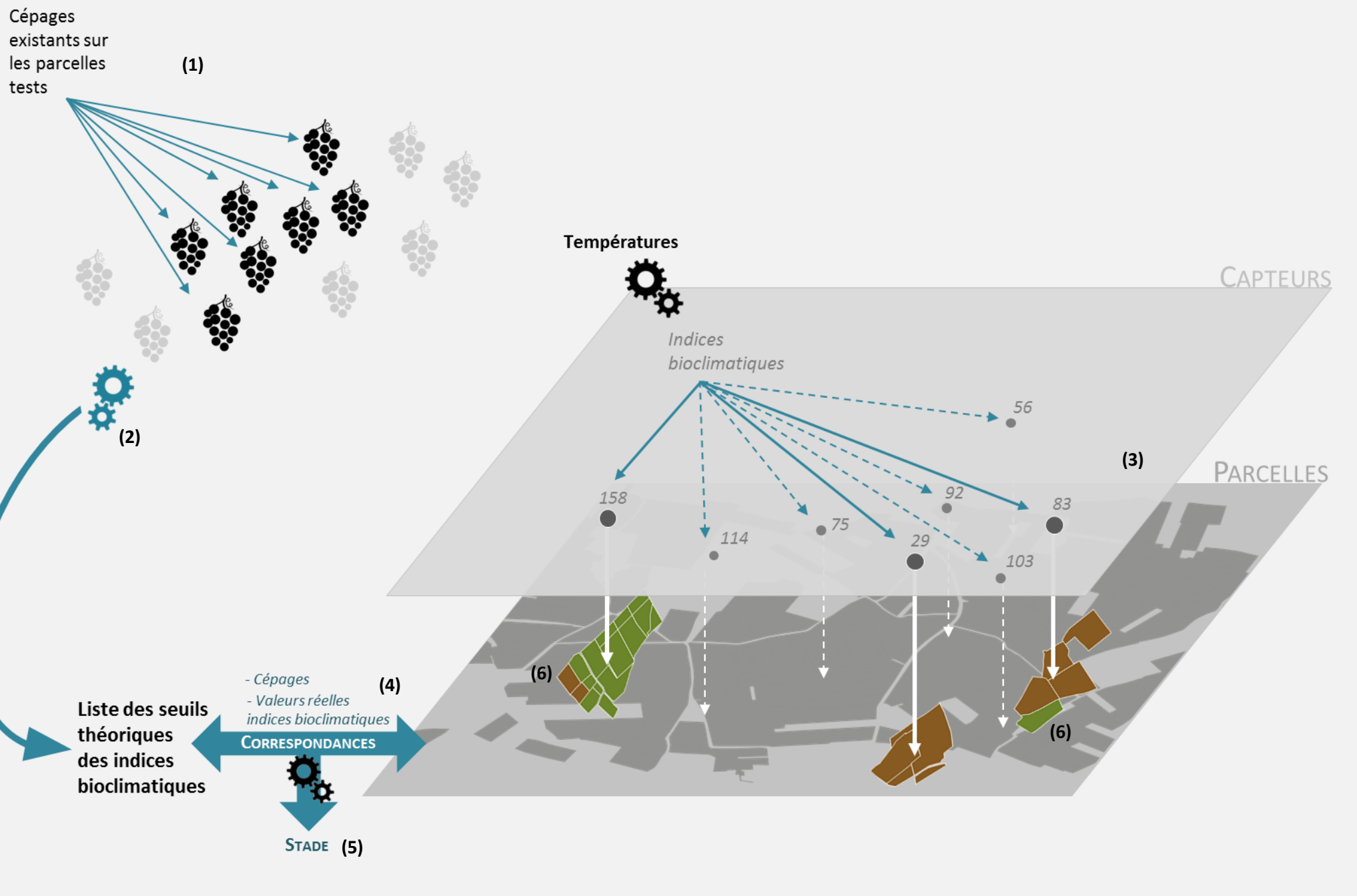


Figure 16: Représentation schématique du flux d'informations régissant l'évolution de la vigne sous GAMA

Les cépages sont mis en correspondance avec les seuils des indices bioclimatiques à travers une boucle se basant sur le nombre de cépages existants et l'ensemble des seuils récupérés par requête (2). A chaque boucle, les données d'intérêts sont extraites pour intégrer une liste générale comportant uniquement les informations relatives à l'évolution des sept types de vignes présentes.

Une fois les indices journaliers calculés et associés aux agents parcelles correspondants (3), ils sont comparés à des seuils théoriques à atteindre (4) pour que la vigne change de stade (5) si besoin.

Chaque stade clé (débourrement, floraison, nouaison, véraison, maturité, cf. 1.3.4, Figure 3) correspond à un code couleur permettant de visualiser graphiquement l'évolution phénologique de la vigne. Etant dépendante du cépage, il est ainsi possible de retrouver des situations similaires aux points (6), où des groupes de parcelles, bien qu'attachées aux mêmes données de température, donc aux mêmes valeurs d'indices bioclimatiques, ne présentent pas le même stade phénologique, car issu d'un cépage différent.

L'évolution complète d'une année agronomique (conditions classiques) est illustrée en *Annexe II*.

## II.5. Modélisation multi-échelles

Le modèle a pour finalité de déterminer des « zones potentielles d'aptitudes ». Elles sont produites par des agrégats de parcelles, calculés à partir de critères propres aux parcelles ou à son environnement.

### II.5.1. Agrégation des parcelles à forte potentialité

Une fois identifiées, les parcelles sont agrégées pour ne former qu'une entité spatiale, à un niveau d'échelle supérieure. Bien qu'il existe une fonction d'agrégat sous GAMA, c'est une fonction spatiale PostGIS qui est utilisée, bien plus efficace. En effet, le traitement sous GAMA oscillait entre une durée d'une dizaine de secondes pour les agrégats le plus réduits à plusieurs minutes pour les plus étendus ; contre un résultat quasi instantané en passant par PostGIS.

L'agrégation est réalisée via une instruction *ST\_Union* (cf. *Annexe IV*) commandée par un *reflex*. Le résultat est associé à une nouvelle *species* créée seulement lors du déclenchement du processus. La géométrie de la forme constituée provient de la table résultante du *ST\_Union*. Afin de bien visualiser le résultat, l'agrégat est situé au-dessus de toutes les autres couches avec une légère transparence et une couleur très vive (cf. *Figure 17*).

Les données n'étant pas suffisantes pour le moment, le *reflex* a été basé sur des attributs arbitraires. Ce fonctionnement laisse cependant la possibilité d'effectuer des analyses multi-critères, représentatives de la complexité du fonctionnement d'un système viticole. Ainsi, tout ou partie des contraintes environnementales influençant l'évolution de la vigne dans le temps (température, nature du sol, phénologie, encépagement) peut être pris en compte dans le *reflex* d'agrégation. La temporalité de la fonction d'agrégation est également paramétrable, à des dates fixes (dernier jour de l'année agronomique, horizon temporel + 20 ans etc.) ou en fonction d'évènements particuliers (maturité, créneau possible de vendange, etc.).

Le processus d'agrégation développé pour ADVICLIM est illustré en *Figure 18*. Les filtres d'agrégation sont représentés comme différentes couches par lesquelles transitent les informations générales sur la parcelle testée. Les critères d'agrégation sont combinés et comparés à des valeurs théoriques ou connues. Une des couches fait ici référence au bilan hydrique calculé par le modèle WaLIS se basant entre autre sur la pédologie propre à chaque parcelle. L'intégration future de cette variable dans le modèle est détaillée en partie *III.3.1*.



*Figure 17: Capture d'écran de la création d'un agrégat sous GAMA*

Le résultat de la fonction d'agrégation est exporté en fichier de forme également à travers un *reflex*, chaque nom de fichier peut être différencié, grâce à l'intégration de variables spécifiques à chaque tour : date, capteur, cycle, commune, etc.

Le modèle du Layon offre une très grande précision au niveau local. Cette résolution nécessite de disposer de nombreuses données sources difficiles à acquérir à l'échelle d'une région de production. Dans ce contexte, il s'avère indispensable de calculer des agrégats spatiaux compatibles avec des jeux de données issus de réseaux d'observation régionaux ou de sorties de modèles climatiques basés sur des résolutions plurikilométriques. Le modèle développé sur le site pilote du Layon pourrait ainsi avoir également une fonction d'agrégation générale, non seulement pour évaluer les parcelles à forte potentialité d'adaptation, mais également sur d'autres critères individualisés ou non (sol, cépages, bilan hydrique, maturation etc.).

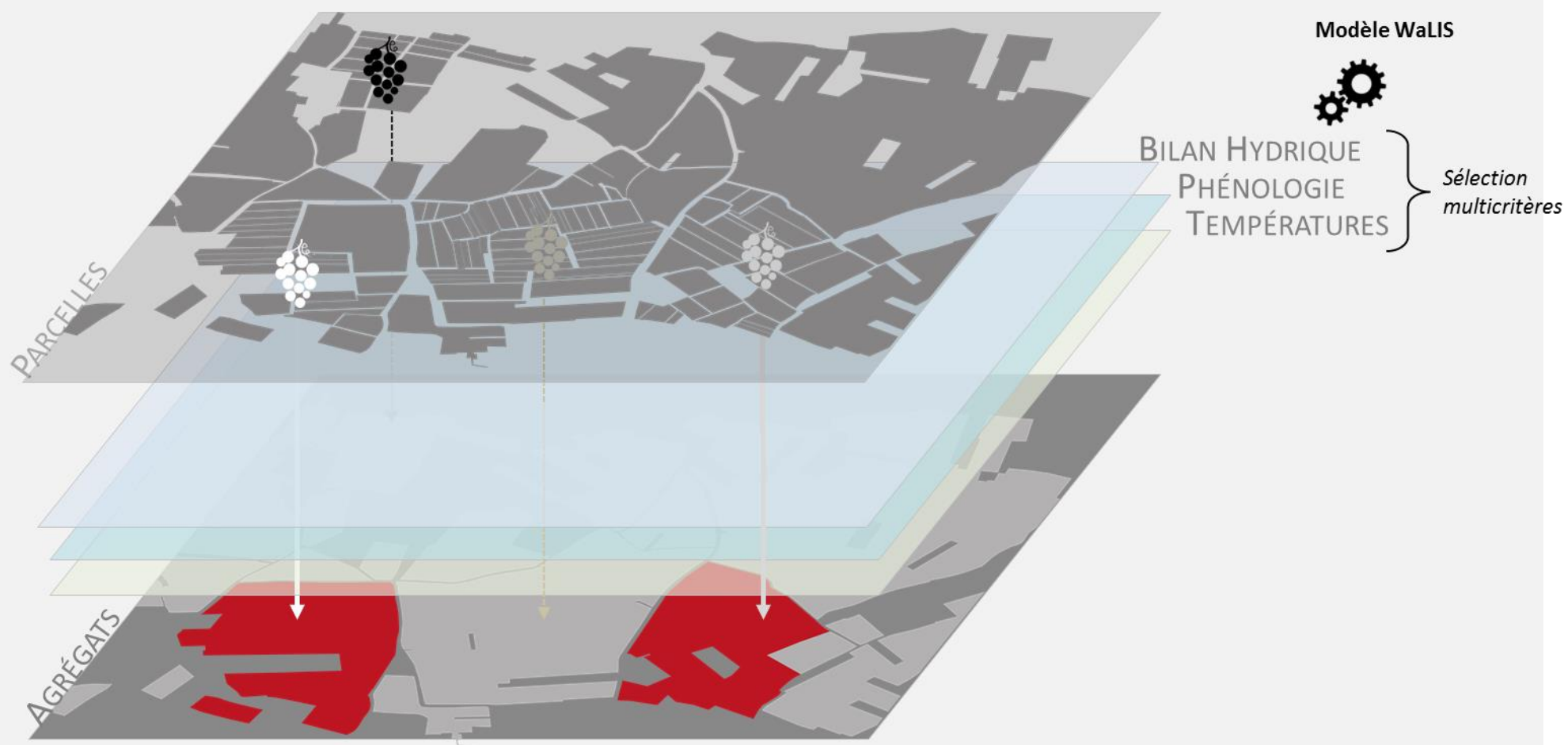


Figure 18: Schématisation de la fonction d'agrégation développée sous GAMA pour le projet ADVICLIM



# III. ANALYSE CRITIQUE

---

## III.1. Difficultés rencontrées

La modélisation est un domaine très spécialisé qui requiert des connaissances approfondies sur les méthodes et principes de simulation en eux-mêmes mais aussi sur le terrain d'étude.

### III.1.1. Difficultés liées au domaine d'investigation

Le vocabulaire dédié à la viticulture est spécifique, tout comme les techniques agricoles. Des recherches ont été faites afin de déterminer les répercussions exactes de chaque action agronomique, pour qu'elles soient simulées de manière la plus réaliste possible. L'apprentissage de ce vocabulaire complètement inconnu quelques mois auparavant a demandé quelque temps. De plus, évoluant dans une thématique abordée l'angle de la recherche, le niveau technique des publications lues sur un sujet encore mal maîtrisé s'est révélé assez ardu.

### III.1.2. Difficultés inhérentes aux limites du logiciels

GAMA est une plateforme relativement récente qui dispose certes d'une documentation assez fournie mais incomplète (absence totale de description et d'exemple pour certaine fonction) et parfois obsolète.

Plusieurs moyens existent afin de trouver une information sur GAMA : la documentation en ligne, la documentation intégrée au logiciel, les tutoriels intégrés au logiciel, les tutoriels en ligne et surtout le forum GAMA. Ce dernier dispositif a constitué une aide précieuse, les membres de la communauté étant très réactifs.

