*5*

Version

—

—

—

19/05/2019 : Reprises Lena + Envoi à Hélène

24/10/2019 : Reprises Hélène + Envoi à Anne & Lucie

27/10/2019 : Correction des mockups qui produisaient des pdf buggés

**S ommaire**

**Introduction . . . . . . . . . . . . . . . . . . . . . . . . .**

**Capter les sorties de SimFeodal . . . . . . . . . . . . . . .**

**2**

**3**

3

**5.1**

5.1.1

5.1.2

Masse des données . . . . . . . . . . . . . . . . . . . .

Tenir compte de la stochasticité du modèle - les répli-

cations . . . . . . . . . . . . . . .

Des réplications aux expériences Des données aux indicateurs . .

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

**.**

.

.

.

**.**

.

.

.

**.**

.

.

.

.

.

.

**.**

7

9

10

**12**

12

16

18

23

27

32

**35**

35

49

5.1.3

5.1.4

**5.2**

**Comment explorer les sorties de SimFeodal ? . . . .**

5.2.1

5.2.2

5.2.3

5.2.4

5.2.5

5.2.6

Observer les simulations en direct ou *a posteriori* . .

Générer les indicateurs . . . . . . . . . . . . . . . . Organiser les indicateurs en rapports paramétrables Organiser les rapports : les *dashboards* . . . . . . . . Interagir avec les rapports : exploration interactive . Explorer en comparant : la plateforme SimEDB . . .

**5.3**

**Organiser les données . . . . . . . . . . . . . . . . . . .**

* + 1. Assurer la capacité d’interrogation des données . . . .
    2. Structuration des données de SimFeodal . . . . . . . . **Une plate-forme d’exploration de données de simulations : SimEDB . . . . . . . . . . . . . . . . . . . . . . . . . . . .** 5.4.1 Contraintes . . . . . . . . . . . . . . . . . . . . . . . .

5.4.2 Construire une plate-forme interactive pour l’explora- tion de sorties de simulation . . . . . . . . . . . . . . **Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . .**

**5.4**

**57**

57

67

**83**

1

2019-10-27

**Explorer visuellement des données de simulation massives pour analyser le comportement d’un modèle.**

Chapitre 5

**Introduction**

-> Définir notamment la démarche :

—

—

—

—

—

contraintes générales

+ type explo SimFeodal

-> choix méthodos

+ spécificités SimFeodal

-> choix méthodos/techniques

2

A écrire une fois que le chapitre aura été découpé.

Chapitre 5

5.1. CAPTER LES SORTIES DE SIMFEODAL

**5.1**

**Capter les sorties de SimFeodal**

Pour évaluer un modèle, on s’appuie sur plusieurs indicateurs de sortie de simu-

lation, de types divers (indicateurs numériques, graphiques, cartographiques

etc.,

rique). Quand le nombre d’indicateurs devient

important, comme c’est le cas dans le modèle SimFeodal (

teurs), la consultation des indicateurs pendant le déroulement

d’une simulation devient difficile. La complexité de ces indicateurs augmente

dans le cas d’un modèle stochastique, où il est nécessaire de multiplier les ré- plications afin d’avoir une idée fiable des tendances simulées par le modèle. Le travail de paramétrage d’un modèle requiert de plus de mener différentes

expériences, c’est-à-dire de faire varier les paramètres (

4) du modèle,

démultipliant encore la masse des sorties, et avec elle, la complexité de leur

analyse.

Nous détaillons ici les contraintes qu’entraînent ces différentes spécificités des

données issues de simulations. Ces contraintes sont transversales à plusieurs types de modèles, et on peut noter que ~~c ertains~~ autres types de modèles peuvent faire face à d’autres contraintes, ~~p ropres ou génériques.~~

Dans l’ensemble, les modèles peuvent être amenés à soulever les problèmes génériques à la production de données, quelles qu’en soient la source. Cette recherche n’a pas vocation à dresser le portrait de l’ensemble des contraintes et solutions relatives à l’enregistrement et au stockage de données. Cette partie vise à identifier les plus fortes limites qui rendent difficile l’enregistrement des données issues d’un modèle de simulation à base d’agents, descriptif, fortement stochastique, et exploratoire tel que SimFeodal.

**5.1.1**

**Masse des données**

**Facteur**

**multiplicatif**

×5000

[agents]

×20

[pas de temps]

×20

[réplications]

*agrégé du modèle ?*

×25

[expériences]

*meilleur ajustement*

×100

[explorations]

*paramètres ?*

Tableau 5.1 – Synthèse des questionnements liés à l’exploration de données

issues de modèle et de la multiplication des données et indicateurs selon la

hiérarchie des simulations.

3

***Lucigares du pharaon***

*2019-10-29 13:29:50*

--------------------------------------------

pour la mise en page finale : je mettrai les légendes en plus petit

***Lucigares du pharaon***

*2019-10-29 13:36:45*

--------------------------------------------

à mettre après "du modèle", là ça coupe ta proposition inutilement

***Lucigares du pharaon***

*2019-10-29 13:44:00*

--------------------------------------------

d'

***Lucigares du pharaon***

*2019-10-29 13:44:33*

--------------------------------------------

inutile

***Lucigares du pharaon***

*2019-10-29 14:20:33*

--------------------------------------------

En tant que spécialiste du "je tourne les choses de manière négative", je te propose la reformulation suivante (sans saut à la ligne):

"Cette partie vise (ainsi ?) à identifier les limites qui rendent difficile l'enregistrement des données issues d'un modèles de simulation à base d'agents, descriptif, fortement stochastique, et exploratoire tel que SimFeodal. C'est ce contexte spécifique de production des données et des contraintes qui lui sont associées que nous discutons ici. Les problèmes et les solutions rencontrés dans la production des données par des modèles s'inscrivent pour autant dans le cadre plus générique de l'enregistrement et du stockage des données, qu'elle qu'en soit la source. " Et pas la peine de dire que tu fais pas un "portrait exhaustif", personne ne s'attend à ce que tu le fasses ! Mais comme ça c'est + clair que tu te focus sur tes pbms et tes solutions à toi.

***Lucigares du pharaon***

*2019-10-29 13:59:19*

--------------------------------------------

contraintes sur quoi... ? la visualisation, l'analyse, l'enregistrement ?

Ta partie, ainsi que ton paragraphe du dessous, disent bien "enregistrement" / "captation" des données, mais au dessus tu parle de consultation (là ça va encore), puis d'analyse... Du coup on reste sur l'analyse dans notre tête. bref > ici reprécise la contrainte sur l'enregistrement des données

par exemple "Ces différentes spécificités issues de simulation entraînent différentes contraintes, d'abord sur leur enregistrement, que nous détaillons ici."

**Intitulé**

**Données**

**Indicateurs**

***Type de question***

**Quantité**

**Poids**

**Type**

**Quantité**

Un agent

1 ligne

≈ 200

octets

Visualisation en direct

Attributs de l’agent

*Le comportement individuel des agents semble-t-il plausible ?*

Un pas de temps

5 000

lignes

≈ 1 Mo

Visualisations en direct

3-4 indicateurs directs

*Le déroulé d’une simulation*

*semble-t-il correct ?*

Une simulation

100 000

lignes

≈ 20 Mo

Visualisations en direct

≈ 10 indicateurs

Une expérience

2 millions de lignes

≈ 400 Mo

Indicateurs de sortie

≈ 30 indicateurs (variabilité des réplications)

*Quel est le comportement*

Un modèle paramétré ou calibré

100 millions de lignes

≈ 10 Go

Indicateurs de sortie

≈ 3000

indicateurs (à comparer entre les expériences)

*Peut-on obtenir un du modèle ?*

Un modèle exploré/connu

10 milliards de lignes

≈ 1 To

Indicateurs numériques synthétiques

—

*Comment le modèle réagit-il aux variations de ses*

chap

tation des indica

chap 3, partie présen-

cf. chapitre 3, partie théo

Chapitre 5

5.1. CAPTER LES SORTIES

Dans un premier temps, il convient de noter que l’ensemble des indicateurs

observés en sortie de SimFeodal reposent sur des données qu’il est nécessaire de produire et d’enregistrer tout au long de la simulation. Ainsi, pour pouvoir tracer le graphique de l’évolution du nombre d’agrégats au cours du temps, il faut ~~a voir accès à cette information, et~~ dès lors, enregistrer, à chaque pas de temps, cette valeur dans un fichier numérique adapté. Cette information, en tant que telle, est assez faible, aussi bien en valeur sémantique qu’en valeur prise en mémoire. La masse représentée par cette information est toutefois démultipliée par la quantité d’indicateurs de sortie : plus ces derniers sont nombreux, plus la quantité de valeurs à stocker augmente. À chaque pas de temps, il faudra enregistrer les valeurs de plusieurs variables (voir tableau 5.1). Cette pratique est habituelle, et un format de données tabulaire se prête bien à un tel enregistrement : une ligne pour chaque pas de temps, et une colonne pour chaque variable à enregistrer. On obtiendrait ainsi en sortie de simulation un tableau contenant 20 1 lignes et une cinquantaine de colonnes 2, ce qui serait assez raisonnable pour une unique simulation.

**Évolution des indicateurs.** Dans le cadre d’un modèle exploratoire (

1), cette solution doit être écartée car elle ne permet pas de faire face

à d’éventuelles évolutions des indicateurs au fur et à mesure du cycle de vie

du modèle. Au fur et à mesure que le calibrage d’un modèle progresse, par exemple, on peut chercher à raffiner les indicateurs utilisés. Les indicateurs utilisés au départ peuvent alors ne plus montrer assez de variabilité entre des versions très proches de modèles. En effet, quand l’ajustement des données produites par la simulation aux objectifs empiriques s’améliore, il peut être difficile de départager plusieurs versions concurrentes d’un modèle. On peut alors ~~c hercher~~ à créer de nouveaux indicateurs, qui donneront une information plus détaillée sur l’un des résultats de simulation.

Cet exemple illustre le cas des ajouts d’indicateurs de sorties, mais on peut aussi

être confronté à des modifications des indicateurs existants. Dans SimFeodal, par exemple, le nombre de paroissiens moyen à chaque pas de temps peut être un indicateur utile au départ, mais ~~o n peut être amené à faire évoluer cet~~

~~indicateur en une étude de~~ la médiane si les nouvelles étapes de paramétrage du modèle en augmentent la variabilité.

Les indicateurs peuvent donc évoluer au cours du temps de vie du modèle

(cf.

3), ~~o u plus simplement, on peut être amené à réaliser une~~

~~observation plus fine des sorties du modèle au fur et à mesure de l’analyse~~

~~de ce dernier.~~ On se trouverait alors dans une situation impossible requérant de ré-exécuter les simulations après avoir adapté ou mis en place l’indicateur

1. Il s’agit ici du nombre de pas de temps de SimFeodal. On notera que ce nombre est parti- culièrement faible au regard de très nombreux modèles de simulation, en particulier vis-à-vis de ceux qui visent à provoquer l’émergence d’un phénomène. Ces modèles sont en général théoriques, et n’ont qu’une faible correspondance entre pas de temps et durée réelle du phé- nomène modélisé. Dans le cas de SimFeodal, où le temps est un élément crucial du modèle, la

résolution temporelle du modèle ne peut être diminuée artificiellement (voir

2.2.2.2), et l’on se satisfait donc de ce nombre d’itérations relativement faible.

2. Ce qui correspondrait par exemple à environ une colonne par indicateur, en plus des quelques colonnes de bases relatives à l’état d’ensemble de la simulation.

4

***Lucigares du pharaon***

*2019-10-29 14:18:21*

--------------------------------------------

mettre le tableau en dessous ?

***Lucigares du pharaon***

*2019-10-29 14:28:10*

--------------------------------------------

pour moi pas utile, à moins que tu discute spécifiquement "l'accès". En fait si tu veux garder ça je dirait plutôt "créer dans le modèle cette information" puis "enregistrer la valeur correspondante".

***Lucigares du pharaon***

*2019-10-29 15:10:00*

--------------------------------------------

je pense que tu devrais garder "indicateur", au moins pour cette phrase, peut-être pour les suivantes car dans ton tableau tu n'utilise pas "variable".