## III.2. Limites

Le projet en étant encore au stade de prototype, plusieurs problématiques limitantes ont été soulevées au cours du stage :

- L'absence d'une partie des données en Anjou a réduit le cadre géographique
- L'utilisation d'un algorithme du plus proche voisin pour rattacher les données des stations aux parcelles reste discutable
- Le modèle WaLIS, sur lequel repose la disponibilité en eau est basé sur un climat méditerranéen
- Le manque d'interopérabilité entre GAMA, PostGreSQL/PostGIS et R pourra à terme devenir une source de lenteur et d'approximation
- L'hétérogénéité des données climatiques ralentie leur intégration

### III.2.1. Des données absentes

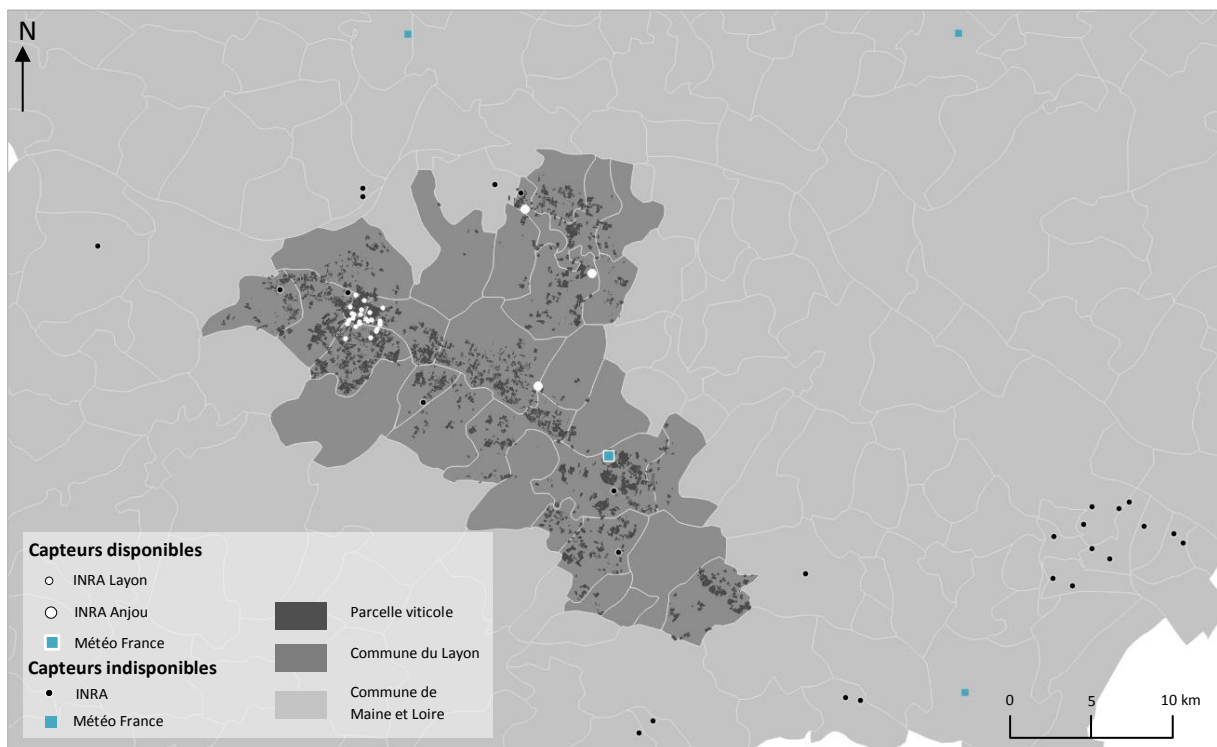
La vocation du projet était non seulement de passer du site pilote « Quart de Chaume » aux Coteaux du Layon, mais également, à terme, d'étendre le modèle à toute la zone de l'Anjou. Plusieurs

analyses ont déjà été réalisées sur cette région, l'INRA possède donc un certain nombre de données potentiellement exploitables. Ainsi, nous avons pu obtenir l'intégralité du parcellaire viticole, ses caractéristiques pédologiques, encépagement, etc. La position des différents capteurs implantés dans la région nous a également été transmise.

Après investigation, la grande majorité des données enregistrées par ces capteurs n'est finalement pas exploitable. La répartition spatiale des stations météorologiques n'est pas suffisamment homogène pour permettre une exploitation comparable des données en tout point du domaine de simulation. En effet, le site pilote des Coteaux du Layon possède un réseau d'acquisition à haute résolution, puisque 25 capteurs permettent de réaliser une analyse relativement fine de la situation climatique. En revanche, sur le reste des vignobles de l'Anjou, seules quatre stations (trois de l'INRA et une Météo France) peuvent être mobilisées sur la période (cf. *Carte 5*). Des modèles d'interpolation pourraient être mobilisés afin de pallier cette limite mais ils restent complexes à mettre en œuvre et peu fiables sur certains paramètres (la pluviométrie notamment).

L'utilisation de ces données à l'échelle de la parcelle n'est donc pas pertinente. La construction d'une couche d'agrégats par simulation (analyse des similarités agro-pédologiques et climatiques) pourrait permettre une exploitation adaptée de ce jeu de données.

Enfin, les séries temporelles disponibles ne s'étendent que de 2010 à 2014. Cette période est trop réduite pour construire de réels scénarios climatiques d'autant qu'il s'agit d'années aux profils climatiques peu différenciés.



Carte 5: Situation des différents capteurs météorologiques en Anjou

### III.2.2. Pondération de l'algorithme du plus proche voisin

Afin de rattacher une parcelle à un capteur, la méthode utilisée est celle du plus proche voisin (cf. II.3.3). Bien que valable, elle reste néanmoins discutable. En effet, la température peut subir des variations d'ordres locaux, sous l'influence de la topographie ou par la proximité de cours d'eau, de tissu urbain, de forêt etc.

#### *Topographie*

La topographie peut avoir une influence sur la température relevée, comme le décrit la *Figure 19*. L'implantation des capteurs météorologiques ADVICLIM prend en compte ce type de variabilité, la pente, l'altitude, l'ensoleillement ont été des critères de mise en place. Malheureusement, ces variables ne peuvent être prises en compte pour certaines parcelles en raison de leur éloignement des sites de mesures. Afin de produire un algorithme d'interpolation adapté, il pourrait être envisageable d'utiliser le Bd Alti® IGN.

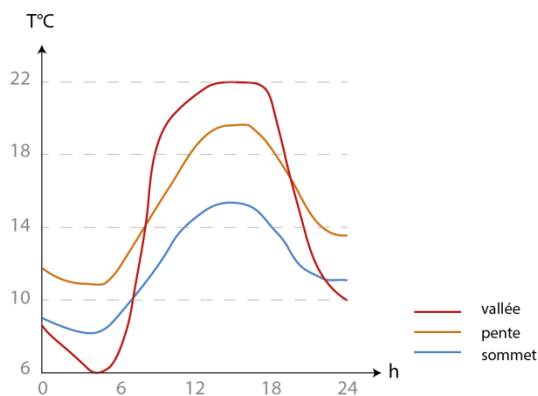


Figure 19: Influence de la topographie sur l'évolution des températures journalières estivales (D'après E. Gratia)

En altitude la température aura tendance à être en moyenne plus haute en début de journée mais n'atteindra pas des valeurs aussi importantes qu'en basse altitude. Les tendances s'inversent en basses altitudes. On remarque que la situation d'une station ou d'une parcelle en pente pourra également avoir son influence, les rayons solaires n'ayant pas le même angle d'incidence.

Les différences topographiques sur la zone du Layon restent cependant relativement faibles (Bonney, 2013). La situation est plus nuancée sur l'ensemble de l'Anjou, une pondération pourrait être pertinente à cette échelle.

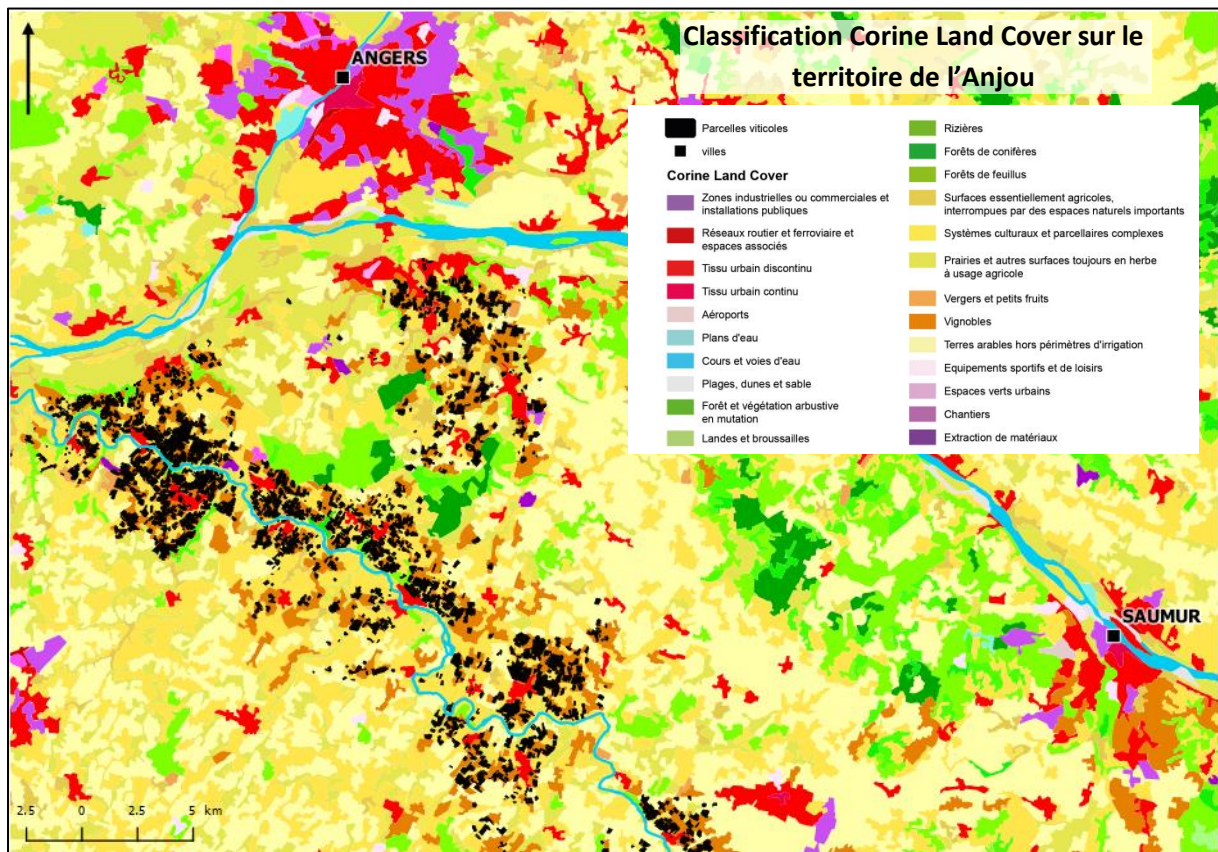
#### *Environnement direct*

L'environnement entourant les parcelles viticoles peut également pondérer les températures relevées par les capteurs *Tinytag* sur le terrain. C'est en tout cas le présupposé qui a été fait lors du développement collectif d'un modèle test, durant la réunion *Maps8* auquel participait Mathias Rouan, un des initiateurs du projet, à Géomer.

En effet, la proximité d'un tissu urbain, tout comme de systèmes culturels, peut amener à une légère augmentation de la température au niveau local. A contrario, la contiguïté avec un cours d'eau ou encore une zone forestière peut être synonyme d'un rafraîchissement. Afin de reproduire sommairement l'environnement de chaque parcelle, la classification *Corine Land Cover* pourrait être utilisable.

La résolution de *Corine Land Cover* n'est néanmoins pas suffisante pour être utilisée sur les parcelles test du Layon. Cette éventuelle utilisation n'aurait de sens que si la zone est étendue à l'ensemble de l'Anjou. Après évaluation de la zone d'impact de chacun des facteurs diminuant ou augmentant potentiellement la température, une pondération des variables pourrait ainsi être mise en place.

La *Carte 6* constitue un exemple des données *Corine Land Cover* disponibles sur le territoire viticole de l'Anjou.



Carte 6: Environnement du parcellaire sur l'Anjou selon la classification *Corine Land Cover* 2012

### III.2.3. Un climat méditerranéen à l'origine du modèle de disponibilité en eau WaLIS

Le modèle simulant au pas de temps journalier la fraction d'eau du sol disponible pour la vigne a surtout été étudiée en climat méditerranéen (Celette *et al.*, 2010). Le coefficient associé aux cultures, dépendant de l'interception des radiations et de leur transpiration est en effet calculé à partir de données induites par le climat méditerranéen (Celette *et al.*, 2010). Il a cependant fait ses preuves lors de tests réalisés dans le pays Gascon (Dufourcq *et al.*, 2013).

Ces deux climats et ces deux types de cultures pouvant différer de ceux retrouvés dans le Layon, il pourrait être pertinent de modifier quelques variables du modèle pour l'ajuster localement.

### III.2.4. Un temps de calcul qui rend peu exploitable le modèle auprès des viticulteurs

Le couplage des modèles WaLIS (code R), et ADVICLIM (code GAMA), pose pour le moment d'importants problèmes de temps de calcul. En effet, plusieurs heures sont nécessaires pour simuler une année agronomique lorsqu'ils sont couplés.

### III.2.5. Des données brutes nécessitant un prétraitement chronophage

Les données provenant des stations météorologiques sont traitées par un opérateur extérieur et les objectifs d'archivages pour l'INRA ne rejoignent pas ceux du projet ADVICLIM. En effet, stockées dans un tableur, les valeurs mensuelles des variables d'intérêts (température, pluviométrie, vent etc.) sont segmentées dans des feuilles de classeur indépendantes, à la mise en forme hétérogène. Cette classification rend l'intégration des données chronophage et peut être source d'erreur, l'obtention d'un tableur propre nécessitant une multiplication de la manipulation des données, de type copier/coller (la mise en œuvre d'une macro s'avérerait trop complexe compte tenu de l'hétérogénéité des données).

## III.3. Bilan et perspectives

Le modèle ADVICLIM développé sous GAMA remplit ses objectifs en termes de simulation de l'activité viticole telle qu'elle a été volontairement simplifiée dans le cadre du stage. La base de données a été homogénéisée et organisée, le code du modèle est optimisé afin de pouvoir intégrer des données avec le moins de modifications possibles.

Du point de vue de la gestion des différents niveaux d'échelles, les recherches ne sont pas terminées, les possibilités qu'offre GAMA semblent prometteuses grâce à son couplage avec PostGIS. Cette interopérabilité reste néanmoins contextuelle, certaines fonctions nécessitant, en effet, un temps de calcul extrêmement long sous PostgreSQL/PostGIS alors qu'ils sont quasi instantanés sous GAMA, et vice et versa. On remarque tout de même que les opérations spatiales bénéficient d'un traitement plus efficace sous PostGIS que sous GAMA. Ce résultat est-il induit par la connexion générale à une base de données ? Il serait nécessaire de chercher plus avant les raisons de ces différences.

### III.3.1. La disponibilité en eau, facteur limitant de la croissance

Comme une majorité d'organismes foliés, la vigne requiert eau et rayonnement solaire afin de réaliser sa croissance. Cette eau est puisée dans le sol et doit être en quantité suffisante. Sa disponibilité dépend en grande partie du type de sol et de l'entretien qui en est fait. Cette variable peut-être approximée à travers le *curve number* du sol (CN), que l'on pourrait traduire par le coefficient de ruissèlement associé au sol. Le *Tableau 6* retranscrit les valeurs de CN liées aux

groupes et aux types de sol. Le pourcentage représente l'eau qui ne peut pas être utilisée par la plante (qui a ruisselé).

L'enherbement correspond en viticulture à l'herbe volontairement laissé dans le cavaillon<sup>19</sup>. Cette partie enherbée pourra permettre de contrôler de manière plus fine les apports en eau à la vigne. Le désherbage libèrera une fraction utilisable (l'eau n'est plus exploitée par l'herbe et est rendue disponible pour la vigne), au contraire, un enherbement concurrent limitera les apports à la vigne.

Tableau 6: Détermination des curve number (coefficient de ruissèlement) selon le type de sol et l'entretien du sol (D'après Romero et al., 2007)

Groupe de sol	A	B	C	D
<b>Sol nu</b>	72	83	89	93
<b>Sol labouré</b>	57	71	76	80
<b>Sol enherbé</b>	55	75	82	89

Avec :

**A** : Sol avec forte capacité d'infiltration. Sol sableux (<10% argile et >90% sable ou graviers). Certains sols avec une forte teneur en cailloux (>35%) et une faible densité peuvent être classés dans ce groupe.

**B** : Sol avec une bonne capacité d'infiltration. Typiquement entre 10 et 20 % d'argile, et 50 à 90% de sable.

**C** : Sol avec une capacité d'infiltration modérée. Typiquement entre 20 et 40% d'argile, moins de 50% de sable.

**D** : Sol avec une capacité d'infiltration faible. Typiquement plus de 40% d'argile, moins de 50% de sable.

Les groupes de sols sont basés sur la classification du triangle des textures pédologiques (cf. Figure 20). Comptant à l'origine plus d'une dizaine de catégories liées au pourcentage d'éléments présents dans le sol (argiles, limons et sables), les classes ont été réduites au nombre de quatre (A, B, C, D).

A la lumière de ces éléments, on remarque qu'un sol plutôt sableux aura plus de capacité aquifère qu'un sol à forte teneur en argile, même entretenu. Ajouté à cela, il est nécessaire de prendre en compte l'épaisseur de la couche utile (celle capable de retenir l'eau). Cette variable est prise en compte dans le calcul de la Réserve Utile Maximale (RUM) fournie par l'INRA.

Le calcul du bilan hydrique est réalisé par le modèle WaLIS (implémenté sous R, cf. 1.3.3), dont la résultante est une valeur de FTSW : fraction transpirable du sol, qui traduit la disponibilité en eau pour la vigne pour chaque parcelle, conditionnant le potentiel de vigueur de celle-ci et donc par extension sa capacité à croître.

<sup>19</sup> Le cavaillon est la bande de terre située entre les pieds de vignes.

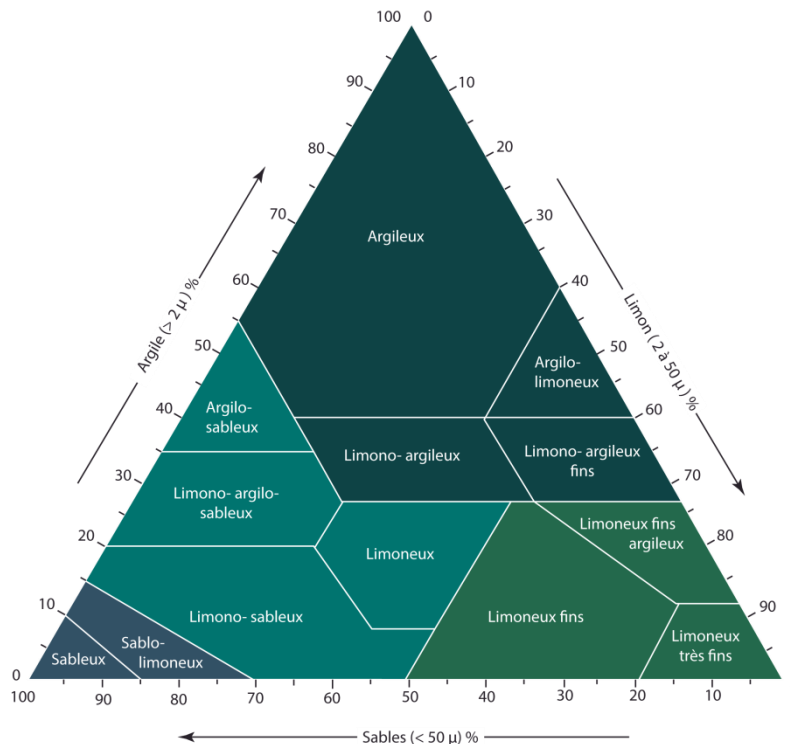


Figure 20: Triangle des textures pédologiques (classification américaine)

### III.3.2. Agrégats spatiaux et niveaux d'échelles

Au stade de développement atteint à la fin de ce stage, il n'est pas encore possible de faire remonter des informations en relation avec l'état théorique futur des parcelles de vignes. La remontée d'informations est donc simulée, en se basant sur des données rétrospectives, qui permettent de déterminer des parcelles homogènes (ou hétérogènes) sur des critères phénologiques et climatiques.

#### *Un paramétrage plus fin des conditions d'environnement*

Les données nécessaires à une agrégation plus réaliste nous sont parvenues relativement tard et n'ont pu être exploitées. On suppose que les caractéristiques pédologiques des parcelles devraient rester relativement stables au cours du temps, rendant la valeur de la fraction d'eau utilisable par la vigne uniquement dépendante des variables climatiques et des actions agronomiques mises en œuvre par les viticulteurs. En recoupant ces données avec les variables climatiques et les caractéristiques intrinsèques de la vigne, il pourra être envisageable de déterminer les parcelles les plus adaptées à un scénario climatique et un horizon temporel définis. Les parcelles seront agrégées en suivant un processus similaire que celui du modèle test.

Dans l'idéal, il serait non seulement nécessaire de définir si l'encépagement est toujours adapté aux conditions climatiques mais aussi de déterminer si la typicité des terroirs est conservée. L'intégration d'un l'agent superviseur du type INAO, permettrait de contrôler la conformité des caractéristiques de la vigne avec le cahier des charges de l'appellation concernée.

#### *Une modélisation explorable à plusieurs niveaux*

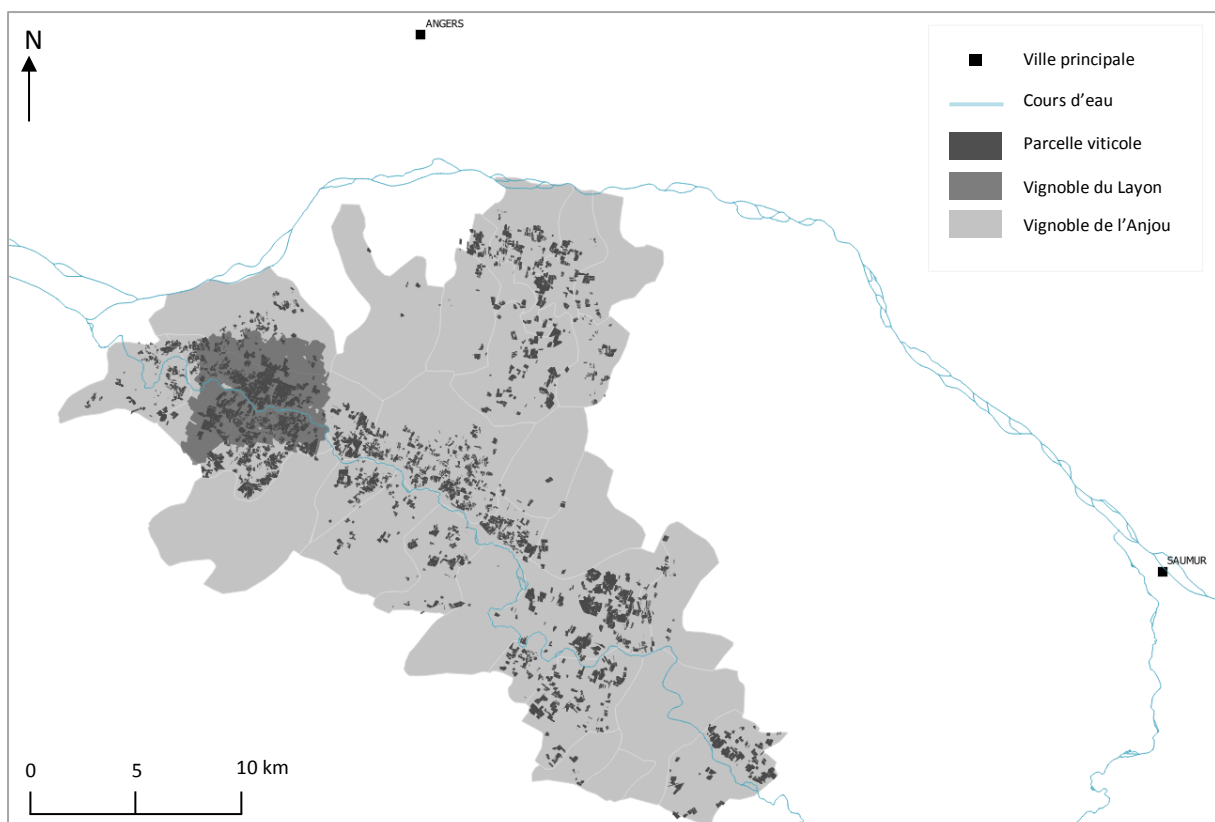
Bien qu'intéressant à petite échelle, le modèle ADVICLIM peut également se révéler utile à grande échelle. Dans ce cadre, la mise en place d'une sélection de données, tant par critères géographiques

que par critères scalaires pourrait être pertinente. L'emprise spatiale du domaine de simulation pourrait dans ce cadre déterminer le niveau d'abstraction du modèle afin de mobiliser les méthodes les mieux adaptées à la résolution spatiale choisie.

### III.3.3. Utilité d'étendre le modèle à l'Anjou

Le prototype développé sur le Layon offre des résultats à une échelle spatiale plus petite que ceux obtenus sur la zone du Quart-de-Chaume. Plus diversifiés, les caractéristiques intrinsèques des parcelles permettent également de donner un aperçu plus proche de la réalité. Il paraît néanmoins difficile de généraliser les conclusions qui pourraient être tirées du modèle ADVICLIM – Layon à un territoire plus étendu. La *Carte 7* illustre ainsi les relations spatiales entre les territoires du Layon et de l'Anjou.

L'optique de pouvoir à terme, avec suffisamment de données climatiques, étendre le modèle à l'ensemble du territoire viticole de l'Anjou permettrait d'établir une simulation à méso-échelle mieux adaptée à la prise en compte des données des modèles climatiques (la résolution la plus fine disponible actuellement à l'échelle de l'Anjou est de 8 km<sup>2</sup>). Le niveau de précision élevé obtenu sur la zone du Layon reste pour l'instant difficilement transposable sur l'ensemble des sites du projet ADVICLIM. Il propose néanmoins une restitution adaptée à la prise en compte des variabilités climatiques à échelle fine qui sont déterminantes pour la phénologie de la vigne.



Carte 7: Mise en perspective de la zone test étudiée lors du stage avec la zone d'étude totale potentielle



### III.3.4. Interactions parcelles – INAO : composante socio-économique

L'INAO, l'Institut National de l'Origine et de la Qualité (de son ancien nom Institut National de Appellations d'Origine), fixe un certain nombre de critères à respecter afin de pouvoir utiliser l'Appellation d'Origine Contrôlée (AOC) « Coteaux du Layon ». Les dispositions peuvent être résumées de façon générique à :

- La couleur et le types de produit
- La zone géographique
- Le type de cépage
- Le type de conduite agronomique
- La période de récolte
- La maturité
- Les pratiques œnologiques (% alcoométrique volumique)
- La teneur en sucres fermentescibles
- ...

L'AOC « Coteaux du Layon » suppose, entre autres, des tries successives de raisins arrivés à surmaturité (forte concentration en sucre). Cette surmaturité est favorisée par les années chaudes et sèches, favorisent cette forte teneur en sucre associée au développement d'une pourriture noble. Au lieu d'engendrer des vins de catégorie moelleux ou liquoreux, les baies récoltées au cours d'années défavorables feront souvent l'objet d'une cuvée de vin blanc sec.

Ainsi, au sein du modèle créé sous NetLogo, l'INAO est associé à un agent superviseur, qui fixe et contrôle le respect du cahier des charges. A terme, l'agent INAO a la possibilité de le modifier en fonction des données renvoyées par les viticulteurs. Au stade d'ébauche, cet agent n'a pas encore été intégré à la version du modèle rendu pour ce stage.