***Lucigares du pharaon***

*2019-10-29 15:08:04*

--------------------------------------------

pour moi pas très clair pourquoi c'est une solution "impossible"

***Lucigares du pharaon***

*2019-10-29 15:59:39*

--------------------------------------------

être amené

***Lucigares du pharaon***

*2019-10-29 16:13:29*

--------------------------------------------

on peut finalement lui préférer la médiane

***Lucigares du pharaon***

*2019-10-29 16:04:07*

--------------------------------------------

je trouve ça pas utile, tu viens de donner les 2 possibilités de changements d'indicateurs, qui sont globalement résumés par "les indicateurs peuvent donc évoluer etc.".

***Lucigares du pharaon***

*2019-10-29 16:53:55*

--------------------------------------------

ne pas mettre "éventuelle" puisque tu dis après que c'est très fréquent... en fait c'est le cas et c'est tout. Cela dit dans la suite je propose de supprimer la phrase où tu le dit + je pense que ce n'est pas la peine de dire dès la première phrase que ce n'est pas une solution adaptée, mais de l'introduire dans la conclusion de ce sous-sous paragraphe, ce que je te propose dans le commentaire tout en haut de la page 5 (à toi de voir pour ça) proposition pour ici, en fonction de ça:

"Dans le cadre d'un modèle exploratoire, il est très fréquent de faire évoluer les indicateurs au cours du cycle de vie du modèle. "

chap2, section

encadré chap

chapitre

cf.

Chapitre 5

5.1. CAPTER LES SORTIES

voulu.

**Données agrégées et individuelles.** Un ~~d ernier~~ point invalide cette so-

lution d’enregistrement : une part importante des indicateurs s’appuie sur des données individuelles et non agrégées. Ainsi, on peut, à chaque pas de temps, enregistrer le nombre de paroisses, leur superficie moyenne ou encore le nombre moyen de paroissiens que chacune dessert. Mais cela ne permet en aucun cas d’en dresser une cartographie, c’est-à-dire de réaliser une carte de la localisa- tion et des aires d’attraction des paroisses. Cela demanderait, par définition, d’enregistrer la géométrie de chaque paroisse à chaque pas de temps, les confi- gurations spatiales (localisation de chacune et donc distribution spatiale de l’ensemble) variant à chaque simulation.

Pour faire face à cette situation, on a donc fait le choix, dans SimFeodal, d’en-

registrer les états des variables à des niveaux d’agrégation multiples, y compris au niveau de l’agent, à chaque pas de simulation. Dans le cas des paroisses, le volume de données résultant reste contenu : on obtient un tableau d’environ 2000 lignes 3 et une dizaine de colonnes 4. L’enregistrement systématique de l’état de chaque agent à chaque pas de temps est toutefois bien plus gênant dans le cas d’autres agents, par exemple les foyers paysans. Il est en effet nécessaire d’enregistrer les attributs de chacun d’entre eux pour être en mesure d’étudier les liens entre les valeurs de satisfactions et les choix de déplacement, ou en- core d’observer la composition précise de la distribution des satisfactions. Avec 4000 foyers paysans au minimum à chaque pas de temps, les données changent d’ordre de grandeur 5 : chaque simulation requiert de générer un fichier conte- nant des dizaines de milliers de lignes, pour un total, pour cet unique fichier, d’une dizaine de mégaoctets occupés.

Au final, l’enregistrement d’un état représentatif d’une simulation est difficile.

Cela requiert de disposer de suffisamment d’éléments numériques pour pouvoir générer les indicateurs de sortie et rendre compte d’une partie de leur évolu- tion. La masse de données produite est de ce fait nécessairement conséquente, comme indiquée dans la ligne « simulation » du tableau 5.1.

**Coût temporel des simulations.** Un dernier aspect de l’exploration de

données issues de simulations, primordial, vient finir invalider l’approche d’en- registrement direct des indicateurs : le coût temporel. L’exécution d’une simu- lation requiert un certain temps de calcul (3 à 4 minutes pour une exécution

du modèle SimFeodal dans la version présentée dans le

2). Ce temps

de calcul ne peut être optimisé que dans des proportions faibles sans avoir à

bouleverser l’implémentation des mécanismes, ce qui représenterait un coût

temporel encore plus important (voir la succession d’étapes du

4). En

l’état actuel du modèle, la production des données a donc un coût temporel

3. Avec une moyenne de 150 paroisses, cela représente 20 [pas de temps] × 150 [paroisses] ≈ 3000

lignes pour chaque simulation.

4. Les identifiants de la simulation (nom, graine aléatoire), le pas de temps, l’identifiant de la paroisse, puis les différents attributs et enfin la géométrie stockée dans une colonne textuelle.

5. 20 [pas de temps] × 4000 [foyers paysans] ≈ 80 000 lignes pour une exécution du modèle.

5

***Lucigares du pharaon***

*2019-10-29 15:04:26*

--------------------------------------------

deuxième

***Lucigares du pharaon***

*2019-10-29 16:44:49*

--------------------------------------------

on s'emmêle un peu car tu ne précise pas assez : ce que le tableau pas de temps / indicateurs permet de faire VS ce dont toi tu as besoin. Ou plutôt tu parles de l'un puis de l'autre puis de l'un... bref. Proposition (pour tout le paragraphe):

"Un second point invalide en effet un enregistrement classique par indicateurs agrégés : à termes, les indicateurs que nous cherchons à obtenir renvoient à des données individuelles et non agrégées. Si les informations sur le nombre de paroisses, leur superficie moyenne, ou encore le nombre moyen de paroissiens que chacune dessert nous intéressent, elles ne sont pas suffisantes. Effectivement, nous cherchons également à dresser une cartographie des résultats du modèle, par exemple en représentant la localisation et les aires d'attraction des paroisses. L'enregistrement par indicateurs agrégé ne permet aucun cas d'obtenir de telles informations, qui nécessitent, par définition..."

***Lucigares du pharaon***

*2019-10-29 15:46:54*

--------------------------------------------

mettre 3000 par rapport à la note

***Lucigares du pharaon***

*2019-10-29 15:48:22*

--------------------------------------------

lourd ?

***Lucigares du pharaon***

*2019-10-29 15:49:45*

--------------------------------------------

"unique"

***Lucigares du pharaon***

*2019-10-29 15:49:41*

--------------------------------------------

"déjà contraignante?"

***Lucigares du pharaon***

*2019-10-29 15:50:38*

--------------------------------------------

"d'invalider"

***Lucigares du pharaon***

*2019-10-29 16:51:41*

--------------------------------------------

Proposition de reprise après avoir lu toute la sous-partie, et en particulier par rapport au paragraphe p.6 :

"Les indicateurs peuvent donc évoluer au cours du temps de vie du modèle (cf blabla), ce qui rend plus compliquée la mise en comparaison des différentes simulations. Avec une structure de données fixe et agrégée, l'introduction de nouveaux indicateurs ou la modification des indicateurs existants nécessiteraient alors de ré-exécuter systématiquement toutes les simulations précédentes pour obtenir les nouvelles valeurs souhaitées - une situation qu'il faut à tout prix éviter au risque d'accroître excessivement le nombre de simulations. Autrement dit, la production d'un tableau classique de données agrégés n'est pas adaptée au processus d'exploration du modèle, et cette solution doit être écartée. Pour répondre à cette contrainte, il est en définitive plus adapté d'enregistrer pour chaque simulation des indicateurs non pas agrégés mais à l'échelle individuelle des agents (foyers paysans et paroisses par exemple), et d'opérer seulement dans un deuxième temps des calculs pour obtenir des indicateurs agrégés et comparables. Cette solution permet en outre de répondre à d'autres objectifs et contraintes du modèles SimFeodal. "

***Lucigares du pharaon***

*2019-10-29 16:45:01*

--------------------------------------------

"ainsi que"

chapitre

chapitre

Chapitre 5

5.1. CAPTER LES SORTIES

élevé.

Qui plus est, ce coût est fortement dépendant du nombre de simulations exé-

cutées : si le modèle est exploratoire, donc sujet à de nombreux changements, et notamment à l’ajout ou à la modification des indicateurs observés, il peut donner lieu à d’encore plus nombreuses simulations. Avec une structure de données fixe et agrégée, on ne pourrait introduire de nouveaux indicateurs, et la mise en comparaison des simulations précédentes impliquerait leur adapta- tion et re-production, c’est-à-dire la ré-exécution systématique de l’ensemble des simulations précédentes à chaque changement dans les indicateurs. L’in- troduction de nouveaux indicateurs est pourtant très fréquente, en particulier dans le cas d’un modèle exploratoire où l’on affine au fur et à mesure de l’évo- lution du modèle ce que l’on peut y observer.

En tenant compte de ces éléments, on a tout intérêt à se prémunir de ré-

exécutions du modèle, et donc à enregistrer l’état de variables qui ne seraient pas encore mobilisées pour la production d’indicateurs. Dans l’exemple du nombre de paroissiens, il faudrait en enregistrer au minimum les moyennes, médianes, et sans doute quelques paramètres de dispersions en plus, voir les quantiles, afin d’adapter les indicateurs de sortie de la manière la plus adé- quate aux sorties des différentes versions du modèle. Dans le cas contraire, pour chaque changement ou ajout d’indicateur, il faudrait relancer des exécu- tions du modèle sur l’ensemble des jeux de paramètres précédents afin d’être en mesure d’avoir des indicateurs comparables entre les versions.

Enregistrer l’ensemble des variables d’un modèle est aisé dans le cas d’un

modèle théorique simple, par exemple dans le cas d’un modèle comme celui de Schelling (Schelling 1971). Cela se complique quand il s’agit d’enregistrer les variables d’un modèle plus complexe comme SimFeodal. Celui-ci requiert

en effet bien plus de variables globales (

etc.), pour représenter l’état du système dans son ensemble

à chaque instant. Surtout, SimFeodal est un modèle qui voit interagir plusieurs

sortes d’entités, chacune relatives à différents niveaux de granularité spatiale et sociale. Afin d’avoir tous les éléments en main une fois la simulation achevée, il est donc nécessaire d’enregistrer l’ensemble des variables non seulement glo- bales, mais aussi afférentes à chacun des types d’agents. D’un unique tableau de données exhaustif en sortie ~~d u modèle de Schelling,~~ on passe donc à plusieurs tableaux, dont les variables respectives seront propres à chaque type d’agent.

De plus, pour que les résultats de simulations soient comparables tout au long

du développement du modèle, en dépit de l’évolution des indicateurs qui les constituent, il est nécessaire d’enregistrer l’état individuel des agents, et pas seulement des données agrégées. Cela démultiplie la quantité d’informations qu’il est nécessaire de stocker (ligne « simulation » du tableau 5.1), qui plus est quand le nombre d’agents est important 6.

6. Dans les dernières versions de SimFeodal, dont les résultats sont présentés dans le

6, on est passé de 4 000 foyers paysans à 40 000. On peut donc décupler les valeurs des

colonnes « quantité » et « poids »…

6

***Lucigares du pharaon***

*2019-10-29 16:55:48*

--------------------------------------------

Ici très très clair, mais tu répètes beaucoup de chose déjà dites... - et ici de manière plus claire que parfois plus haut. Du coup les propositions que je t'ai faites de réécriture au dessus réinjectent des morceaux de ce paragraphe. Pour ici, proposition plus concise, du coup :

"Qui plus est, ce coût est fortement dépendant du nombre de simulations exécutées, qui est d'autant plus important dans le cas d'un modèle exploratoire. Comme on l'a vu, l'enregistrement direct d'indicateurs déjà agrégés impliquerait, lorsque ces derniers sont modifiés, que l'ensemble des simulations précédentes soient adaptées et ré-exécutées au fur et à mesure de l'exploration du modèle."