Pour compléter le modèle, l'agent superviseur INAO devra pouvoir non seulement imposer et contrôler le cahier des charges mais aussi le faire évoluer en fonction des résultats de simulation. Ce procédé a été mis en place par le groupe de travail dont faisait partie Mathias Rouan dans le cadre de l'Ecole Thématique MAPS 8.

Il a été pris comme postulat qu'à partir d'une valeur seuil de parcelles dont la récolte ne permet plus de produire des cuvées répondant au cahier des charges INAO, celui-ci pourrait être modifié. Un bilan réalisé régulièrement rend ainsi compte des parcelles restant ou sortant de l'appellation en fonction du profil climatique choisi. A partir de cette expérimentation, on peut également imaginer de nouvelles préconisations en termes de cépage, en comparant le cahier des charges modifié avec celui mis en place dans d'autres zones, pour d'autres encépagements. En fonction de ces préconisations, l'agent viticulteur pourrait faire le choix de conserver ou de changer de cépages sur les parcelles concernées.

### III.3.5. Simplifier pour mieux modéliser, des horizons temporels plus lointains

Plusieurs enseignements peuvent être tirés du travail réalisé à l'Ecole Thématique MAPS 8, notamment le fait de considérer les années agronomiques dans leur globalité, et non pas de manière journalière comme elles sont appréhendées dans le prototype développé au cours du stage. Même si l'imbrication de plusieurs échelles de temps au sein du modèle avait été envisagée dès le début du stage, cette agrégation temporelle impliquerait de gérer plusieurs niveaux d'abstraction dans le modèle.

De façon similaire, l'intégration du modèle WaLIS à la simulation ADVICLIM pourrait être réalisée à partir de régressions multiples recherchant des corrélations entre les valeurs des balances hydriques et les données de températures et de pluviométrie. En effet, le modèle tournant sous R requiert un éventail de données très conséquent, qu'il serait éventuellement possible de réduire à des moyennes annuelles pour plus d'efficacité.

La mise à jour prochaine du logiciel GAMA 1.7 devrait intégrer des opérateurs temporels. Bien que le système de « calendrier » développé dans le modèle ADVICLIM reste relativement efficace, cette évolution faciliterait considérablement la gestion du temps dans le modèle.

### III.3.6. Bilan personnel

La conduite de ce projet s'est révélée extrêmement enrichissante tant au niveau technique qu'en termes d'encadrement et d'opportunités d'avenir.

Outre la manipulation de base de données de façon plus chevronnée qui apparaît désormais comme essentielle dans les activités du géomaticien, la découverte d'un environnement de développement entièrement dédié à la simulation ouvre des perspectives plus larges à l'utilisation des données géographiques.

Le retour à un contexte de recherche, intégré à un équipe pluridisciplinaire, a permis énormément d'échanges et une mise en perspective de la place prépondérante des systèmes d'informations géographiques dans les recherches actuelles.

La période impartie pour la réalisation du prototype, même rallongée, n'a pas été suffisante pour assimiler l'ensemble des tenants et aboutissants du projet. Une lecture plus approfondie des différentes études et articles connectés aux sujets permettront d'intégrer l'ensemble des composantes et des enjeux de l'adaptation des pratiques viticoles au changement climatique. C'est dans ce cadre que ce travail se poursuivra, durant une année au sein du laboratoire LETG/Brest Géomer.

## III.4. Conclusion

La démarche de modélisation entreprise lors de ce stage propose une approche globale visant à simuler l'évolution de parcelles viticoles sous contraintes climatiques dans un contexte de changement global. L'utilisation de la plateforme GAMA a permis de modéliser les interactions entre différentes composantes de la viticulture, ici intentionnellement réduits à deux entités : la parcelle et le climat, matérialisé par une station météo. La finalité était double, d'une part restituer la dynamique de la vigne à partir des indices bioclimatiques et d'autre part de déterminer un protocole visant à redéfinir des zones d'aptitudes agro-climatiques.

Au cours de la phase de développement, les relations formalisées entre les agents ont fait l'objet d'une recherche d'optimisation constante, tirant partie des meilleures performances offertes par le couplage entre modèle multi-agent et base de données spatio-temporelles. Il s'agissait en particulier d'améliorer l'intégration de données spatio-temporelles composites comme variable de forçage, afin de restituer la variabilité du déroulement des activités viticoles.

A l'issue de cette première phase de développement, on peut considérer que la plateforme GAMA semble adaptée à la gestion d'interactions multi-scalaires. De nombreuses perspectives d'améliorations restent cependant à mettre en œuvre dans la suite du projet. L'intégration complète de l'ensemble des paramètres environnementaux, pluviométrie, pédologie, entre autres, permettra de rendre compte d'une variabilité plus fine des caractéristiques intrinsèques des parcelles. Dans ce cadre, l'exploitation du couplage entre les modèles ADVICLIM et WaLIS (Celette *et al.*, 2010, Dufourcq *et al.*, 2013), implémenté sous R devra être améliorée. La composante socio-économique représenté par le couple INAO / viticulteurs et les boucles de rétroaction formées, devrait permettre de répondre plus finement au second objectif de l'étude concernant la mise en évidence de zones d'aptitude agro-climatique.

A terme, l'absence de données sur l'ensemble de la zone potentielle d'étude de l'Anjou pourrait avoir un caractère particulièrement limitant, la faible étendue géographique du seul territoire du Layon ne permettant pas de tirer des conclusions généralistes.

Par ailleurs, à travers le monde, de nombreux domaines viticoles font face à des conditions climatiques moins propices à la culture de la vigne (Leeuwen *et al.*, 2008, Ohmart, 2007). Ainsi en Argentine, en Australie, en Chine etc., ce sont des techniques agricoles adaptées à des températures extrêmes qui sont mises en place : paillage des cèpes, arrosage goutte à goutte ou plus impressionnant, l'édification de murs autour de chaque pied sur l'île volcanique de Lanzarote (Gioda, 2008). L'étude et l'analyse de ces modes de culture constitueraient une base de connaissances plus large, qu'il serait intéressant de mobiliser dans le cadre du modèle ADVICLIM, afin de simuler une évolution des pratiques.

# RESSOURCES ET BIBLIOGRAPHIE

---

## Articles

Baggiolini M., 1952, « Les stades repères dans le développement annuel de la vigne et leur utilisation pratique », *Rev. Rom. Agr. Viticult*, 8, 4-6.

Bonnefoy C., Quenol H., Planchon O., Barbeau G., 2010, « Températures et indices bioclimatiques dans le vignoble du Val de Loire dans un contexte de changement climatique », *EchoGéo*.

Celette F., Ripoche A., Gary C., 2010, « WaLIS – A simple model to simulate water partitioning in a crop association: the example of an intercropped vineyard » *Agric. Water Manage*, 97, 1749-1759.

Drogoul A., Vanbergue D., Meurisse T., 2003, « Multi-agent based simulation: Where are the Agents? » *Third International Workshop, MABS 2002 Bologna, Italy, July 15–16, 2002 Revised Papers*.

Dufourcq T., Barraud G., Delpuech X., Debord C., Gaudin R., 2013, « Application du modèle de bilan hydrique WaLIS pour prédire le potentiel hydrique foliaire de tige sur cépage Colombard en Côtes de Gascogne » *Ciência e Técnica Vitivinícola – Volume 28, Proceedings 18th international Symposium of the GiESCO*.

Ferber J., 1995, « Les Systèmes Multi Agents : vers une intelligence collective », *InterEditions, Paris*.

Gioda A., 2008, « Les vignes de Lanzarote », *Agricultures singulières, IRD Editions, fiche 65. 311 – 313*.

Huglin P., 1978. « Nouveau mode d'évaluation des possibilités héliothermiques d'un milieu viticole », *C. R. Aca. Agric.*, p 1117-1126.

Huglin P., Schneider C., 1998, « Biologie et écologie de la vigne », *Tec & Doc Lavoisier, Paris, France*.

Leeuwen C., Garnier C., Agut C., Baculat N., Barbeau G., *et al*, 2008, « Heat requirements for grapevine varieties is essential information to adapt plant material in a changing climate », *VIIth International terroir Congress*.

Morlat R., Barbeau G., Asselin C., 2001, « Facteurs naturels et Humains des terroirs viticoles français : méthode d'étude et valorisation », *Études et Recherches sur les Systèmes Agraires et le Développement*.

Ohmart C., 2007, « Climate Change and Viticulture », *Wines & Vines*.

Shanmuganathan S., Sallis P., Pavesi L., Jarur Muñoz MC., 2008, « Computational intelligence and geo-informatics in viticulture », *Second Asia International Conference on Modelling & Simulation*.

Taillandier P., Grignard A., Gaudou B., Drogoul A., 2014, « Des données géographiques à la simulation à base d'agents : application de la plate-forme GAMA », *Cybergeog : European Journal of Geography*.

Tissot C., Le Tixerant M., Rouan M., 2005, « Le simulateur DAHU : une plateforme de modélisation des activités humaines en zone côtière », *Norois*.

Tissot C., Rouan M., Brosset D., Neethling E., 2010, « Scénarios d'adaptation des vins de terroir au changement climatique à une échelle de temps de 15-30 ans avec l'utilisation d'une plate-forme Multi-agents (SMA) », *Compte rendu final TERADCLIM*, p 38 – 47.

Tissot C., Rouan M., Neethling E., Quenol H., Brosset D., 2014, « Modeling of vine agronomic practices in the context of climate change », *EDP Sciences*.

Tissot C., Neethling E., Rouan M., Kervern S., Barbeau G., Quenol H., en cours, « Simulating environmental impacts on viticultural ecosystems. Case study from the middle Loire Valley (France) »

### **Communications**

Neethling E., Quenol H., Barbeau G., 2014, « Observation et Modélisation du Changement Climatique à l'échelle des Terroirs Viticoles », *Présentation lors du Colloque Euroviti 2014 « Gestion du régime hybride de la vigne »*.

Rouan M., 2015, « A Spatial Data Infrastructure dedicated to scientific research and observation of the coastal environment. » *IODE International Coastal Atlas Network (ICAN) Workshop 7 : Coastal Web Atlases - Supporting Ecosystem Based Management, Apr 2015, Cape Town, South Africa*

Tissot C., Neethling E., Rouan M., Barbeau G., 2015, « Adaptation of cultural practices to climate change; First results and on going development », *Présentation lors du Congrès ADVICLIM à Brighton*.

Tissot C., Neethling E., Rouan M., Barbeau G., 2015, « Modeling environmental impacts on vine behavioral dynamics and vineyard management strategies », *Présentation lors du Congrès ADVICLIM à Brighton*.

### **Webographie**

Documentation PostgreSQL, (<http://www.postgresql.org/docs/9.3/interactive/index.html>)

WikiGAMA, ([https://code.google.com/p/qama-platform/wiki/G\\_Overview](https://code.google.com/p/qama-platform/wiki/G_Overview))

Google Groups GAMA, (groupe fermé nécessitant une inscription)

# TABLE DES ILLUSTRATIONS

## Table des tableaux

Tableau 1: Correspondance entre le vocabulaire de GAMA et celui de NetLogo (D'après Taillandier et al, 2014)	12
Tableau 2: Caractéristiques des enquêtes approfondies et indépendantes réalisées auprès des viticulteurs (D'après Tissot et al, 2010)	13
Tableau 3: Relations entre stades et indices bioclimatiques en	17
Tableau 4 : Données utilisées pour la mise en place du modèle	23
Tableau 5: Exemples comparatifs du temps nécessaire au fonctionnement du modèle ADVICLIM avec ou sans connexion à la base de données PostGreSQL, en moyenne sur une année simulée	30
Tableau 6: Détermination des curve number (taux de ruissèlement) selon le type de sol et l'entretien du sol (D'après Romero et al., 2007)	46

## Table des figures

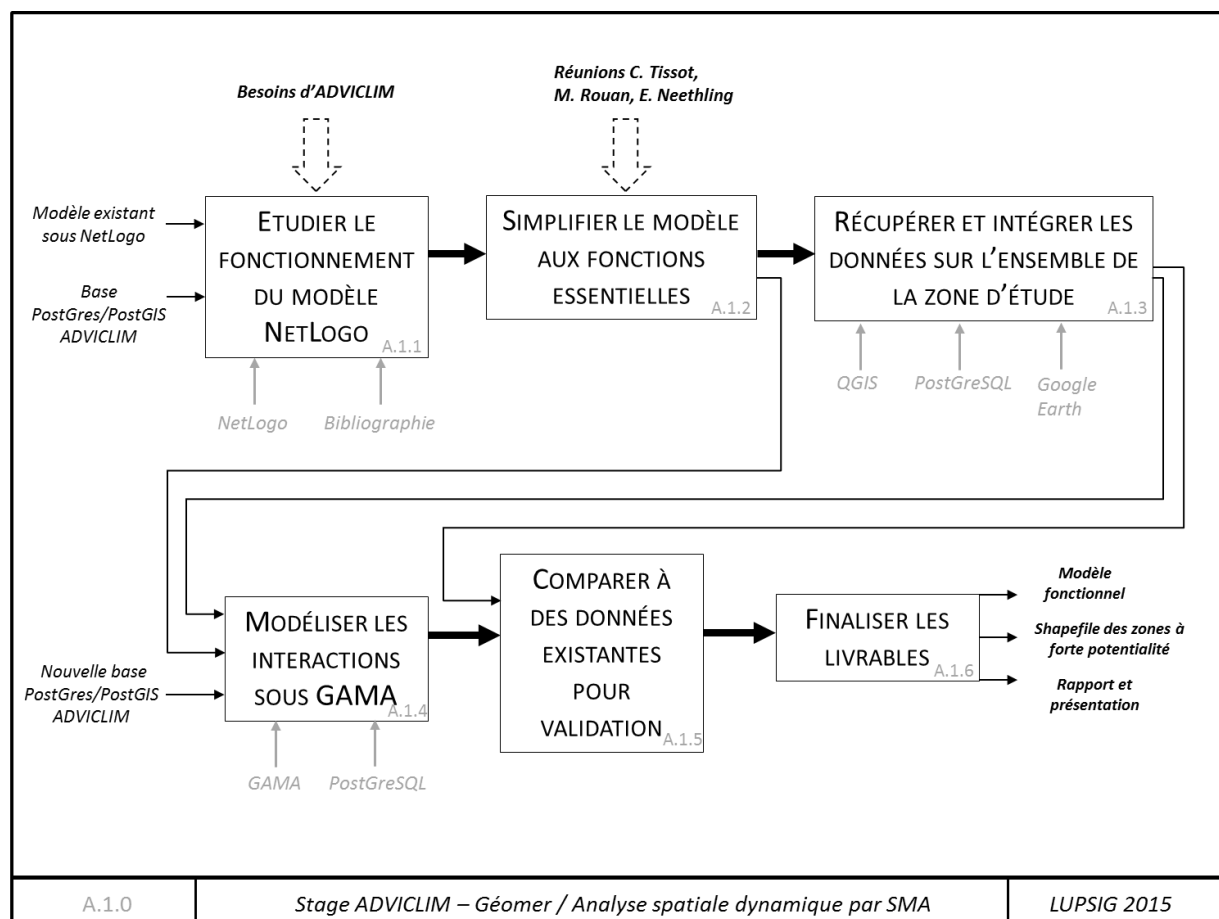
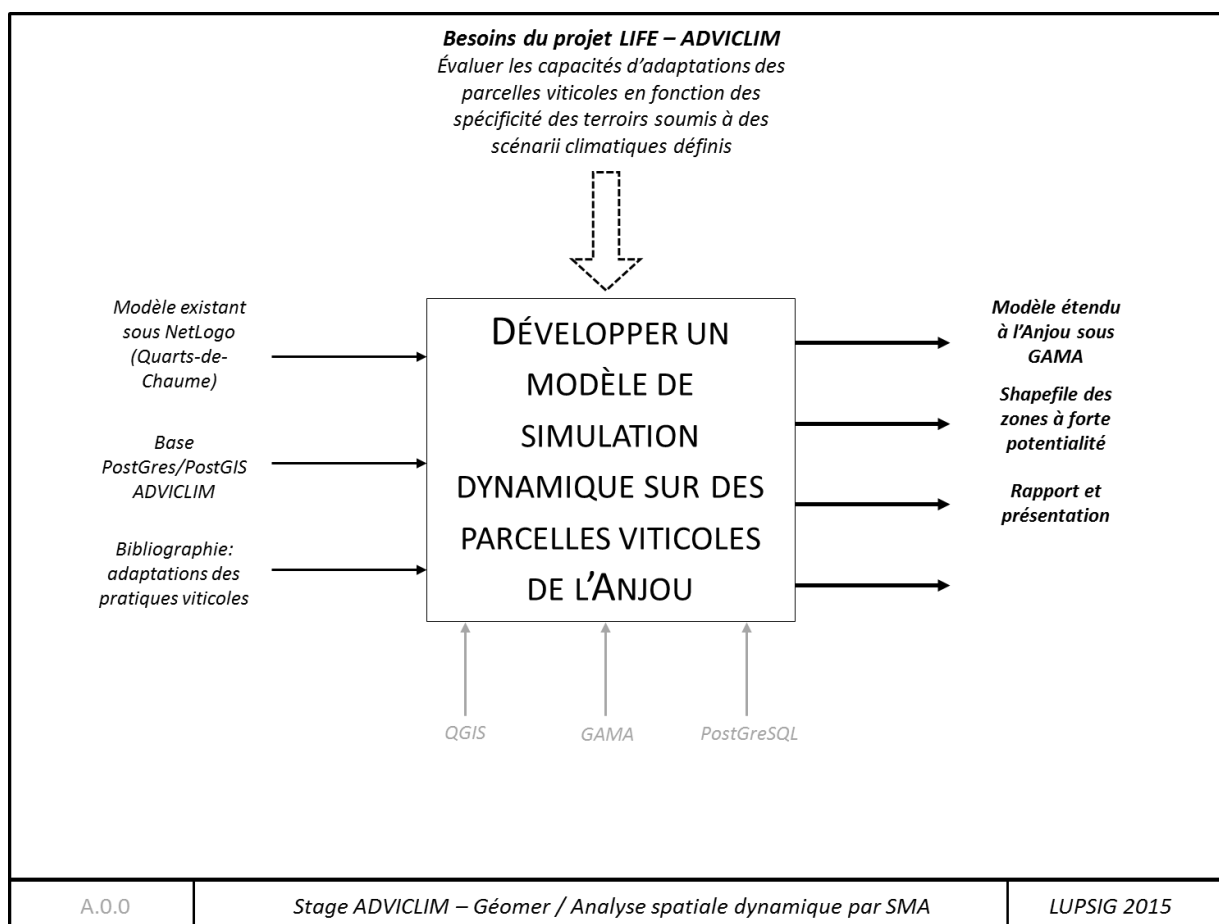
Figure 1: Evolution temporelle du projet et contexte du stage, de 2008 à 2019	8
Figure 2: Exemples de courbes de profils climatiques sur deux années types et leurs incidences sur les principaux stades d'évolution de la vigne, en fonction des indices bioclimatiques de degrés-jours et de Huglin (D'après Tissot et al, 2015)	14
Figure 3: Les stades repères de la vigne (D'après Baggiolini, 1952)	16
Figure 4: <b>a.</b> Formule permettant d'obtenir le cumul de degrés-jours <b>b.</b> Formule permettant d'obtenir l'indice de Huglin. Avec T max : température maximale, T min : température minimale, T moy : température moyenne k : coefficient de longueur du jour.	17
Figure 5: Capture d'écran du modèle développé par le LETG sous NetLogo sur la zone Quarts-de-Chaume	19
Figure 6: Diagramme de la base PostGreSQL/PostGIS couplé avec le modèle SEVE (Tissot et al, 2010)	19
Figure 7: Représentation schématique de l'utilisation des différents logiciels mis en jeu et de leurs interactions	22
Figure 8: Ensemble des éléments composant la plateforme GAMA	26
Figure 9: Schématisation des échanges d'informations au sein du modèle d'un point de vue opérationnel	26
Figure 10: Schématisation des échanges d'informations au sein du modèle d'un point de vue opérationnel	28
Figure 11: Initialisation des paramètres des connexions à la base de données dans le modèle ADVICLIM sous GAMA	29
Figure 12: Exemple de requêtes faisant appel à une base PostGres en GAML	29
Figure 13: Exemple des informations retournées par la console à l'utilisateur lors du premier tour du modèle sous GAMA	30
Figure 14: Attribution des capteurs par parcelles en fonction des relations de voisinages. <b>a.</b> Capture d'écran du résultat de l'algorithme « Distance to the nearest hub » de QGIS <b>b.</b> Résultat de l'analyse des relations capteurs/parcelles sous QGIS	32
Figure 15: Fonction SQL intégrée au script en GAML régissant la création d'une table source intègre pour les températures relevées par chaque capteur	35
Figure 16: Evolution des indices bioclimatiques au cours d'une année agronomique - Graphique généré sous GAMA	36
Figure 17: Représentation schématique du flux d'informations régissant l'évolution de la vigne sous GAMA	37
Figure 18: Capture d'écran de la création d'un agrégat sous GAMA	39
Figure 19: Schématisation de la fonction d'agrégation développée sous GAMA pour le projet ADVICLIM	40

<i>Figure 20:Influence de la topographie sur l'évolution des températures journalières estivales (D'après E. Gratia)</i>	43
<i>Figure 21: Triangle des textures pédologiques (classification américaine)</i>	47
<i>Figure 22: Ensemble des diagrammes descriptifs des processus fonctionnels mis en place dans le cadre du stage ADVICLIM</i>	57
<i>Figure 23: Diagrammes de Gantt réalisés dans le cadre du projet ADVICLIM a. Planning prédictif b. Planning réalisé</i>	58
<i>Figure 24: Représentation graphique de la distribution de la charge de travail en pourcentage durant le stage ADVICLIM</i>	59

## Table des cartes

<i>Carte 1: Répartition mondiale des sites expérimentaux, 15 vignobles au total, 11 pays représentés (D'après TERADCLIM, 2011)</i>	8
<i>Carte 2: Répartition des 5 sites pilotes au niveau européen, déployés sur 4 pays.</i>	9
<i>Carte 3: Situation et caractéristiques des vignobles du Val de Loire au sein des vignobles français</i>	21
<i>Carte 4: Zone d'intérêt du Coteaux du Layon</i>	21
<i>Carte 5: Situation des différents capteurs météorologiques en Anjou</i>	42
<i>Carte 6: Environnement du parcellaire sur l'Anjou selon la classification Corine Land Cover 2012</i>	44
<i>Carte 7: Mise en perspective de la zone test étudiée lors du stage avec la zone d'étude totale potentielle</i>	48

## Annexe I : Organisation du stage





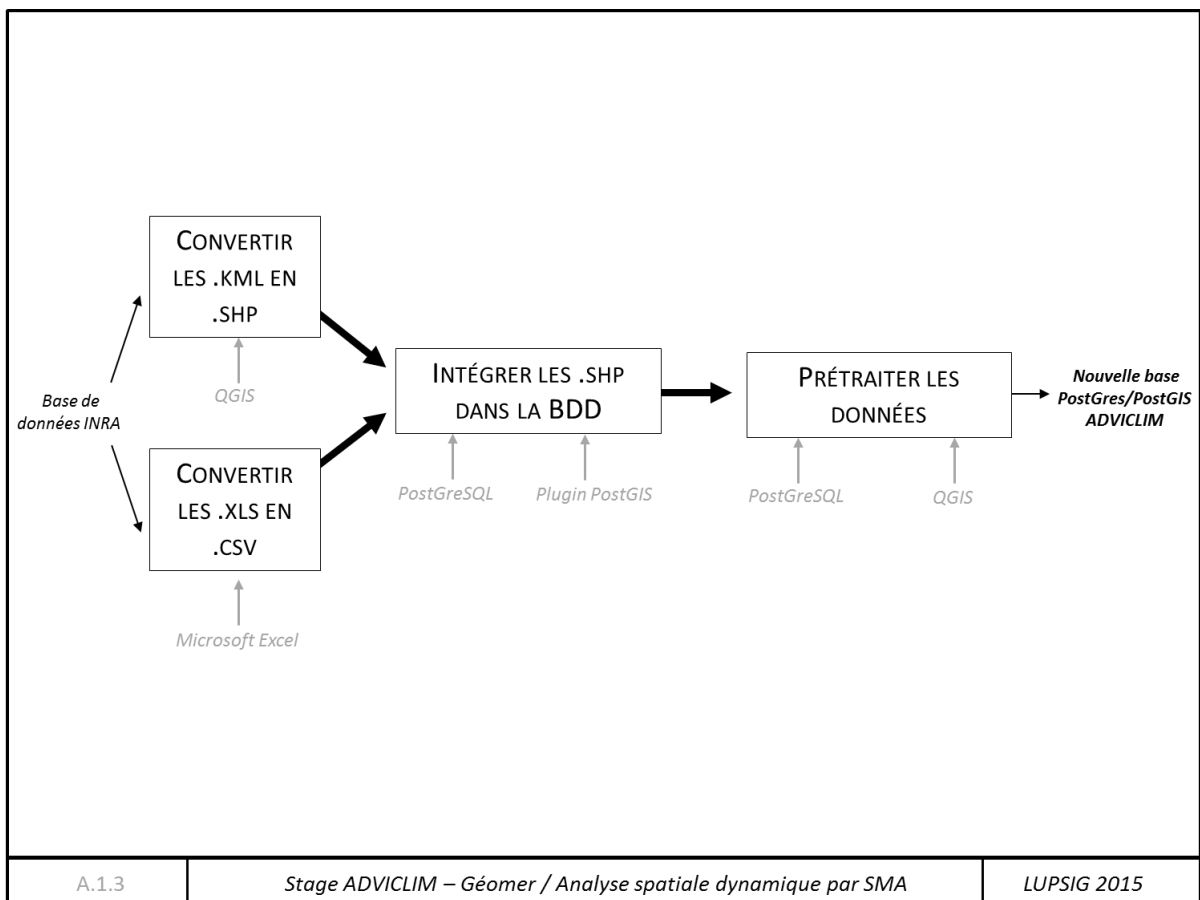
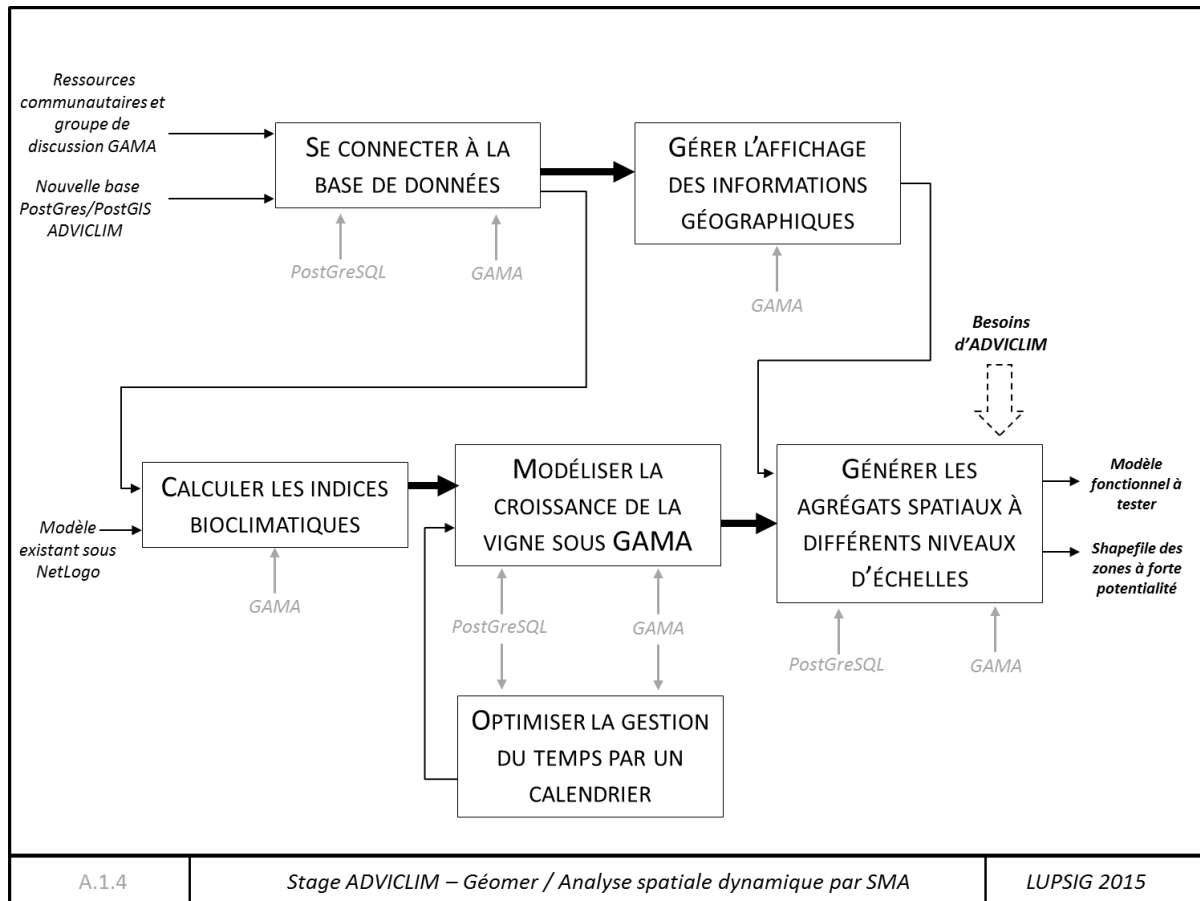