***Lucigares du pharaon***

*2019-10-29 15:57:43*

--------------------------------------------

hyper clair

Peut-être juste pour bien marquer que c'est une première limite > "En définitive, les objectifs d'analyse et d'interprétation que nous nous sommes fixés et le caractère exploratoire du modèle SimFeodal aboutit à la création d'une grande masse de données à enregistrer et traiter".

pitre

cha-

riables, paramètres

ref chapitre 4, dans distinctions va-

Chapitre 5

5.1. CAPTER LES SORTIES

**5.1.2**

**Tenir compte de la stochasticité du modèle - les répli-**

**cations**

**Un modèle stochastique.** Comme on l’a vu dans le

3, une simu-

lation ne suffit pas à évaluer le modèle. SimFeodal est ~~a insi~~ un modèle stochas-

tique, c’est-à-dire qu’une large partie des mécanismes qui l’animent sont basés sur des tirages aléatoires. Cet aléa est évident dans les mécanismes faisant ap- pel à un tirage aléatoire explicité, par exemple le choix de déplacement ou non

d’un foyer paysan (

ment). Dans le cas de ce mé-

canisme, un foyer paysan mobile se déplacera selon une probabilité dépendant

de sa satisfaction. Et s’il y a probabilité, il y a donc aléa. Même avec une forte satisfaction — 99% par exemple —, il reste 1% de chance qu’un foyer se dé- place, ce qui, sur un grand nombre de tirages (chaque foyer paysan, à chaque pas de temps), aboutit à une probabilité de réalisation non négligeable. ~~E t~~ cette probabilité de réalisation sera encore supérieure pour des foyers paysans ayant des niveaux de satisfaction légèrement moindre mais cependant globalement très élevés, supérieurs à 90% par exemple. En analysant les sortie du modèle, on aura donc la présence d’*outliers*, qu’il sera important d’isoler. Ils présenteront en effet des comportements contre-intuitifs puisque résultant d’une probabilité extrêmement faible. L’aléa a un poids important dans ce type de mécanisme.

~~Même dans le cas de mécanismes plus anodins,~~ l’aléa est fortement présent, puisqu’il est au cœur de la conception de SimFeodal. L’ordre d’exécution, c’est- à-dire l’ordre aléatoire dans lequel les agent sont appelés pour exécuter leurs mécanismes, aura donc un impact important sur les indicateurs de sortie de simulation, sans que cet impact ne puisse être caractérisé au moyen d’indi- cateurs agrégés. Par exemple, les seigneurs peuvent créer des châteaux, sous

condition de puissance (cf.

2.3.12). Pour créer ces châteaux,

il faut que des agrégats soient « disponibles », c’est-à-dire ne comportent pas

de château pré-existant à une certaine distance. Cette contrainte devient rapi- dement le facteur principal de la limitation de l’apparition de châteaux. Si un seigneur est plus souvent que les autres « appelé » en premier pour exécuter ce mécanisme, il pourra profiter des nouveaux agrégats disponibles pour créer ses châteaux. A force de création de châteaux, il sera relativement plus puissant, et pourra donc créer d’autant plus de châteaux relativement aux autres seigneurs. Il y aura donc une hiérarchie forte dans le nombre de châteaux possédés par seigneur.

Au contraire, si l’ordre d’appel des mécanismes favorise des seigneurs diffé- rents à chaque pas de temps, alors plus de seigneurs seront en mesure de créer des châteaux, et la hiérarchie sera alors plus faible.

Ces mécanismes sont sensibles à l’ordre d’appel, et il est ~~a insi~~ difficile de dis- cerner ce qui relève d’une tendance simulée et ce qui relève de fines variations dues à l’aléa, dans le comportement du modèle.

**Un modèle complexe.** On pourrait objecter qu’en considérant les agents

de manière agrégée, donc globale, les tirages de probabilité seront appliquées à suffisamment d’individus pour que le résultat statistique soit cohérent et ro- buste au niveau de la population dans son ensemble. En corollaire, le compor- 7

***Lucigares du pharaon***

*2019-10-31 14:42:38*

--------------------------------------------

"une seule/unique" ?

***Lucigares du pharaon***

*2019-10-31 14:47:58*

--------------------------------------------

"comme c'est le cas par exemple pour"

***Lucigares du pharaon***

*2019-10-31 14:49:11*

--------------------------------------------

"l'existence d'une probabilité implique, par définition, qu'il y a aléa"

Dans tous les cas, ne pas commencer la phrase par "et"

***Lucigares du pharaon***

*2019-10-31 15:06:43*

--------------------------------------------

pas de retour à la ligne

***Lucigares du pharaon***

*2019-10-31 15:07:07*

--------------------------------------------

donc

***Lucigares du pharaon***

*2019-10-31 15:08:41*

--------------------------------------------

Inverser :

"Les comportements issus de ces probabilités extrêmement faibles constitueront néanmoins des résultats contre-intuitifs dans le modèle : ce seront des outliers qu'il sera nécessaire d'identifier comme tels dans l'analyse des sorties du modèle."

***Lucigares du pharaon***

*2019-10-31 15:20:17*

--------------------------------------------

Changements un peu d'ordre des phrases :

"De manière plus globale/par ailleurs (si tu mets "tout d'abord" pour le premier cas), l'aléa est en fait au cœur de la conception du modèle SimFeodal, puisque l'ordre d'exécution, c'est-à-dire l'ordre dans lequel les agents sont appelés pour exécuter leurs mécanismes, est aléatoire. Or, l'ordre d'exécution a un impact important sur... etc."

***Lucigares du pharaon***

*2019-10-31 16:00:19*

--------------------------------------------

"Une grande partie / l'ensemble des mécanismes sont donc sensibles..." + pas de saut de paragraphe (là on dirait que c la conclusion de toute ta sous partie)

***Lucigares du pharaon***

*2019-10-31 15:19:47*

--------------------------------------------

"Tout d'abord"

chapitre 2, section

cf. chap2, mécanisme déplace

chapitre

Chapitre 5

5.1. CAPTER LES SORTIES

tement de chaque agent serait régulé par tant d’états aléatoires qu’on entre-

rait dans le cadre d’application de la loi forte des grands nombres, les agents adoptant alors en moyenne un comportement proche de l’espérance (moyenne théorique) de chaque tirage. Avec ces considérations, on pourrait justifier la robustesse probable des différentes exécutions de SimFeodal.

SimFeodal n’est toutefois pas simplement un modèle stochastique, mais avant

tout, un modèle complexe, c’est-à-dire s’inscrivant dans le champs des systèmes complexes. Sans ~~v ouloir ici~~ entrer dans les détails des implications et raisons de ceci, on peut simplement ~~e n~~ retenir qu’un modèle tel que SimFeodal est ex- trêmement sensible aussi bien aux conditions initiales qu’aux différents tirages

aléatoires (voir l’analyse de sensibilité,

6.3).

Pour illustrer, on peut s’appuyer sur un exemple, caricatural mais possible :

à l’initialisation, tous les foyers paysans, placés aléatoirement dans l’espace, seraient concentrés dans un espace d’étendue restreinte. Seul un énorme agré-

gat émergerait donc, et aucun pôle ne serait susceptible ~~d ès lors~~ de diviser ~~c et~~

~~agrégat géant.~~ On atteindrait ainsi une situation très éloignée des configura- tions spatiales observées empiriquement, et très éloignée aussi des réalisations habituelles du modèle. En présence d’un seul agrégat, les possibilités de dé- veloppement d’attracteurs (châteaux et paroisses) pourraient tout aussi bien être fortes que faibles. À partir d’une telle configuration initiale, on ne peut savoir si la situation convergerait vers un agrégat « paradisiaque », extrême- ment développé et doté de pôles satisfaisants, ou au contraire, vers un agrégat

« prison », où aucun des foyers paysans ne serait satisfait, mais n’aurait non plus d’alternative.

Cet exemple fictif, volontairement caricatural, ne s’est pas présenté jusqu’ici,

mais le cas échéant il faudrait pouvoir repérer ce type de comportement aber- rant. Cela serait par exemple utile pour éventuellement les distinguer des autres simulations et ne pas le laisser contrôler l’analyse d’un jeu de paramètres don- nées. De plus, cet exemple concerne uniquement une configuration initiale qui présenterait des caractéristiques tout à fait exceptionnelles. Plus générale- ment, il existe un grand nombre de situations initiales potentielles éloignées de l’empirique et même du vraisemblable. Les réalisations aberrantes peuvent apparaître à toute étape de la simulation, et déformer les tendances obser- vées dans les indicateurs de sortie du modèle. Au delà de l’initialisation, elles peuvent être issues de tirages aléatoires particulièrement défavorables, ou en- core apparaître suite à une succession d’événements improbables qui s’auto- renforceraient. Pour distinguer ces réalisations aberrantes de ce que l’on pour- rait caractériser d’une tendance normale, il est nécessaire de multiplier les ré- plications, afin de constituer un contexte suffisant pour isoler ces événements anormaux.

**Enregistrement des réplications.** On ne peut donc pas raisonner sur une

unique simulation pour évaluer un jeu de paramètres (

3). On ne peut

pas non plus se contenter d’une agrégation des résultats des différentes répli-

cations, sous la forme de moyennes ou d’écarts-types, selon qu’on s’intéresse par exemple à la tendance générale ou qu’on cherche à observer les variations 8

***Lucigares du pharaon***

*2019-10-31 15:23:33*

--------------------------------------------

ici

***Lucigares du pharaon***

*2019-10-31 15:23:53*

--------------------------------------------

A titre d'illustration

***Lucigares du pharaon***

*2019-10-31 15:24:17*

--------------------------------------------

restreint

***Lucigares du pharaon***

*2019-10-31 15:24:47*

--------------------------------------------

qu'

***Lucigares du pharaon***

*2019-10-31 15:25:54*

--------------------------------------------

dès lors

***Lucigares du pharaon***

*2019-10-31 15:29:22*

--------------------------------------------

Je rajouterai une petite phrase conclusive en mode "En d'autre termes, une situation initiale donnée peut aboutir à des situations très diverses et parfois complètement contraires."

***Lucigares du pharaon***

*2019-10-31 15:27:32*

--------------------------------------------

ref ici ou en ndbp

***Lucigares du pharaon***

*2019-10-31 15:29:41*

--------------------------------------------

s

***Lucigares du pharaon***

*2019-10-31 15:30:13*

--------------------------------------------

pas très clair, pourquoi est-ce que l'exemple "contrôlerai"... peut-être juste "influence excessivement ??"

***Lucigares du pharaon***

*2019-10-31 15:30:51*

--------------------------------------------

"En outre, des situations"

***Lucigares du pharaon***

*2019-10-31 15:32:39*

--------------------------------------------

"Du fait des contraintes liées à la stochasticité et à la complexité du modèle, ..."

cf. chap

chap 6, section

Chapitre 5

5.1. CAPTER LES SORTIES

que peut entraîner l’aléa.

Pour ces raisons, et pour être en mesure d’embrasser l’entière diversité des

sorties de simulations issues de variation de la graine aléatoire, il est donc nécessaire de mener plusieurs réplications de chaque simulation, et d’enre- gistrer l’entièreté des sorties de simulations dans chacun des cas. Le jeu de données produit par une simulation, contenant quelques dizaines de milliers de lignes, est ainsi obligatoirement multiplié par le nombre de réplications. Pour l’exploration de SimFeodal, après différents tests, ce nombre a été fixé à

parlé dans le chapitre 3 (évaluation),

soit

tain.). La vingtaine de mégaoctets

issue d’une simulation devient donc approximativement 400 mégaoctets, et le

nombre de lignes contenues, par exemple pour les foyers paysans, passe d’en- viron cent mille à deux millions 7 (voir la ligne « expérience » du tableau 5.1).

**5.1.3 Des réplications aux expériences**

Comme décrit dans le

4, le paramétrage de SimFeodal a demandé

plusieurs étapes 8. De plus, chacune de ces étapes représente plusieurs sous- étapes – les expériences – faites d’essais et d’erreurs, en faisant varier à chaque fois les valeurs de paramètres de SimFeodal. Afin de construire le modèle, puis de l’explorer de manière plus systématique, il a été nécessaire de tester des dizaines de configurations de paramètres. Pour comparer, à chaque nouvelle version du modèle, les résultats produits par rapport aux résultats de la version précédente, il est indispensable de conserver, au minimum, l’ensemble des jeux de données de cette version précédente.

Cet archivage des résultats immédiatement précédents n’est pourtant pas suf-

fisant, pour des raisons tenant à la reproductibilité et à traçabilité du modèle obtenu au final. On serait en effet tenté, à chaque nouvelle version « majeure » du modèle, de ne conserver que les indicateurs de sorties des versions précé- dentes en considérant que le modèle a atteint une phase de maturité supérieure à chaque fois. Les étapes intermédiaires, reléguées au rang de brouillons ou d’esquisses, deviendraient alors inutiles. Le processus de conception et de pa-

ramétrage d’un modèle n’est pourtant pas linéaire (cf.

4), et on peut

avoir besoin de comparer une version intermédiaire « actuelle » à une version

intermédiaire précédente, quitte à réaliser qu’une modification erronée a été ajoutée au modèle.