Figure 21: Ensemble des diagrammes descriptifs des processus fonctionnels mis en place dans le cadre du stage ADVICLIM

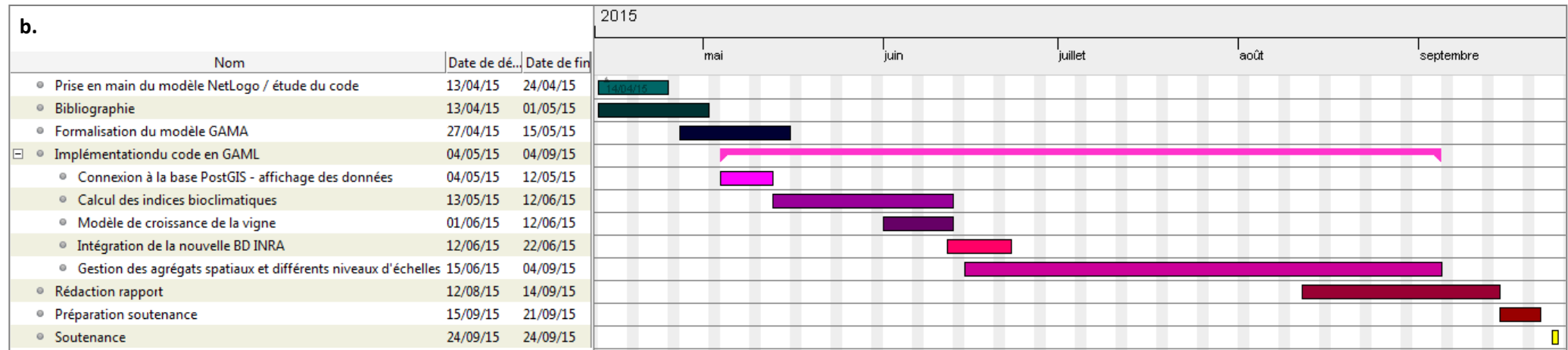
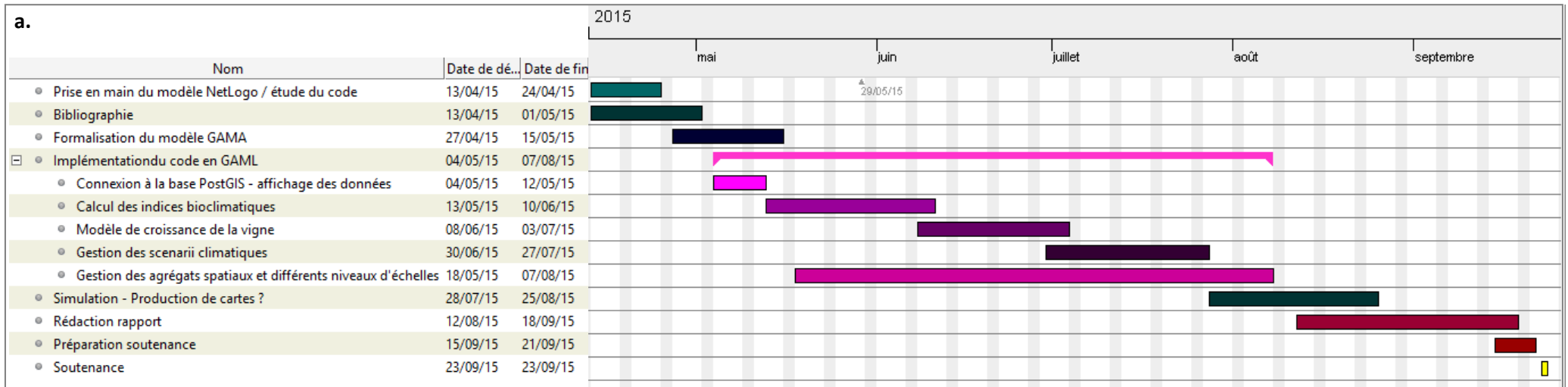


Figure 22: Diagrammes de Gantt réalisés dans le cadre du projet ADVICLIM **a.** Planning prédictif **b.** Planning réalisé

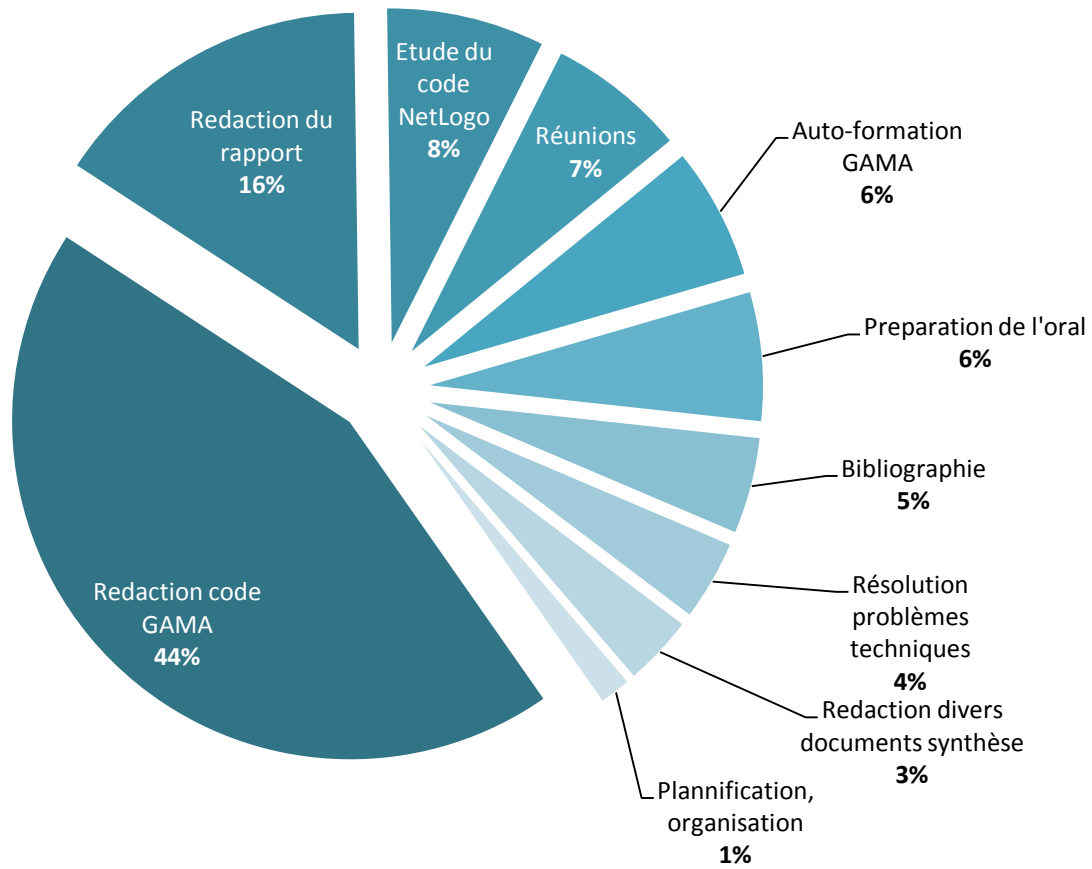
**REPARTITION HORAIRE DE LA CHARGE DE TRAVAIL AU COURS DU STAGE**

Figure 23: Représentation graphique de la distribution de la charge de travail en pourcentage durant le stage ADVICLIM

## Annexe II : Code complet

```

model chartsIndices

global {

    var BOUNDS type:map //définition des paramètres de connexion pour Les Limites
d'affichage de La carte
    init:
    ['host'::'localhost',
     'dbtype'::'postgis',
     'database'::'DahuVigne_layon',
     'port'::'5432',
     'user'::'postgres',
     'passwd'::'*****',
     "select":: "SELECT ST_AsBinary(geom) as geom from enveloppe",
     'srid'::'2154'];

    var PARAMS type:map //définition des paramètres de connexion à La base de données
    init:
    ['host'::'localhost',
     'dbtype'::'postgis',
     'database'::'DahuVigne_layon',
     'port'::'5432',
     'user'::'postgres',
     'passwd':: '*****',
     'srid'::'2154'];

    //requêtes de sélection des couches à afficher
    var parcelles_layon type: string init: "SELECT numero, nom_commun, commune,
conseil, code_cepag, orientatio, nomviti, codeviti, domaine, vins,
capteurshu, ST_AsBinary(centroid) as centroid, ST_AsBinary(geom) as geom
FROM parcelle_utb_layon_bdtopo" ;
    var tiny_tag type: string init: "SELECT nom, ST_AsBinary(geom) as geom,
code FROM buffer_capteurs_coteaux_du_layon" ;
    var enveloppe type: string init: "SELECT ST_AsBinary(geom) as geom FROM
enveloppe" ;
    // définition des Limites
    geometry shape <- enveloppe(BOUNDS);

    // aller chercher Les données dans La BD pour en faire une Liste déroulante
    var annee_dep type: string init: "2013" among: ['1988', '1989', '1990',
'1991', '1992', '1993', '1994', '1995', '1996', '1997', '1998', '1999',
'2000', '2001', '2002', '2003', '2004', '2005', '2006', '2007', '2008',
'2009', '2010', '2011', '2012', '2013', '2014'] ;
    // déclaration des différentes variables globales
    list Periode ;
    string date_j;
    string date_dep;
    string date_fin;
    int date_t ;
    string month;
    int Daycounter;
    float step <- 1#d;
    // températures minimales et maximales récupérées en base par La suite
    list tmintmax;
    float tmin;
    float tmax;
    // indice calculés à chaque pas de temps à partir des requêtes de températures

```

```

float indice_dj <- 0;
float indice_huglin <- 0;
// booléen indiquant s'il est nécessaire de calculer l'indice de Huglin ou pas (à
// partir du 1er avril)
bool Huglin <- false ;
// maps et listes régissant la mise en place des seuils et des correspondance
// indices/cépages
matrix lesCepages;
string cepagesExist;

map<string, list<int>> mapStadeDJ;
list listeStadeDJ;
list seuilCepageDJ;//

list listeStadeHug;
list listeCepageHug;

int idDepList;//

// code couleur régissant la couleur à donner aux différents stades
int code_couleur <-1;
list couleur;
string codes_stade <- 'ABCDEFGHIJKLMNO'; // La variable contient la liste des
// codes stades

bool NouvelleAnnee<-false;

int nbCapteurs;
string capteurG;

// ensemble des requêtes pour les variables globales
string querySeuilDJ <- "SELECT indice_degre_jour FROM
relation_cepage_stade WHERE code_cepage = ? and code_stade = ? " ;
string querySeuilHUG <- "SELECT indice_de_huglin FROM
relation_cepage_stade WHERE code_cepage = ? and code_stade = ? " ;
string queryPeriode <- "SELECT date_layon FROM temp_layon WHERE
date_layon between ?::date and ?::date";
string queryDate_t <- "SELECT CAST (EXTRACT(EPOCH FROM ?::timestamp)
AS INTEGER)";
string queryListeStade <- "SELECT code_cepage, indice_degre_jour FROM
relation_cepage_stade ORDER BY code_cepage, indice_degre_jour";
string queryMonth <- "SELECT to_char(?::date, 'Month YYYY')";
string queryCouleur <- "SELECT couleur_stade::integer FROM
stades_phenologiques ORDER BY code_stade" ;
string queryListeStadeDJ <- "SELECT code_cepage, indice_degre_jour
FROM relation_cepage_stade ORDER BY code_cepage, indice_degre_jour";
string queryListeStadeHug <- "SELECT indice_de_huglin FROM
relation_cepage_stade WHERE code_stade = 'N'";
string queryListeCepageHug <- "SELECT code_cepage FROM
relation_cepage_stade WHERE code_stade = 'N'";
string queryCepages <- "SELECT DISTINCT code_cepag FROM
parcelle_utb_layon";
string queryTminTmax <- "SELECT * FROM tempe_temp";
string queryCapteurs <- "SELECT count(code) FROM (SELECT
DISTINCT code FROM temp_layon) AS nbCapteurs ;" ;

init{

  create species: dbCaller number: 1

```

```

{
  create species:tinyTag from: list(self select [params:: PARAMS,
  select:: tiny_tag]) with: [
    nom::'nom',
    code :: 'code',
    shape::'geom'];

  create species:parcelles from: list(self select [params:: PARAMS ,
  select:: parcelles_layon]) with:[
    numero::'numero',
    nom :: 'nom_commun',
    commune::'commune',
    conseil::'conseil',
    cepage::'code_cepag',
    orientation::'orientatio',
    viticulteur::'nomviti',
    domaine::'domaine',
    vins :: 'vins',
    capteur :: 'capteurshu',
    centroid :: 'centroid',
    shape::'geom'];

  create species: fondDeCarte from: list (self select [params:: PARAMS,
  select:: enveloppe]) with: [shape::'geom'];
}

//initialisation de la date
write annee_dep;
date_dep<- "" + annee_dep + "-10-31";
date_fin <- "" + (annee_dep as_int 10 +1) + "-10-31";

}

reflex Periode when: cycle=0 {ask dbCaller{
  //sélection de la période choisie par l'utilisateur
  Periode <- (self select(params: PARAMS, select: queryPeriode, values:
  [date_dep, date_fin]));
  //sélection des températures sur la période
  tmintmax <- (self select(params:: PARAMS, select:: queryTminTmax))[2];
  write tmintmax ;
  }
  write Periode ;
}

reflex init when: cycle=0{
  //récupération en base de toutes les données cépages / indices
  ask dbCaller{
    nbCapteurs <- (self select(params: PARAMS, select:
    queryCapteurs))[2][0][0];
    write "capteurs : " + nbCapteurs;

    lesCepages<- (self select(params: PARAMS, select:
    queryCepages))[2];

    listeStadeDJ<-(self select(params: PARAMS, select:
    queryListeStadeDJ))[2];

    couleur <- (self select(params: PARAMS, select:
    queryCouleur))[2];
  }
}

```

```

    listeStadeHug <- (self select(params: PARAMS, select:
    queryListeStadeHug))[2];

    listeCepageHug <- (self select(params: PARAMS, select:
    queryListeCepageHug))[2];
  }
  //boucle de remplissage de la liste des seuils pour les indices bioclimatiques en
  fonction des cépages
  loop i from: 0 to: length(rows_list(lesCepages))[0]-1{
    cepagesExist <- rows_list(lesCepages)[0][i];

    idDepList <- listeStadeDJ index_of [cepagesExist,0];

    seuilCepageDJ <- seuilCepageDJ + copy_between
    (listeStadeDJ, idDepList, (idDepList + (length(codes_stade))));
  }
}
//récupération de la date du jour grâce à l'index de la liste de la période
reflex leTempsQuiPasse{
  date_j <- "" + Periode[2][cycle+1][0] + "" ;
  write date_j;
  Daycounter <- Daycounter +1;
}
//passage du booléen indiquant que l'indice de Huglin doit être calculé à "vrai"
reflex Huglin when: copy_between(date_j, 6,11) = '04-01'{
  Huglin <- true;
  write "passage à l'indice de Huglin ";
}

//réinitialisation des variables globales après vendange au 31 octobre
reflex nouvelleAnnee when: copy_between(date_j, 6,11) = '10-31'{
  Huglin <- false;
  indice_dj <- 0;
  indice_huglin <- 0;
  code_couleur <- 1;
  NouvelleAnnee <- true;
  Daycounter <- 0;
}

reflex cluster when: cycle = 365{
  ask parcelles{
    if capteur = '13'{
      capteurG <- capteur;
    }
  }
  ask dbCaller{
    do executeUpdate(params: PARAMS, updateComm: "DROP TABLE parc_test;");

    do executeUpdate(params: PARAMS,
    updateComm: "CREATE TABLE parc_test AS SELECT ST_Union(geom) as geom
    FROM parcelle_utb_layon_bdtopo WHERE capteurshu = '"+capteurG+"'");

    string queryUnion <- "SELECT ST_AsBinary(geom) as geom FROM
    parc_test";

    create parc from: list (self select [params:: PARAMS, select::
    queryUnion]) with: [shape:: 'geom']{

```

```

    }
  }
}

entities {
  species parcelles {
    var numero type: int;
    var nom type: string;
    var commune type: string;
    var cepage type: string;
    var domaine type: string;
    var viticulteur type: string;
    var vins type: string;
    string capteur;

    float valeurDJ; // variable Locale permettant de sélectionner La valeur de
    // compteur pour Le capteur concerné dans La MAP des compteurs
    float valeurH; // GAMA n'accepte pas Le calcul direct, il faut passer par
    // La variable

    string stade_actuel <- "A"; // La variable contient Le code du stade en
    // cours
    string stade_suivant <- "B"; // La variable contient Le code du stade
    // suivant, permettant de définir Le seuil à atteindre

    // Les valeurs d'id permet d'incrémenter Les différentes Listes d'évolution
    int id_code <- 0 ;
    int idDepList;
    int idHug;

    int code_couleur <-1;

    int seuil_suivant_dj <- 10000; // La variable contient Le seuil en cours
    // (qui doit être atteint pour passer au prochain stade) requête SQL
    int seuil_suivant_hug <- 10000; // Les deux seuils sont initialisés à
    // 10000, valeur impossible à atteindre quel que soit Le contexte, afin de ne
    // pas déclencher artificiellement Le reflex evolue
    bool changementStade <- false;
    bool vendange<-false;

    var color type: rgb init: rgb(135, 89, 33) ;

    reflex initStade when: cycle=0 or NouvelleAnnee = true {
      // Les id sont initialisés à La valeur voulue: celle qui correspond au seuil
      // A pour Le cépage de La parcelle.
      idDepList<- seuilCepageDJ index_of [cepage,0];
      seuil_suivant_dj<-seuilCepageDJ[id_code+idDepList+1][1];

      // Pour Les cépages 2 Listes indépendantes de même Longueur sont
      // gérées dans Le même temps.
      idHug<-listeCepageHug index_of [cepage];
      seuil_suivant_hug <- listeStadeHug[idHug][0];

      vendange<- false;
    }
  }
}

```



```

}

reflex capteurs{ask tinyTag{
//Les valeurs obtenues pour chaque capteurs sont recherchées dans La MAP des
capteurs en fonction du code qui lui est attribué
  myself.valeurDJ <-counterCapteurDJ[myself.capteur] ;
  myself.valeurH <-counterCapteurH[myself.capteur] ;
}
}

// Le principe est le même pour toutes les "Listes" évolutives, l'id calculé ou défini
précédemment permet de sélectionner la valeur adéquate
reflex evolve when:((valeurDJ >= seuil_suivant_dj and stade_actuel
!='0'and stade_actuel !='N') or (valeurH >= seuil_suivant_hug and
stade_actuel !='N')) {
  write "pour le capteur " + capteur + " parcelles: "+ numero;

  id_code <- id_code + 1; // incrémentation de l'id du stade_actuel
  stade_actuel <- codes_stade at (id_code); // sélection du code
correspondant à l'id dans la liste codes_stades
  write stade_actuel;
  stade_suivant <- codes_stade at (id_code+1) ; // sélection du
code correspondant à l'id du stade suivant dans la liste codes_stades

  if stade_suivant = 'N'{
    seuil_suivant_hug <- listeStadeHug[idHug][0] ;
  }
  if stade_actuel = 'N'{
    write "vendange envisageable " + capteur;
  }
  else {
    seuil_suivant_dj <-
seuilCepageDJ[id_code+idDepList+1][1];
  }

//changement de la couleur de la parcelle en fonction du stade
changementStade <- true;

code_couleur <- couleur[id_code][0];

if (changementStade = true ){
  if code_couleur = 1{
    color <- rgb(135, 89, 33) ;}
  if code_couleur = 2{
    color <- rgb(107, 133, 33);}
  if code_couleur = 3{
    color <- rgb(187, 204, 8);}
  if code_couleur = 4{
    color <- rgb(140, 166, 46);}
  if code_couleur = 5{
    color <- rgb(158, 105, 96);}
  if code_couleur = 6{
    color <- rgb(122, 68, 63);}
  }
}

// réinitialisation des variables après le 31 octobre (date temporaire)
reflex vendange when: copy_between(date_j, 6,11) = '10-31'{
  vendange <-true;
  color <- rgb(135, 89, 33) ;
}

```

```

ask tinyTag{
  dj_counter <- 0;
  huglin_counter <- 0;
  counterD<- nil;
  counterH<- nil;
  counterCapteurDJ<- nil;
  counterCapteurH<- nil;}

  stade_actuel <- "A";
  stade_suivant <- "B";
  id_code <- 0 ;
  seuil_suivant_dj <- 10000;
}

aspect base
{
  draw shape color: color ;
}
}

species tinyTag {

  var nom type: string ;
  var code type: string ;

  float dj_counter <- 0; // idem
  float huglin_counter <- 0;

  pair<float> counterD;
  pair<float> counterH;
  var counterCapteurDJ type: map;
  var counterCapteurH type: map;

  list Ltmintmax;

  var color type: rgb init: rgb('black') ;

  aspect base
  {
    draw shape border: rgb('white') color: color ;
  }

  reflex init when: cycle=0{//récupération et segmentation de la liste
générale des températures en fonction du code de capteur
    Ltmintmax<- copy_between (tmintmax, ((code as_int 10) * 366)-
366, ((code as_int 10) * 366)-1);

    write "capteur " + code + Ltmintmax ;
  }

  //Calcul des degrés jours: jours dont la température dépasse 10 degrés (t minimale
d'activité physio)
  //Le reflex de calcul s'effectue jusqu'au stade N, ensuite indice_huglin, compteur lancé à
partir du 1er avril

  reflex calcul{
    tmin<-Ltmintmax[Daycounter][0];
    tmax<-Ltmintmax[Daycounter][1];
  }
}

```

```

indice_dj <- ((tmin + tmax)/2)-10;

if (indice_dj >=0) {
  dj_counter <- dj_counter + indice_dj;}

if Huglin = true{
indice_huglin <- (((((tmin + tmax)/2)-10) + (tmax - 10))/2);

if (indice_huglin >=0){
  huglin_counter <- huglin_counter + indice_huglin;
}
}

counterD <- [code, dj_counter] ;
counterCapteurDJ <- tinyTag collect each.counterD ; // Les
valeurs de DJ sont emmagasinées dans une MAP dont la clé est le code
capteur

if Huglin = true{
  counterH <- [code, huglin_counter];
  counterCapteurH <- tinyTag collect each.counterH;} // idem

ask parcelles{
  changementStade <- false;}
}
}

```

*// La species parc représente les agrégats spatiaux. Elle est créée à partir du reflex cluster global. Seul ces caractéristiques physiques sont ici contrôlées*

```

species parc{
  rgb color <- rgb('red');
  aspect geometry {
    draw shape border: rgb('red') color: color;
  }
}

species dbCaller skills:[SQLSKILL]{
}

species fondDeCarte{
  var color type: rgb init: rgb(192, 192, 192) ;

  aspect base
  {
    draw shape color: color ;
  }
}
}
}