La conservation des résultats de chacune des expériences joue donc à nouveau

un rôle multiplicatif dans la masse de données à conserver (voir le tableau 5.1). En supposant que le modèle ait compté environ 25 versions 9, on obtient envi-

7. Si cette quantité de données semble tout à fait raisonnable et peut largement être traitée sur un ordinateur classique, on peut toutefois noter qu’elle dépasse déjà le maximum de lignes (limitées à soit un peu plus d’un million) que les tableurs classiques – LibreOffice ou Microsoft Excel dans leurs dernières versions – sont en capacité de gérer.

220,

8.

9. Dans SimFeodal, par exemple, on distingue 6 versions « majeures » (cf.

4, elles-mêmes composées de 4 à 10 sous-versions)

9

lisé.

***Lucigares du pharaon***

*2019-10-31 15:34:28*

--------------------------------------------

pour bien expliciter, peut-être ajouter "à l'initialisation du modèle" ? Car jusque là tu n'en parles pas de la graine aléatoire... peut-être aussi faire ref à là où tu la présente ?

***Lucigares du pharaon***

*2019-10-31 15:38:14*

--------------------------------------------

ajouter "(une "expérience")", mais si tu décris plus bas ce qu'est une expérience

***Lucigares du pharaon***

*2019-10-31 15:40:24*

--------------------------------------------

comme étapes" est très générique ici, ce n'est pas très clair la hiérarchie étapes/sous-étapes. Est-ce que le tableau que tu mentionnes en note sera ici et le précisera ? Sinon, je dirais plus simplement "Parmis ces étapes, on compte des "expériences", faites d'essais et d'erreurs, et qui font varier... etc" (voir commentaire juste après)

***Lucigares du pharaon***

*2019-10-31 15:41:47*

--------------------------------------------

à chaque fois de quoi ? entre chaque expérience j'imagine. Du coup "pour chacune" ou juste ne rien mettre et on comprend bien.

***Lucigares du pharaon***

*2019-10-31 15:43:48*

--------------------------------------------

proposition :

" il a été nécessaire de réaliser des dizaines d'expériences, autrement dit de tester etc." OU alors rajouter "autrement dit de réaliser des dizaines d'expérience" à la fin.

***Lucigares du pharaon***

*2019-10-31 15:44:45*

--------------------------------------------

"seraient dès lors considérées comme"

***Lucigares du pharaon***

*2019-10-31 15:46:54*

--------------------------------------------

pas clair le "intermédiaire actuelle / intermédiaire précédente". Plus simplement ? "On peut avoir besoin de comparer n'importe quelle version avec n'importe quelle autre version, par exemple pour identifier si une modification ajoutée au modèle plusieurs versions auparavant n'est pas erronée".

***Lucigares du pharaon***

*2019-10-31 15:58:22*

--------------------------------------------

"Il est donc nécessaire de conserver les résultats de chacune des expérience, ce qui joue un rôle multiplicatif..." + "voir la ligne "modèle paramétré ou calibré"" dans la parenthèse

***Lucigares du pharaon***

*2019-10-31 15:57:52*

--------------------------------------------

attention des fois du mets "cf", des fois "voir", et des fois rien : par exemple au dessus (ligne "simulation" du tableau)

tableau dans chap

à corriger quand tableau fina

Y mettre un tableau avec les versions + nombre de sous-version du modèle.

chapitre

chapitre

20 réplications (

J’en aurais

sans doute

mais à laisser ici jusqu’à ce

que ce

cer

Chapitre 5

5.1. CAPTER LES SORTIES

ron 500 simulations à stocker, puis à ~~d evoir~~ analyser et comparer. Cela repré-

sente une somme considérable de données (voir tableau 5.1), qui se chiffrent en dizaines de millions d’enregistrement 10.

En matière de stockage, il ne s’agit jamais que de quelques gigaoctets de don- nées, pourtant à la limite de ce que l’on peut traiter sur un ordinateur indivi- duel 11.

On ne mentionne ici que les expérimentations issues des étapes de paramé- trage. Les phases suivantes, visant à l’exploration du comportement du mo- dèle (analyse de sensibilité, calibration…), demandent ~~ainsi~~ d’exécuter, et donc d’enregistrer, une masse bien plus importante de simulations.

**5.1.4**

**Des données aux indicateurs**

Dans l’ensemble, l’enregistrement et la sauvegarde des données issues de si-

mulations constituent, pour les modèles de simulations basés sur de nombreux agents et mécanismes, une contrainte importante vis-à-vis de l’exploration du comportement de ces modèles.

~~C’est particulièrement le cas pour SimFeodal, où l’on ne peut se contenter de~~  produire à la volée les indicateurs pour des raisons de reproductibilité 12.

**Analyser une masse de données.** La masse de données en sortie est im-

pressionnante et requiert dès lors, d’un point de vue technique, d’utiliser des outils adaptés à la manipulation de grands jeux de données. Cela exclut de fait l’outillage habituel, accessible, de la géographie quantitative, ne laissant par exemple pas la possibilité d’utiliser les outils à interface graphique classiques. Au contraire, face à des données de cet ordre, seules des solutions statistiques, basées sur des analyses en ligne de commande, peuvent être mobilisées. Ces so- lutions doivent en plus être appuyées par des capacités de calculs importantes, sans toutefois justifier encore l’usage de technologies de calcul intensif 13. . Cela pose une contrainte dans l’accessibilité aux analyses : le traitement des données requiert des compétences spécifiques en analyse de données volumi- neuses. Dans un contexte interdisciplinaire caractérisé par une large hétérogé- néité en matière de pratiques quantitatives, il n’est pas possible de se contenter d’envoyer les jeux de données produits aux thématiciens – qui ne disposent le

20 × 5000 × 20 × 25

[pas de temps] [agents] [réplications] [jeux de paramètres]

≈ 100 000 000 de lignes enre-

10.

gistrées pour les agents (4 000 foyers paysans et environ 1 000 autres agents).

11. Selon une approximation courante, on ne peut charger en mémoire des données d’une taille supérieure à la moitié de la mémoire vive. Approximation qui approche du tiers quand on prend en compte les autres processus en cours, et éventuellement des modifications à l’échelle de l’ensemble du jeu de données plutôt que sur des extraits. Pour pouvoir traiter ces 5 Go de données (tableau 5.1), l’ordinateur utilisé doit donc disposer d’au moins 16 gigaoctets de mémoire vive, et encore, au prix d’un traitement potentiellement lent et bloquant.

12.

13. Le « *High-Performance Computing* » (HPC) par exemple, mobilisé pour l’étude de données plus massives, c’est-à-dire trop importantes pour être analysées sur un unique ordinateur ou serveur. Voir ReY-CoYrehourcq (2015, p. 223–283) pour un historique détaillé des usages HPC en SHS, et particulièrement en géographie.

10

***Lucigares du pharaon***

*2019-10-31 15:59:05*

--------------------------------------------

à

***Lucigares du pharaon***

*2019-10-31 15:59:55*

--------------------------------------------

pas de saut de paragraphe

***Lucigares du pharaon***

*2019-10-31 16:00:52*

--------------------------------------------

"qui sont"

***Lucigares du pharaon***

*2019-10-31 16:02:12*

--------------------------------------------

"Enfin, on a présenté ici que..."

***Lucigares du pharaon***

*2019-10-31 16:02:33*

--------------------------------------------

"encore plus"

***Lucigares du pharaon***

*2019-10-31 16:03:05*

--------------------------------------------

nombre ?

***Lucigares du pharaon***

*2019-10-31 16:06:03*

--------------------------------------------

"habituel et accessible"

***Lucigares du pharaon***

*2019-10-31 16:14:38*

--------------------------------------------

, et en particulier pour SimFeodal

La reproductibilité sera abordée dans le chapitre 1 (positionnement).

Chapitre 5

5.1. CAPTER LES SORTIES

plus souvent pas de ces compétences – : ils seraient alors en difficulté pour en

tirer les analyses nécessaire à leur interprétation.

**Analyser une masse d’indicateurs.** D’un point de vue thématique, et

c’est là l’objectif, cette masse de données doit servir à la production d’indi- cateurs, nombreux et divers aussi bien dans leur forme que dans les caracté-

ristiques des processus qu’ils décrivent (

teurs). Les mêmes

raisonnements, « multiplicatifs », que pour les données s’appliquent ainsi aux

indicateurs. On peut rendre compte de la variabilité des réplications direc- tement dans les indicateurs produits (par exemple avec des représentations graphiques de type *box-plot*, utilisés ici pour une large partie des indicateurs).

La production de tels indicateurs au niveau de la variabilité inter-expérience

est pourtant difficile, si tant est qu’elle soit souhaitable. De fait, chaque expé- rience doit pouvoir être comparée aux précédentes sur la base de leurs seules réplications respectives. Dès lors, la raison d’être des indicateurs de sortie est de rendre possible une comparaison, indicateur par indicateur, entre chacune des expériences. Il est donc indispensable de générer, pour chaque expérience, l’ensemble des indicateurs. En ne considérant ici encore que 25 expériences, cela fait donc déjà plusieurs centaines 14 d’indicateurs (tableau 5.1).

Le choix ayant été fait de mener une comparaison visuelle (

tifs), ~~on imagine dès lors que~~ celle-ci

va être difficile en présence de tant d’indicateurs.

En sus de la contrainte de l’enregistrement et de la production des indicateurs,

le verrou majeur à l’exploration du comportement ~~a ttendu~~ de SimFeodal est donc la simple capacité à visualiser et à explorer l’ensemble des indicateurs de sortie. Ce qui doit de plus être rendu accessible y compris pour un auditoire

non habitué à la manipulation de nombreuses données et sorties quantitatives.

14. En considérant ainsi une trentaine d’indicateurs, on obtient donc 30 [indicateurs] ×

25

≈ 750 indicateurs uniques.

[jeux de paramètres]

11

***Lucigares du pharaon***

*2019-10-31 16:10:47*

--------------------------------------------

donc mener à

***Lucigares du pharaon***

*2019-10-31 16:13:08*

--------------------------------------------

plus facilement utilisables et interprétables par les thématiciens, mais qui doivent être

***Lucigares du pharaon***

*2019-10-31 16:16:44*

--------------------------------------------

"qu'on a choisi d'utiliser" ?

***Lucigares du pharaon***

*2019-10-31 16:29:17*

--------------------------------------------

j'ai eu du mal à comprendre ce passage parce que tu parles des replications, des expériences, des indicateurs tous ensembles... j'ai eu du mal en particulier sur la fin. Du coup proposition pour mieux cadrer le début, et pour la fin à toi de voir mais il faut clarifier : "Pour une expérience donnée, on peut rendre compte de la variabilité des réplications directement à partir des indicateurs, par exemple en en représentant les valeurs sous forme de box-plot (une représentation graphique que nous avons de fait largement utilisée). Pour comparer les différentes expériences, c'est-à dire analyser la variabilité inter-expérience, il n'est pas souhaitable de créer des indicateurs globaux..."

***Lucigares du pharaon***

*2019-10-31 16:31:05*

--------------------------------------------

sera dès lors

***Lucigares du pharaon***

*2019-10-31 16:33:20*

--------------------------------------------

du fait de ce très grand nombre de valeurs

***Lucigares du pharaon***

*2019-10-31 16:32:02*

--------------------------------------------

Cette visualisation doit pourtant

***Lucigares du pharaon***

*2019-10-31 16:32:24*

--------------------------------------------

pas de saut de paragraphe

***Lucigares du pharaon***

*2019-10-31 16:33:04*

--------------------------------------------

valeurs

***Lucigares du pharaon***

*2019-10-31 16:34:00*

--------------------------------------------

de valeurs en fait > si l'indicateur c'est "la moyenne", ce sont bien les valeurs des moyennes qui sont nombreuses

3 : indicateurs uniques vs fonctions objec

ref. dans chapitre

ref. chap. 3, indica

Chapitre 5

5.2. COMMENT EXPLORER LES SORTIES DE SIMFEODAL ?

**5.2**

**Comment explorer les sorties de SimFeodal ?**

Pour évaluer, de manière approfondie, une expérience (voir tableau 5.1) d’un

modèle tel que SimFeodal, il est nécessaire de passer en revue de nombreux indicateurs de sortie de simulation. Cette évaluation ne vise pas à produire une

« note » unique et synthétique, mais plutôt à tester la capacité de l’expérience à reproduire les dynamiques que le modèle chercher à reproduire. Il ne s’agit pas, à proprement parler, d’une validation du modèle, au sens quantitatif où on pourrait l’entendre.