```

```

experiment station type: gui {

  parameter "Année de début de simulation: " var: annee_dep ;

  output {
    display map {// toutes les species sont représentées, à part le dbCaller
      species fondDeCarte aspect: base ;
      species parcelles aspect: base ;
      species parc aspect: geometry transparency: 0.5;
      species tinyTag aspect: base ;
    }

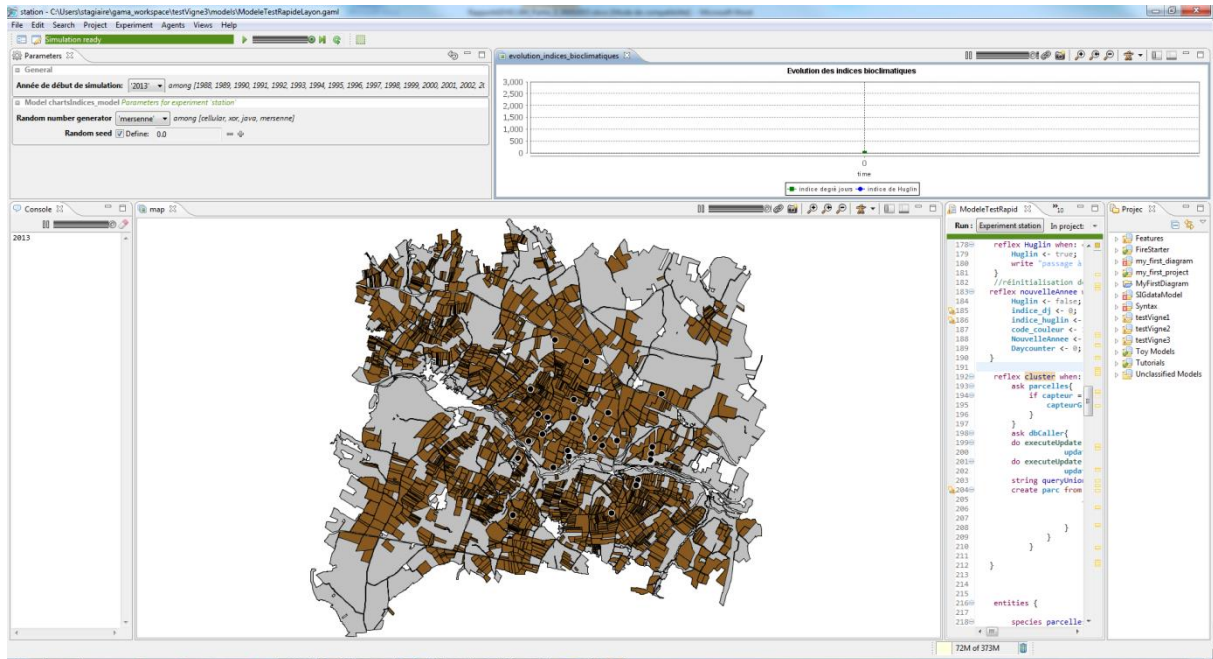
    display evolution_indices_bioclimatiques refresh_every: 5 {
      // représentation graphique de l'évolution des indices bioclimatiques
      chart "Evolution des indices bioclimatiques" type: series
      y_range:({0,3000}) {

        data "indice degré jours" value: mean(tinyTag collect
          each.dj_counter) color: #green; // moyenne sur l'ensemble
          des capteurs

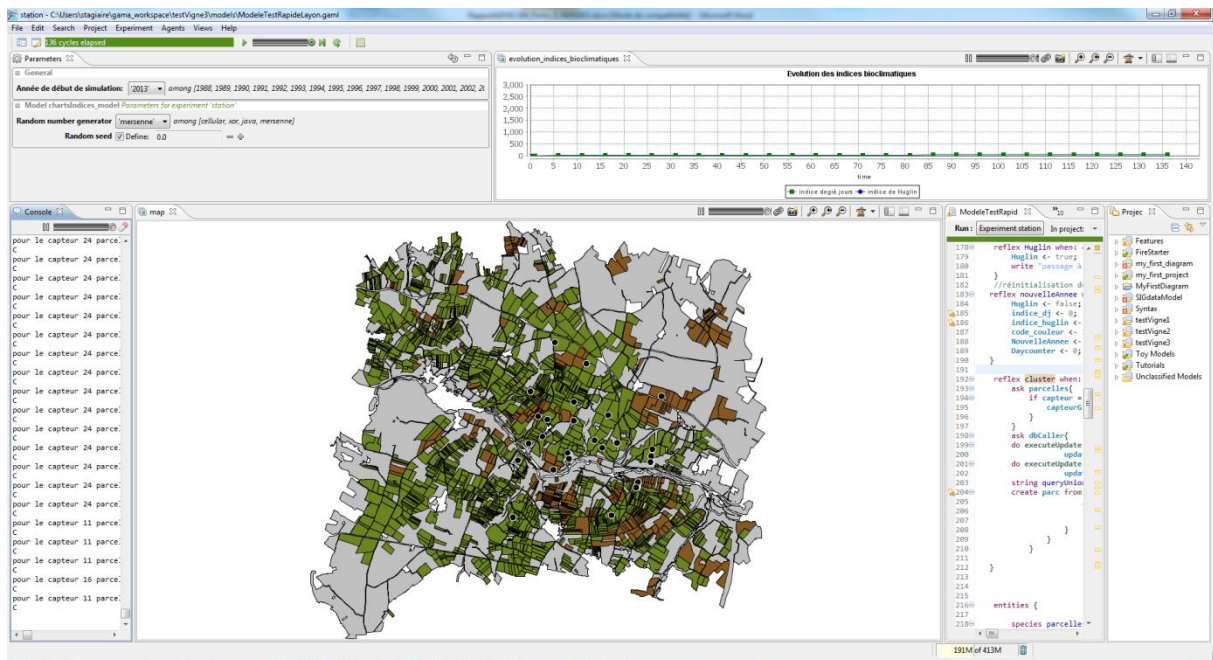
        data "indice de Huglin" value: mean(tinyTag collect
          each.huglin_counter) color: #blue; // moyenne sur
          l'ensemble des capteurs
      }
    }
  }
}

```

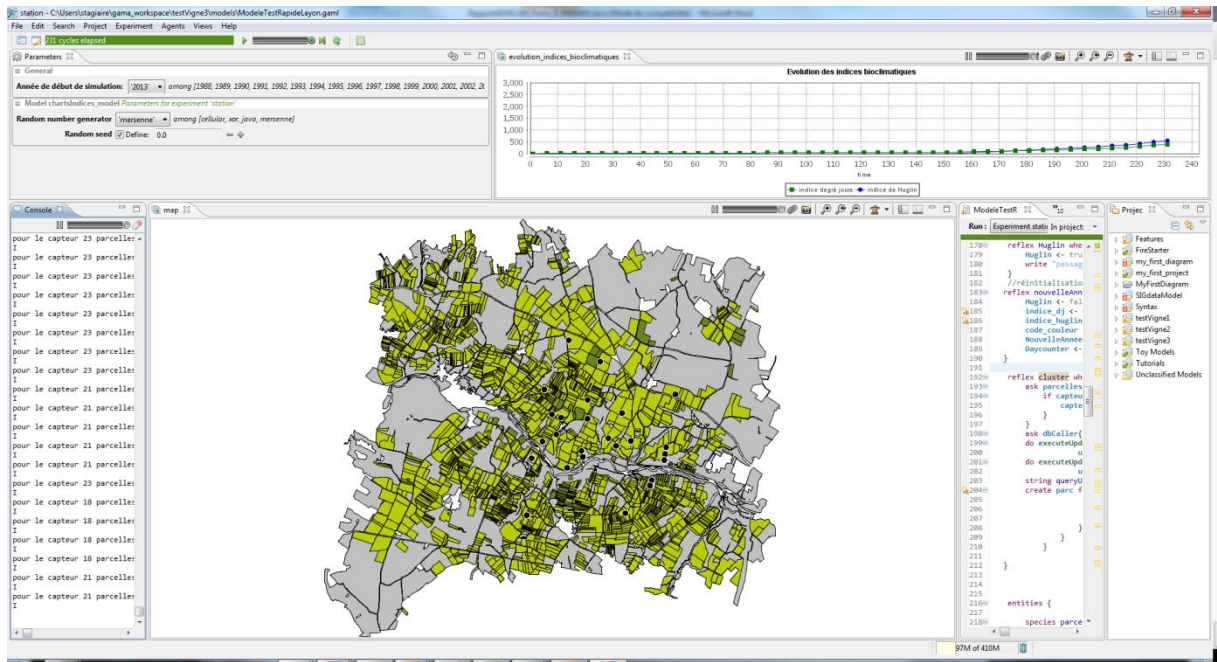
## Annexe III : Exemple d'une année agronomique



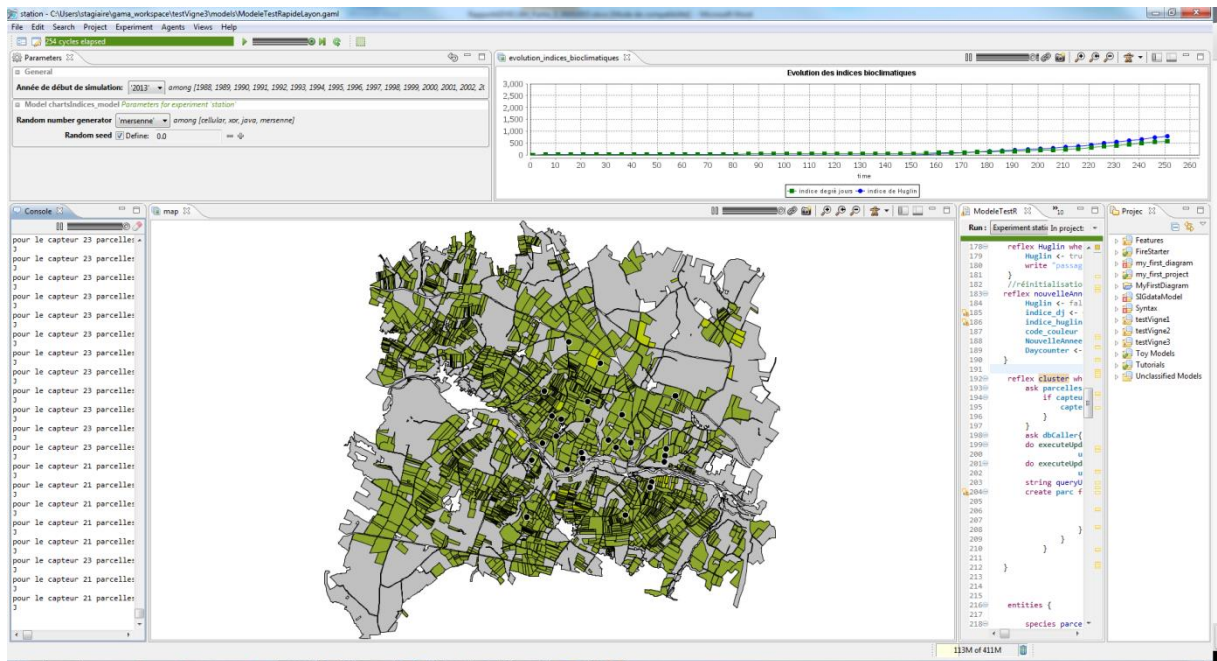
Capture modèle ADVICLIM 1: Cycle 0, début de l'année agronomique 1er Novembre



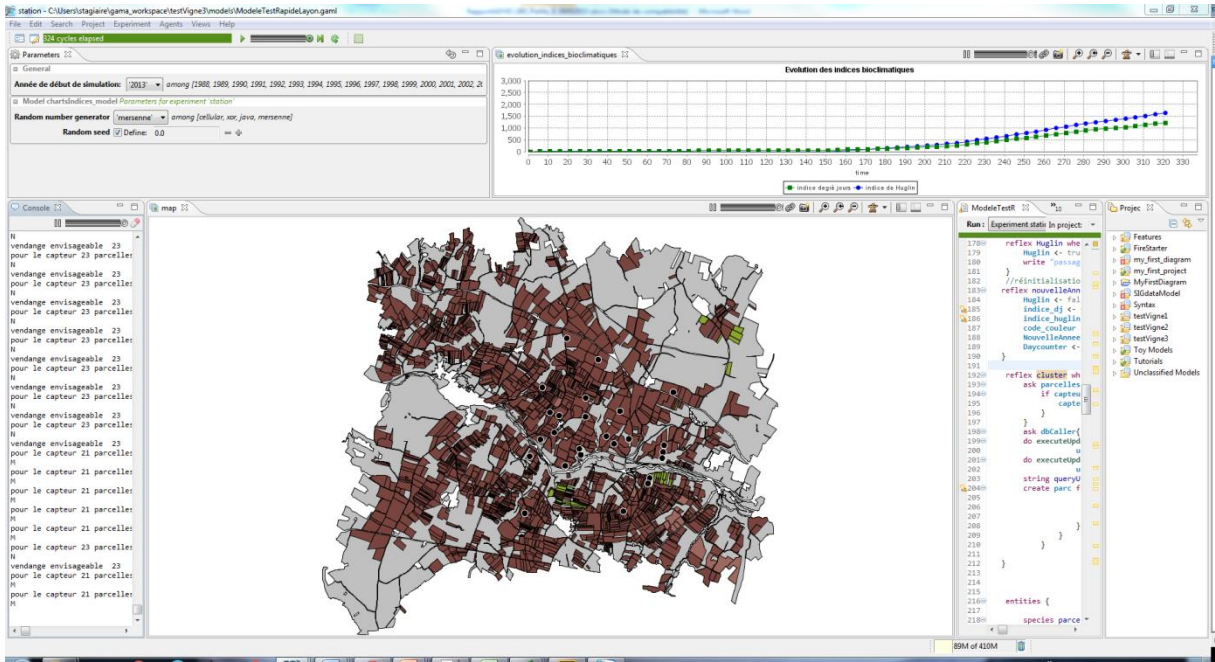
Capture modèle ADVICLIM 2: Une grande partie des parcelles en débourrement, mi-mars



Capture modèle ADVICLIM 3: Floraison, mi-juin



Capture modèle ADVICLIM 4: Nouaison, mi-juillet



Capture modèle ADVICLIM 5: Véraison et maturité, mi-septembre

## Annexe IV : Commandes SQL

Ne sont répertoriées ici que les commandes SQL les plus complexes.

### Boucle de récupération de la température

```
DELETE FROM tempe_temp;

CREATE OR REPLACE FUNCTION table_correct() RETURNS SETOF tempe_temp AS

$BODY$

DECLARE i tempe_temp%rowtype;

BEGIN

    FOR j IN 1..25 LOOP
        FOR i IN 0..365 LOOP

            --result =
            RETURN QUERY SELECT
            CASE
                WHEN (SELECT EXISTS (SELECT tmin FROM temp_layon WHERE date_layon =
                ('2013-11-01'::date + i) AND code = j))
                AND (SELECT tmin NOTNULL FROM temp_layon WHERE date_layon = ('2013-11-
                01'::date + i) AND code = j)

                THEN (SELECT tmin FROM temp_layon WHERE date_layon = ('2013-11-01'
                ::date + i) AND code = j)

                ELSE (SELECT tmin FROM meteo WHERE date = ('2013-11-01'::date + i))

            END,

            CASE
                WHEN (SELECT EXISTS (SELECT tmax FROM temp_layon WHERE date_layon =
                ('2013-11-01'::date + i) AND code = j))
                AND (SELECT tmin NOTNULL FROM temp_layon WHERE date_layon = ('2013-11-
                01'::date + i) AND code = j)

                THEN (SELECT tmax FROM temp_layon WHERE date_layon = ('2013-11-01'
                ::date + i) AND code = j)

                ELSE (SELECT tmax FROM meteo WHERE date = ('2013-11-01'::date + i))

            END;

        END LOOP;
    END LOOP;

RETURN;

END

$BODY$

language plpgsql;
```



```
INSERT INTO tempe_temp SELECT * FROM table_correct();
```

## Capteur le plus proche

```
SELECT code, agid
FROM (
    SELECT a.gid AS agid, code, b.gid AS bgid, rank() OVER(PARTITION BY a.gid ORDER BY
    st_distance(a.centroid, b.centroid)) AS rang
    FROM parcelle_utb_layon a, buffer_capteurs_coteaux_du_layon b
)
    classement
WHERE rang=1
ORDER BY code
```

## Agrégation en fonction d'un critère

```
CREATE TABLE parc_test AS
SELECT
    ST_Union (geom) AS the_geom
    FROM parcelle_utb_layon_bdtopo WHERE capteurshu = '5'
GROUP BY capteurshu;
```