On vise plutôt à explorer le comportement du modèle en fonction des méca-

nismes et valeurs de paramètres choisis. Cela aboutit donc sur un jugement qualitatif sur la capacité du modèle à reproduire les dynamiques souhaitées. Pour mener cette exploration, il convient d’utiliser des outils adaptés, c’est-à- dire de disposer de solutions techniques permettant le calcul et l’affichage des indicateurs à partir des données produites par le modèle.

Dans le travail mené autour de SimFeodal, plusieurs solutions ont été utilisées au cours des différentes étapes de construction du modèle. La restitution pu- rement chronologique de ces solutions ne revêt pas d’intérêt propre, mais les contraintes accumulées au cours de la construction du modèle ainsi que les choix devant permettre de les dépasser nous paraissent très largement géné- riques et généralisables.

La succession de choix d’outils d’explorations se justifie par les verrous dans

l’exploration que chacun de ces outils a permis de débloquer. Cela dresse par là-même un portrait des solutions méthodologiques d’exploration de données de simulations dont on peut faire usage selon les contraintes générales des modèles.

**5.2.1**

**Observer les simulations en direct ou *a posteriori***

Classiquement, le premier réflexe d’un modélisateur, du moins pour les mo-

dèles à base d’agents, est de définir des sorties graphiques pour accompa- gner son modèle. Les différentes plate-formes de modélisation agent mettent d’ailleurs régulièrement en avant les possibilités de représentations qu’offrent leurs environnements 15. Visualiser le déroulement d’un modèle « en direct » (« *online* » dans Grignard et Drogoul 2017), c’est-à-dire au sein de la plate- forme de simulation et au cours l’exécution du modèle, offre ainsi de nombreux avantages : évaluation visuelle du niveau de ségrégation (et de son évolution) dans une implémentation du modèle de Schelling ; visualisation de cohérence du déplacement des nuées d’oiseaux dans un modèle de type « Flocks » (ReY- nolds 1987) ; ou encore suivi d’un indicateur dans le temps – la quantité de ressources collectées – dans un modèle de type « Sugarscape » (Epstein et Ax- tell 1996).

15. Voir par exemple les collections de visualisations sur les pages d’accueil de Gama (https ://gama-platform.github.io/), de NetLogo (https ://ccl.northwestern.edu/netlogo/), de GeoMASON (https ://cs.gmu.edu/∼eclab/projects/mason/extensions/geomason/) ou encore de Repast (https ://repast.github.io/screenshots.html).

12

***Lucigares du pharaon***

*2019-10-31 16:39:57*

--------------------------------------------

pour chaque sortie

***Lucigares du pharaon***

*2019-10-31 16:40:15*

--------------------------------------------

en effet

***Lucigares du pharaon***

*2019-10-31 16:40:30*

--------------------------------------------

pas de saut de paragraphe

***Lucigares du pharaon***

*2019-10-31 16:40:43*

--------------------------------------------

des

***Lucigares du pharaon***

*2019-10-31 16:42:26*

--------------------------------------------

"choisis, de manière à construire un jugement/avis qualitatif..."

Chapitre 5

5.2. EXPLORER LES SORTIES

**Visualisation des simulations en direct.** Il est à noter que dans le cadre

du développement d’un modèle de simulation, l’implémentation du modèle et de son interface graphique sont étroitement liées. D’une part, la plateforme de modélisation choisie contraint fortement le type et la diversité des représen- tations. Gama et GeoMASON, par exemple, proposent des modes de visuali- sation de données géographiques bien plus avancés que NetLogo ou Repast. L’interface graphique développée pour chaque modèle est donc largement in- fluencée par la plateforme de simulation dans laquelle il est exprimé. D’autre part, dans la plupart de ces plateformes de simulation, l’interface graphique est implémentée au même niveau que le code-source du modèle en lui-même. Dans Netlogo et Gama par exemple, l’interface graphique est programmée di- rectement dans le modèle en lui-même, en se basant sur les variables qui y sont déclarées. Il n’est donc pas possible de créer une interface graphique générique au sein des plateformes de simulation, laquelle pourrait s’appliquer à plusieurs modèles différents. Il est nécessaire, pour chaque modèle, de reconstruire l’in- terface depuis les briques de bases proposées par les plateformes, c’est-à-dire un ensemble de primitives graphiques permettant de composer une interface intégrée au modèle.

Dans l’exploration de SimFeodal, la création en direct de quelques graphiques

correspondant à des indicateurs étudiés permet d’assurer un rôle de filtrage, grossier, à l’exécution d’une expérience, c’est-à-dire de l’ensemble des réplica- tions nécessaires à son évaluation. Après une modification du code-source du modèle, et avant de lancer de nombreuses simulations, on exécute quelques simulations « manuellement ». On s’assure alors, en direct, que les indicateurs affichés ne présentent pas de caractères aberrants. Cela permet de vérifier, avant de lancer des calculs plus conséquents, que le déroulement de la simu- lation semble cohérent, c’est-à-dire, le plus souvent pour des modifications mineurs du modèle, qu’un *bug* n’a pas été introduit involontairement.

Le recours à ce type de visualisation en direct des simulations ne peut toutefois

être généralisé, c’est-à-dire sorti de son rôle de pré-filtre, en raison de deux principales contraintes.

**Visualisation en direct et réplication.** La première contrainte, déjà évo-

quée plus haut, est que le modèle SimFeodal est fortement stochastique. Dès lors, la visualisation des indicateurs d’une simulation particulière n’est pas suf- fisante pour estimer le comportement du modèle. En conséquence, les indica- teurs choisis pour l’évaluation de SimFeodal prennent presque tous en compte la variabilité des résultats induite par l’exécution de réplications. Certaines plateforme de simulation multi-agents 16 permettent toutefois de mener conco- mitamment plusieurs réplications d’un même modèle et de visualiser directe- ment pendant l’exécution les résultats agrégés des réplications. Cette première contrainte, liée au besoin d’analyser des réplications plutôt que des exécutions isolées, peut donc être dépassée en adaptant l’implémentation du modèle pour faire usage de ces capacités de multi-simulation.

16. Gama dans sa dernière version (1.8) par exemple, voir

https ://gama-platform.github.io/wiki/RunSeveralSimulations.html.

13

***lue***

*2019-11-11 16:19:01*

--------------------------------------------

"en direct au sein de la plateforme Gama"

***lue***

*2019-11-11 16:19:29*

--------------------------------------------

inutile

***lue***

*2019-11-11 16:20:49*

--------------------------------------------

"utilisé pour des analyses plus poussées que ce rôle de ce pré-filtrage"

***lue***

*2019-11-11 19:51:17*

--------------------------------------------

tu en donne plus que deux... même s'il y en a deux "principales" en fait tu en discutes de 4... sinon mets juste "plusieurs raisons"

Chapitre 5

5.2. EXPLORER LES SORTIES

**Visualisation multi-indicateurs.** La seconde contrainte est cruciale dans

le cas de SimFeodal et invalide l’usage des méthodes de visualisation en di- rect. On l’a vu, l’exploration des sorties de simulation du modèle repose sur la consultation systématique de plusieurs indicateurs, dont le nombre peut se révéler important pour une analyse approfondie.

Tout d’abord, il est concrètement difficile de représenter tous ces indicateurs au

sein de l’interface graphique d’une plate-forme de simulation agent, comme on peut le remarquer, par exemple en figure 5.1 : l’espace occupé par les quelques indicateurs temporels et numériques affichés est déjà important et rend l’in- terface d’ensemble complexe. De plus, et c’est sans doute le verrou majeur, la temporalité de l’exécution d’une simulation – ou même des réplications néces- saires – est bien plus courte que celle requise pour la compréhension des ré- sultats produits. Une simulation requiert ainsi au maximum quelques minutes. Pour pouvoir examiner tous les indicateurs pendant cette durée, il serait néces- saire de mettre la simulation en pause régulièrement, presque à chaque pas de temps. On disposerait alors d’un temps suffisant pour observer les indicateurs (organisés en onglets dans l’interface visible dans la figure 5.1). L’analyse des indicateurs de sortie de simulation demandent en effet un examen approfondi, et non simplement superficiel, pour pouvoir juger de l’adéquation de ce que ces indicateurs représentent vis-à-vis des attentes thématiques.

**Visualisation différée.** Cette contrainte est renforcée par la pratique

d’évaluation que la plupart des chercheurs mobilise. L’évaluation n’est pas une étape unique et finie, il est utile de pouvoir revenir sur les résultats à différents moments. Cela est par exemple nécessaire quand il s’agit de comparer de nou- veaux résultats produits à ceux générés par des expérimentations antérieures. On ne peut alors se contenter d’évaluations en direct, même en y consacrant un temps important, simplement parce que par nature, ces évaluations seront à reproduire en plusieurs occasions, et qu’il ne serait alors pas rationnel de relan- cer, à chaque fois, de nouvelles simulations correspondant à des configurations de paramètres et de mécanismes déjà éprouvées.

**Visualisations multiples.** Un dernier élément contribue à la difficulté

de se baser sur une évaluation en direct : en plus du chercheur, amené à reve- nir de multiples fois sur les résultats d’une expérience, un modèle co-construit peut être évalué par plusieurs chercheurs différents. C’est d’autant plus fré- quent en situation d’interdisciplinarité, où les points de vue de chacun des membres sont complémentaires et nécessaires. Sauf à faire preuve d’une dis- cipline exacerbée, par exemple en réalisant l’ensemble du travail d’évaluation uniquement en séances de travail simultanées et collectives, l’évaluation par plusieurs personne demande que chacun puisse mener ces analyses selon ses propres temporalités. En choisissant de baser l’évaluation d’un modèle uni- quement sur une analyse en direct, il faudrait alors que chaque chercheur, à chaque fois qu’il souhaite évaluer une même expérience, ré-exécute le modèle de nombreuses fois. Cela serait naturellement possible, mais constituerait clai- rement un gâchis certain de ressources.

L’évaluation d’un modèle interdisciplinaire et exploratoire ne peut donc que 14

***lue***

*2019-11-11 19:45:08*

--------------------------------------------

inutile ici mais info importante, je la remonterai au moment où tu appelle la figure, pour la décrire.

"par exemple en figure 5.1. Sur celle-ci, l'espace au centre est occupé dans les deux cas par trois indicateurs temporels et numériques (l'ensemble des indicateurs sont rangés en onglets à droite), qui occupent une place déjà importante et rendent l'interface d'ensemble complexe."

***lue***

*2019-11-11 19:43:12*

--------------------------------------------

pas hyper clair comme dénomination. Un indicateur temporel n'est pas numérique ?

***lue***

*2019-11-11 19:45:35*

--------------------------------------------

en lien avec le com d'en dessous "l'ensemble des indicateurs disponibles"

***lue***

*2019-11-11 19:50:32*

--------------------------------------------

je mettrais plutôt ": il est fréquent de devoir revenir..."

***lue***

*2019-11-11 19:50:08*

--------------------------------------------

et plus spécifiquement c'est le cas pour toi... c'est bizarre du coup que tu en parles de manière générale. Pour le paragraphe, proposition :

"Cette contrainte est renforcée par notre pratique d'évaluation non-linéaire, qui est plus généralement celle utilisée par la plupart des chercheurs."

***lue***

*2019-11-11 19:58:25*

--------------------------------------------

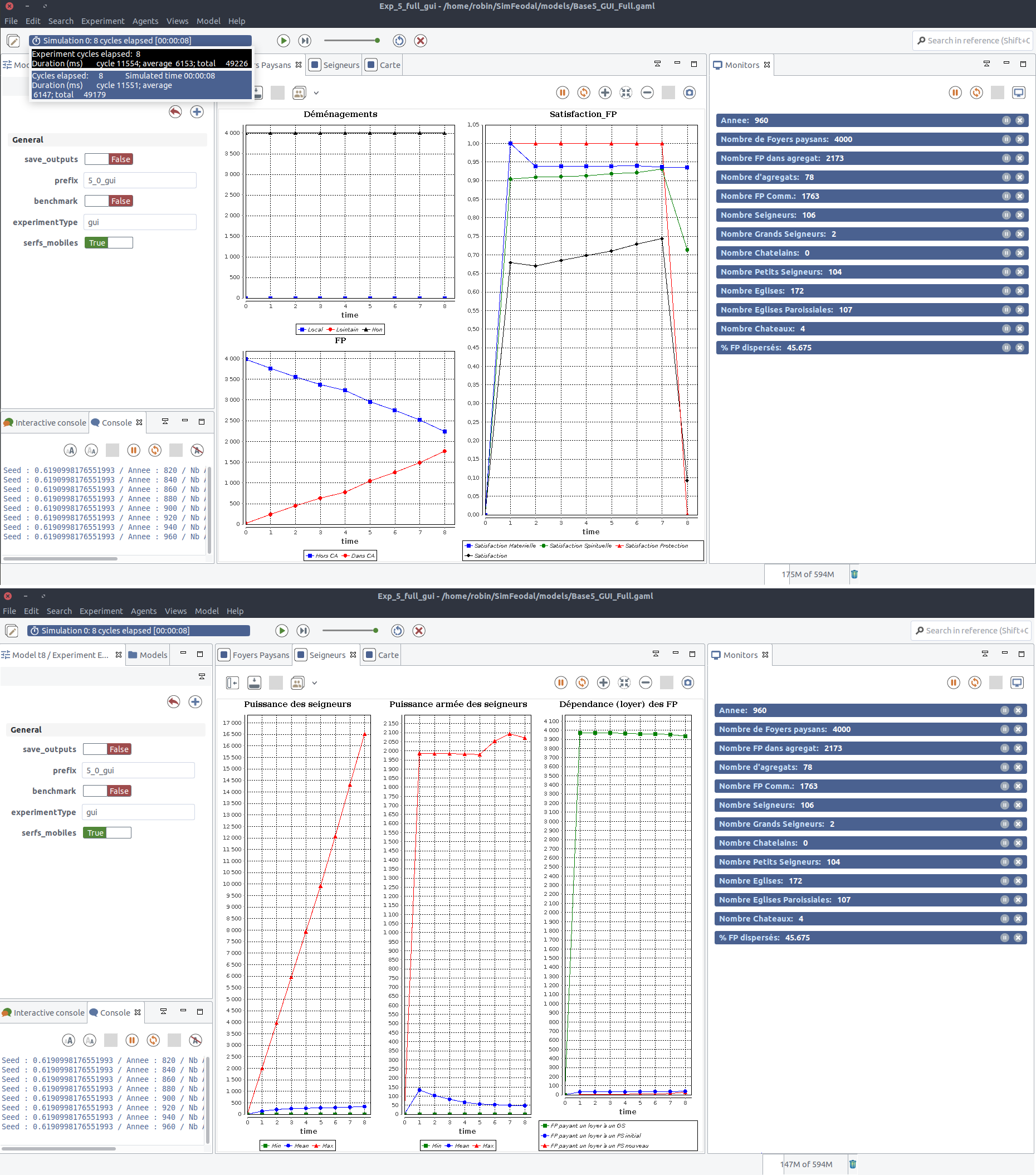
chercheur-modélisateur ?

***lue***

*2019-11-11 20:01:30*

--------------------------------------------

"nécessiterai d'y consacrer de trop nombreuses heures et ressources " (j'imagine que tu parles de ressources de calcul et d'exécution du modèle ?)



Chapitre 5

5.2. EXPLORER LES SORTIES

difficilement être réalisé en direct, qui plus est quand elle demande de faire

appel, dans un cadre de co-construction, à plusieurs points de vue hétérogènes. Les modalités mêmes de l’exploration des sorties d’un modèle à évaluation visuelle requièrent donc que les indicateurs soient accessibles et explorables à des temporalités différentes, par des chercheurs différents, depuis des lieux divers.

Il est donc indispensable que les indicateurs soient enregistrés et consultables

simplement à tout moment, *a posteriori* des simulations, ce qui élimine de fait la visualisation des indicateurs en direct pendant l’exécution des simulations comme unique méthode d’exploration du comportement des modèles explora- toires et descriptifs.

Figure 5.1 – Indicateurs intégrés à l’interface graphique interne de SimFeodal.

Dans ces captures d’écrans, il s’agit d’indicateurs liés aux foyers paysans et aux seigneurs.

**Visualiser en direct pour pré-filtrer.** Dans le cadre de la construction et du paramétrage de SimFeodal, nous avons cependant mobilisé ce type de visualisation en direct, comme on peut le constater dans l’interface graphique

15

***lue***

*2019-11-11 20:00:20*

--------------------------------------------

commentaire pour les figures : elles sont trop petites je pense, faut au moins pouvoir lire les titres des graphiques pour qu'on ait une idée de ce que c'est... à mettre chacun sur une pleine page ? ou n'en mettre qu'un seul en pleine page ?

***lue***

*2019-11-11 20:01:57*

--------------------------------------------

tel

***lue***

*2019-11-11 20:06:39*

--------------------------------------------

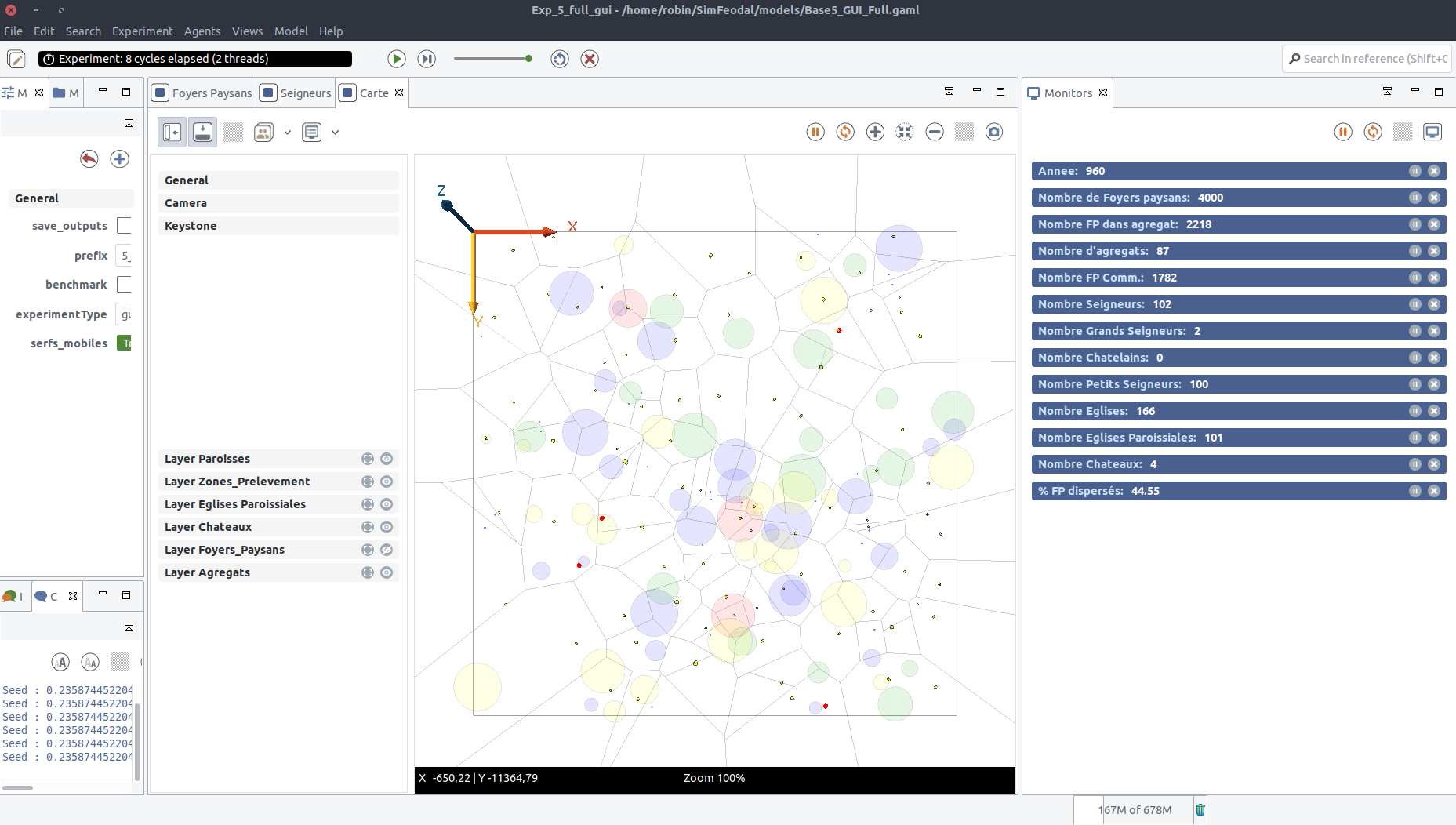
répétition de "différent" marche aussi

***lue***

*2019-11-11 20:17:03*

--------------------------------------------

tout ce paragraphe est hyper répétitif par rapport au premier... tu peux redire juste en une phrase "On garde donc la visu direct uniquement pour un usage de pré-filtrage comme décrit ci-dessus"



Chapitre 5

5.2. EXPLORER LES SORTIES

du modèle (figures 5.1 et 5.2). La visualisation en direct n’est donc pas mo-

bilisable en tant que méthode d’évaluation principale, mais elle peut tout de même, comme dans un usage très classique, être utilisée comme un outil de validation interne pour tester chaque modification dans les valeurs de para- mètres, remplissant alors le rôle de « préfiltre » décrit auparavant. Visualiser une seule simulation, avant d’en exécuter les réplications nécessaires, permet ainsi déjà de vérifier que les modifications apportées dans les valeurs de pa- ramètre ou dans les mécanismes n’ont pas entraîné l’apparition de *bugs* ou d’incohérences immédiatement visibles.

Nous avons donc choisi de développer une interface graphique, très sommaire

mais permettant des allers-retours rapides entre l’implémentation et l’exécu- tion, au sein de l’implémentation de SimFeodal. Cette interface n’affiche qu’un nombre réduit d’indicateurs (figure 5.1), ainsi qu’une représentation cartogra- phique (figure 5.2) utile à une analyse rapide du comportement d’ensemble du modèle.

Figure 5.2 – Visualisation intégrée à l’interface graphique interne de SimFeo-

dal : cartographie synthétique de l’espace modélisé.

~~Ces indicateurs intégrés à l’interface graphique interne de SimFeodal permettent~~

~~de mener une étape préalable à l’évaluation du modèle. Cette étape vise à ef-~~

~~fectuer un premier filtrage des simulations avant d’exécuter les expériences~~

~~en elles-mêmes.~~ L’ajout de cette interface graphique vient donc renforcer l’ou- tillage d’évaluation de SimFeodal. A cette étape, il manque encore un outil véritablement adapté à l’analyse conjointe des réplications du modèle, a pos- teriori de l’exécution des nombreuses réplications.

**5.2.2**

**Générer les indicateurs**

La production des indicateurs doit nécessairement être réalisée en aval de l’exé-

cution des simulations – « *offline* » dans Grignard et Drogoul (2017). Il faut pour cela disposer d’outils adaptés au traitement des données produites, c’est- à-dire répondant aux contraintes identifiées auparavant (sous-section 5.1.4).

16

***lue***

*2019-11-11 20:17:47*

--------------------------------------------

"Pour cette étape de pré-filtrage, nous...

***lue***

*2019-11-11 20:19:15*

--------------------------------------------

très sommaire directement dans Gama ?

***lue***

*2019-11-11 20:19:35*

--------------------------------------------

dans Gama ?

***lue***

*2019-11-11 20:22:50*

--------------------------------------------

encore répétition identique de ce que tu viens de dire. Proposition :

"Cette interface graphique permet de renforcer l'outillage d'évaluation de SimFeodal en permettant une première étape de filtrage des simulations. Pour mener une analyse conjointe et complète des réplications du modèle, en revanche, nous avons développé (ou "nous avons besoin d'") un outil entièrement adapté aux besoins et situés en aval de l'exécution de nombreuses réplications".

Chapitre 5

5.2. EXPLORER LES SORTIES

La contrainte principale est d’être en mesure de gérer la masse de données

produites. On l’a vu, cela élimine d’office les outils de type tableurs, ou encore les outils de manipulation graphique de données les plus courants. Pour les

raisons évoquées dans le chapitre 1 (

?), seules les solutions techniques libres étaient envisageables.

**Interfaces graphiques (GUI) ou en lignes de commande (CLI)** Cer-

tains outils graphiques (*GUI* – *Graphical User Interface*), basés sur des logiciels libres en arrière-plan (PSPP, R Commander, Orange), sont extrêmement aisés à prendre en main et auraient pu constituer un bon choix. Pourtant, avec une trentaine d’indicateurs à produire pour chaque expérience, donc de manière régulière, nous avons préféré nous tourner vers des outils plus orientés vers une interface en ligne de commande (*Command Line Interface*, abrégés *CLI*).

L’utilisation de CLI a plusieurs intérêts gravitant autour de la reproductibilité

des traitements. En premier lieu, ils permettent une adaptation aisée et ra- pide aux différents jeux de données. Ainsi, partant du principe que les données générées par les réplications et expérimentations sont de même structures, il suffit généralement de modifier le chemin d’entrée des fichiers résultants pour reproduire à l’identique une analyse sur un nouveau jeu de données.

De manière plus technique, on peut remarquer que les différents indicateurs de

sortie de simulation choisis présentent souvent des caractéristiques communes, aussi bien dans le traitement nécessaire que dans les formats (graphiques) pro- duits.

**Des outils CLI pour l’analyse de données.** Par exemple, la grande ma-

jorité des indicateurs reposent sur une première agrégation des données par réplication et pas de temps simulé, puis par une seconde agrégation montrant la variabilité des situations générées, au niveau de l’expérimentation 17. En terme de manipulation de données, seuls le calcul de la variable à mobiliser, et éventuellement l’agent caractérisé sont ainsi à adapter dans ces nombreux in- dicateurs de sortie. La variabilité du nombre de foyers paysans et la variabilité du nombre d’agrégats ne diffèrent que par le type d’agent sur lequel le calcul est effectué par exemple. Les variations, en terme de code-sources, sont donc le plus souvent des adaptations minimes (nom de l’agent, type d’agrégation…). Le recours à des traitements en *CLI* permet ainsi un simple copier/coller, voir la création de fonctions dédiées, pour effectuer ces traitements très récurrents.

**Des outils CLI pour la visualisation.** Au niveau des sorties graphiques,

~~on peut aussi remarquer que~~ la structure des graphiques, en elle-même, est assez largement identique : on représente les pas des temps (les années simu- lées) en abscisse, un indicateur statistique en ordonnée, et la variabilité sous la

17. On peut considérer ces agrégations comme une succession d’opération imbriquées : pour montrer l’évolution d’un indicateur tel que le taux de foyers paysans dispersés au cours du temps simulé, il faut (1) calculer le ratio entre nombre de foyers paysans dispersés et nombre total de foyers paysans, (2) pour chacun des pas de temps simulé, (3) pour chaque simulation,

(4) pour l’ensemble des réplications d’une expérience, (5) éventuellement pour chacune des expériences d’une phase plus large d’expérimentation qui ferait varier des valeurs de para- mètres.

17

***lue***

*2019-11-11 20:24:44*

--------------------------------------------

"De plus, pour..."

***lue***

*2019-11-11 20:27:44*

--------------------------------------------

"choix adaptés à nos besoins"

***lue***

*2019-11-11 20:27:59*

--------------------------------------------

"répétée"

***lue***

*2019-11-11 20:35:58*

--------------------------------------------

De plus

***lue***

*2019-11-11 20:36:36*

--------------------------------------------

ref au chapitre où tu les présentes car jusque là on a pas eu de liste d'indicateurs dans ce chapitre...

+ phrase à mettre en dessous du titre "des outils CLI pour l'analyse des données"

***lue***

*2019-11-11 20:40:20*

--------------------------------------------

du code

***lue***

*2019-11-11 20:41:25*

--------------------------------------------

"De même, au..."

***lue***

*2019-11-11 20:41:44*

--------------------------------------------

"entre les différents indicateurs"

outils libres

Positionnement : pourquoi utiliser des

Chapitre 5

5.2. EXPLORER LES SORTIES

forme de *box-plot* minimalistes (« *minimal boxplot* », promus par Edward Tufte

pour minimiser le ratio données-encre (Tufte 2001, p. 123-125)). En disposant d’un environnement de type *CLI*, et qui plus est en faisant usage de solutions graphiques construites sur une syntaxe régulière et générique (voir Fluidifier les étapes de rendu : le « pipeline de visualisation », section 5.4.2.1), il de- vient très confortable de générer les différents indicateurs de sortie souhaités, puisqu’il suffit d’adapter les graphiques déjà conçus.

Avec ces solutions logicielles d’analyse de données et de visualisation, il est

facile de concevoir et d’implémenter les codes informatiques nécessaires à la génération des indicateurs de sortie de simulation. De plus, l’exécution de ces programmes est extrêmement rapide : les différents fichiers de sortie de simu- lation sont lus et parcourus une unique fois pour en tirer toutes les variables nécessaires à l’établissement des indicateurs.

En ayant choisi de mener une évaluation *a posteriori* – plutôt qu’en direct –

basée sur l’observation d’indicateurs générés – par des outils adaptés au trai- tement de données massives – de manière automatisée, on dispose donc, pour chaque expérience, d’un ensemble de fichiers numériques : chacun des indica- teurs de sortie est contenu dans un fichier unique, dans un format facilement exploitable et ré-utilisable.

**5.2.3**

**Organiser les indicateurs en rapports paramétrables**

Du point de vue de la manipulation, la création de fichiers informatiques indé-

pendants correspondant aux différents indicateurs de sortie de simulation est extrêmement pratique : on peut facilement les identifier, les transférer et les adapter, par exemple pour en rendre le contenu plus compréhensible par un public différent.

En revanche, du point de vue de la comparaison des résultats, cette forme

n’est pas la plus adaptée. Si l’on peut facilement comparer un même indica- teur portant sur deux expériences différentes, la tâche se complique quand il s’agit d’avoir une vision globale des différences dans les indicateurs entre deux expériences. Pour cela, la démultiplication des fichiers correspondant aux in- dicateurs se révèle rapidement être un obstacle : l’utilisateur est en effet amené à jongler entre de très nombreux fichiers.

**Les rapports comme instruments de comparaison.** Pour faciliter la

comparaison d’indicateurs multiples ~~– et d’une forte diversité –,~~ il est néces- saire de les organiser au sein d’une structure englobante. Nous entendons ici par organisation, une présentation structurée, suivant un certain ordre, iden- tique selon les expériences, ~~adaptée à une évaluation des résultats.~~ Pour cela, nous avons choisi d’organiser les indicateurs de sortie de simulation au sein de « rapports ». Cela permet, même en présence de nombreuses expériences, de rassembler l’ensemble des indicateurs de sortie propres à chacune dans un unique fichier, à la structure toujours similaire. Un premier apport, majeur, concerne l’archivage des sorties de simulation. Avec des rapports comprenant l’ensemble des indicateurs de sortie de chaque expérience d’un modèle, il est 18

***lue***

*2019-11-11 20:52:16*

--------------------------------------------

"de concevoir, d'implémenter et de répliquer" ? il me semble que c'est la répétition/réplication qu'il faut mettre en valeur ici

***lue***

*2019-11-11 20:54:19*

--------------------------------------------

"et "

***lue***

*2019-11-11 20:55:44*

--------------------------------------------

il me semble que c pas une conclusion mais encore un avantage supplémentaire au choix que tu fais > rajouter un "enfin" en début de paragraphe ?

***lue***

*2019-11-11 21:54:44*

--------------------------------------------

de quoi ?

***lue***

*2019-11-11 21:57:09*

--------------------------------------------

ici je mettrai un petit exemple "un fichier pour l'indicateur x pour l'expérience x par exemple" et le "On peut facilement etc." dans une nouvelle phrase. Ca permettrai d'éclaircir le § du dessous.

***lue***

*2019-11-13 17:10:57*

--------------------------------------------

soit "et divers" soit rien en fait, car "multiples" y'a déjà le côté divers dedans. Ou nombreux et divers ?

***lue***

*2019-11-13 17:11:38*

--------------------------------------------

entre

***lue***

*2019-11-13 18:25:01*

--------------------------------------------

pour moi cette idée va plus loin que "organisation". En plus évidement tu veux qu'elle soit adaptée à l'évaluation

***lue***

*2019-11-13 20:27:25*

--------------------------------------------

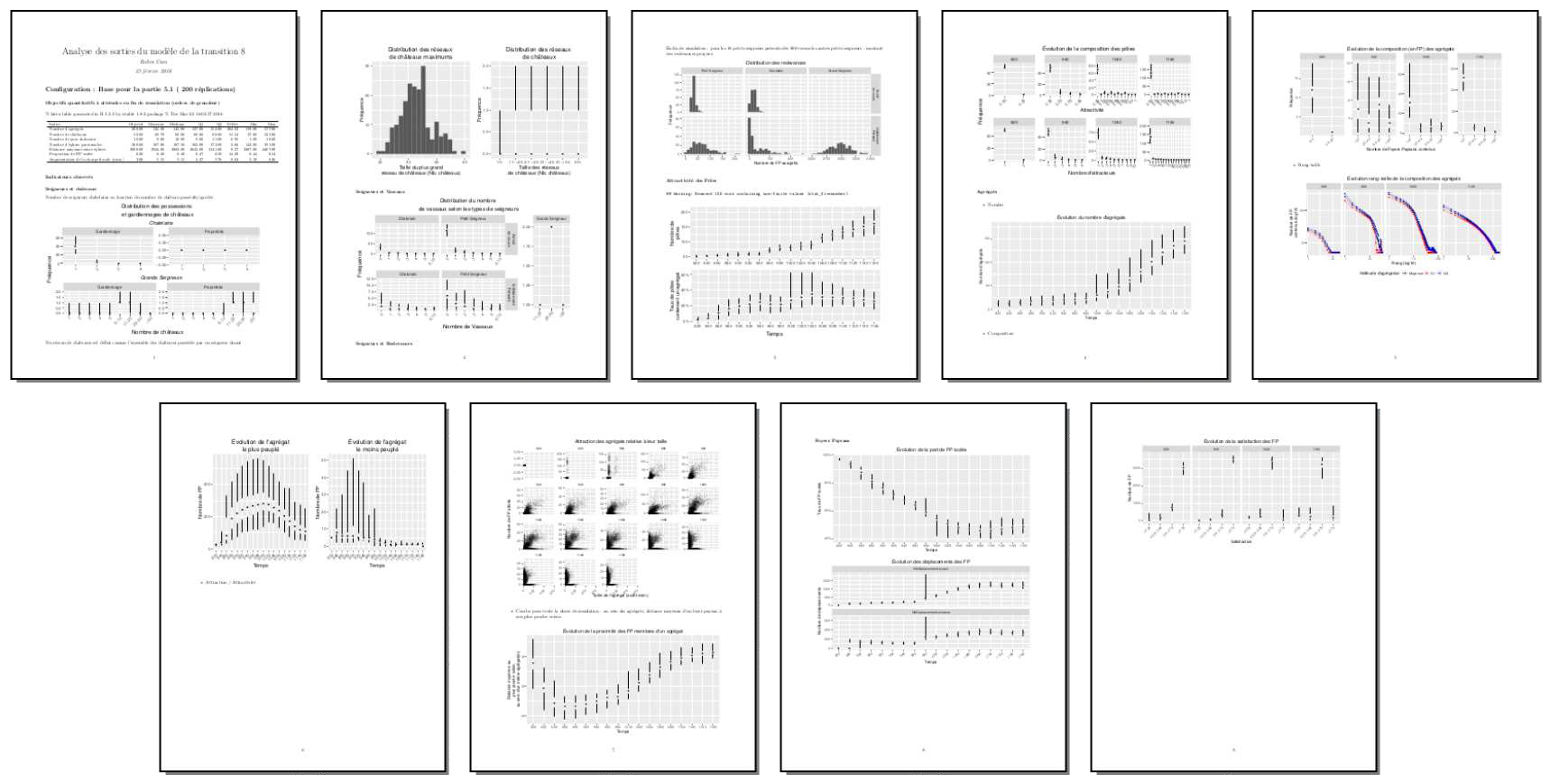
répétition avec au dessus : simplifier tout ce paragraphe

***lue***

*2019-11-13 20:27:37*

--------------------------------------------

de consigner ? pour éviter la répétition...



Chapitre 5

5.2. EXPLORER LES SORTIES

~~ainsi~~ simple de conserver des traces de l’ensemble des versions et sous-version

d’un modèle. Cela permet de garantir une certaine pérennité à ce modèle, à sa documentation, et ainsi simplifie le travail rétrospectif de caractérisation de l’évolution d’un modèle.

L’intérêt majeur de la structuration en rapports est surtout de faciliter la com-

paraison des expériences, c’est-à-dire des indicateurs de sortie des différentes expériences. On peut ainsi, par exemple, placer côte à côte, visuellement, deux rapports rendant compte de deux expériences différentes, et, en les faisant dé- filer simultanément, comparer point par point, c’est-à-dire indicateur par indi- cateur, leurs résultats respectifs, de manière visuelle et intuitive.

Les formes que peuvent prendre des rapports, tout autant que les modalités

de leur production, sont multiples et extrêmement diverses. La forme la plus simple et courante consiste à produire manuellement le rapport en insérant les indicateurs adaptés au fur et à mesure, par exemple dans un traitement de texte. A l’opposé, on peut noter les possibilités de créations entièrement au- tomatisé~~es~~ de rapports complets, comprenant par exemple des descriptions et commentaires textuels générés à la volée, en fonction d’expressions condition- nelles 18.

Figure 5.3 – Un exemple, miniaturisé, de rapport automatique généré pour

une expérimentation (étape 0) de SimFeodal.

Pour SimFeodal, nous avons choisi de restreindre au maximum la manipu-

lation manuelle. On souhaitait générer un rapport entièrement automatique, ne requérant pas d’action spécifique en dehors du choix des données depuis lesquelles créer les indicateurs. Nous n’avons toutefois pas voulu pousser l’au- tomatisation jusqu’à l’ajout de commentaires automatiques des indicateurs de sortie : la richesse – et la difficulté– d’une approche interdisciplinaire telle que la notre est constituée par la multiplication des analyses et points de vue. Il n’y

18. Voir par exemple l’application « SOFIE » du Commissariat général à l’égalité des territoires (CGET), qui génère automatiquement des commentaires relatifs aux inégalités femmes/hommes dans l’accès à l’emploi. Les commentaires propres aux types d’inégali- tés majeures de chaque EPCI sont produits de manière automatique depuis les données. http ://outils.observatoire-des-territoires.gouv.fr/sofie/

19

***lue***

*2019-11-13 20:29:11*

--------------------------------------------

"Deuxièmement, l'intérêt" (pour répondre au "premier apport" )+ pas de saut de paragraphe

***lue***

*2019-11-13 20:29:43*

--------------------------------------------

"de son évolution"

***lue***

*2019-11-13 20:30:40*

--------------------------------------------

"par sa documentation" plutôt ? Dans ce cas "La documentation permet ainsi de garantir une certaine pérennité au modèle et simplifie le travail etc."

***lue***

*2019-11-13 20:32:32*

--------------------------------------------

répétition à supprimer

***lue***

*2019-11-13 20:33:09*

--------------------------------------------

"nombreuses et diverses" ou "multiples" ou "extrêmement diverses"...

***lue***

*2019-11-13 20:33:54*

--------------------------------------------

là c'est la modalité de production dont tu parles, pas de la forme...

***lue***

*2019-11-13 20:35:01*

--------------------------------------------

"il existe des modes"

***lue***

*2019-11-13 20:35:27*

--------------------------------------------

si tu fais ma modif du dessus

***lue***

*2019-11-13 20:38:51*

--------------------------------------------

kesako ? tu peux peut-être diminuer cette note en mode "qui génère automatiquement depuis les données des commentaires relatifs aux inégalités f/h dans l'accès à l'emploi + url."

***lue***

*2019-11-13 20:43:06*

--------------------------------------------

figure à déplacer plus bas je ne me rends pas compte mais peut-être beaucoup trop petite à l'impression (et du coup elle va être jugée inutile)

***lue***

*2019-11-13 20:40:54*

--------------------------------------------

Nous

Chapitre 5

5.2. EXPLORER LES SORTIES

avait donc aucun besoin de générer des annotations standardisés et automa-

tiques, forcément moins aboutis que les analyses de chacun des co-concepteurs du modèle. Le rapport produit (figure 5.3) n’intègre donc que les indicateurs, sous forme de tableaux et de graphiques. Ces indicateurs sont organisés par partie, en l’occurrence en fonction du type d’entités et de comportement qu’ils décrivent.

**Structurer des rapports pour aller vers la reproductibilité des ana-**

**lyses.** On a donc fait le choix de se baser sur des rapports automatisés et minimalistes, ne contenant que les indicateurs dans une forme structurée. Ce choix s’appuie sur des raisons multiples, qui ont toutes en commun une re- cherche de reproductibilité des résultats et des analyses menées. Une repro-

ductibilité théorique (

ment), puisque les résultats

de simulation doivent pouvoir être analysés et reproduits par des chercheurs

potentiellement intéressés. Mais cette recherche de reproductibilité est aussi pratique, rendue aussi nécessaire par les méthodes de modélisation suivies. Tel qu’explicité auparavant (sections 5.1.2 et 5.1.3), celles-ci s’appuient sur de nombreux allers-retours, ce qui requiert une capacité constante à reproduire et à affiner des résultats déjà observés.

La quantité d’expériences requises pour arriver à un état satisfaisant du modèle

est tributaire de ces allers-retours, et le nombre de rapports qu’il faut pouvoir produire est important. La fréquence de production de ces rapports est forte, et le modélisateur a alors tout intérêt à en fluidifier et accélérer le processus de création.

Dans une telle situation, la création d’un rapport automatisé garantit un~~e~~ calcul et une production simplifié et rapide des indicateurs sur les nouvelles données. Cela permet un examen des sorties de simulation presque immédiatement après leur exécution. Cette automatisation permet aussi de mener une seconde éva- luation – après le pré-filtrage constitué par l’observation d’indicateurs en direct

– du bon déroulement « interne » 19. Le caractère fixe d’un rapport automatisé se base ~~ainsi~~ sur une structure de données contraignante, par exemple consti- tués en *n* fichiers dotés de plusieurs colonnes spécifiquement attendues. Les caractéristiques de ces données sont elles-mêmes contraignantes. Un rapport automatisé ne fonctionne, par exemple, qu’en présence d’un nombre pré-défini de réplications complètes. En l’absence d’un de ces critères dans des données en sortie de simulation, le rapport ne peut être généré et émet une erreur. Par exemple, si le nombre de réplications est plus faible qu’attendu, ou encore si tel attribut d’un agent a changé de type informatique, la création du rapport échoue. La présence ou non de cette erreur constitue donc un nouveau filtre de vérification de la validité du modèle. Cela permet, là encore, de détecter des simulations qui présenteraient des comportements incomplets ou aberrants en terme de production de données.

Un autre intérêt majeur des rapports, déjà pointé en avantage des outils de type

*CLI* est leur adaptabilité. On a vu (

3) que les indicateurs à examiner sont

19. Au sens de l’évaluation interne, c’est-à-dire du bon fonctionnement, exempt de *bugs*, du

modèle de simulation implémenté. Voir

3.

20

***lue***

*2019-11-13 20:41:23*

--------------------------------------------

abouties

***lue***

*2019-11-13 20:41:29*

--------------------------------------------

standardisées

***lue***

*2019-11-13 20:41:48*

--------------------------------------------

à mettre à la fin de la phrase

***lue***

*2019-11-13 20:42:39*

--------------------------------------------

que dans l'exemple donné que c organisé comme ça ? Si non virer le "en loccurence"

***lue***

*2019-11-13 20:44:09*

-------------------------------------------- L

***lue***

*2019-11-13 20:45:04*

--------------------------------------------

"La reproductibilité est d'abord théorique" (en tous cas pas de phrase nominative) + pas sûre que l'appel soit important ici tu as déjà pas mal renvoyé au positionnement pour la reproductibilité

***lue***

*2019-11-13 20:46:55*

--------------------------------------------

"du fait des"

***lue***

*2019-11-13 20:47:29*

--------------------------------------------

Comme nous l'avons exposé ?

***lue***

*2019-11-13 20:47:39*

--------------------------------------------

ces méthodes

***lue***

*2019-11-13 20:50:31*

--------------------------------------------

simplifiés et rapides

***lue***

*2019-11-13 20:51:23*

--------------------------------------------

"du modèle."

***lue***

*2019-11-13 20:58:23*

--------------------------------------------

hyper clair et intéressant, mais je trouve ça bizarre dans le déroulement parce que tu présentes là comment en gros techniquement ça marche, et on voit pas trop le rapport au début. Je dirais donc que ça serait plus intéressant de commencer avec ce gros morceau là comme un nouveau paragraphe, et deconclure avec la phrase juste avant : "L'automatisation permet donc de merner une seconde évaluation etc."

***lue***

*2019-11-13 20:58:40*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-13 20:59:14*

--------------------------------------------

supprimer surtout que c aussi ce que tu dis au début de ce chapitre là !

ref chap

chap.

encore une ref au positionne

Chapitre 5

5.2. EXPLORER LES SORTIES

nombreux et surtout, évolutifs, dans le sens où ces indicateurs ont fortement été

modifiés, remplacés, affinés, au cours des étapes de paramétrage de SimFeo- dal. L’utilisation de rapports automatiques permet de minimiser le nombre de modifications à effectuer en cas de changements d’indicateurs. Le programme informatique qui génère les rapports s’appuie ~~a insi~~ sur un code-source unique, générique aux simulations ~~de chaque modèle.~~ Lors d’un changement d’indica- teurs, il suffit alors de modifier ce code-source en une seule place, et tous les appels à ce programme seront alors modifiés en conséquence. À partir de là, pour mettre à jour l’ensemble des rapports déjà produits, c’est-à-dire regrou- pant les indicateurs de chacune des expériences passées, il suffit de ré-exécuter la routine de production des rapports.

Dans le cas de SimFeodal, caractérisé par de fréquents changements dans la

forme et le calcul des indicateurs, cela a représenté un gain de temps et d’effi- cacité très conséquent. Par exemple, lors de certaines phases de paramétrage, on pouvait être amenés à faire évoluer le modèle quotidiennement et à tester, à cette même fréquence, plusieurs jeux de paramètres. Il fallait donc analyser les résultats de plusieurs expériences chaque jour, et régulièrement ajouter des in- dicateurs graphiques afin d’affiner l’évaluation. Ces indicateurs devaient aussi être ajoutés aux expériences des jours précédents, et au final, on re-générait parfois jusqu’à une plus d’une dizaine de rapports sur des cycles temporels courts de quelques jours.

Par extension, cette même démarche d’automatisation, basée sur l’utilisation

d’outils de type *CLI*, devrait pouvoir s’appliquer à l’identique, avec les mêmes

avantages, dans le cadre plus large de l’évaluation visuelle de modèles (

3.1.4).

**Dépasser les limites de la compatibilité d’ensemble.** La reproducti-

bilité, de manière générale, est plus une démarche qu’un état final : on peut toujours trouver un nouvel élément à « rendre reproductible » dans un travail

(

chée). Dans le cas des rapports

automatiques, la reproductibilité recherchée doit permettre de reproduire les

analyses menées – le calcul et la représentation des indicateurs de sortie – sur les jeux de données issus des différentes versions du modèle.

Comme explicité dans l’

sur l’incrémentalité des indica-

teurs, une limite forte empêche d’atteindre une reproductibilité absolue des

analyses du comportement des différentes versions de SimFeodal. Les données générées par les différentes versions du modèle ne sont en effet pas systéma- tiquement « compatibles ». On entend par là qu’elles ne présentent pas toute exactement la même structure, à commencer par les variables enregistrées. Quand bien même il aurait été choisi dès le départ d’enregistrer le plus de sorties possibles, la reproductibilité de l’analyse échoue sur les données pro- duites par le modèle et ses nombreuses versions : le modèle évolue, et avec lui, certaines variables apparaissent et d’autres deviennent caduques. La structure contraignante et précise des données nécessaires à la génération des rapports ne peut être entièrement satisfaite. La prise en compte de l’évolution du mo- dèle demande une adaptation régulière – mais aussi rare que possible – des 21

***lue***

*2019-11-13 21:01:53*

--------------------------------------------

singulier si "on", pluriel si "nous"

encadré 3.2, chap 3

cf. chapitre 1, partie sur reproductibilité recher

chap 3, section

ref

Chapitre 5

5.2. EXPLORER LES SORTIES

programmes qui génèrent ces rapports.

On ne peut donc satisfaire globalement à un objectif de reproductibilité, mais

il est toutefois possible de limiter la déviance à cette ambition. Pour cela, on peut agencer les différentes versions du modèles au sein de « générations » de modèle, c’est-à-dire d’ensembles de versions présentant des attributs com- parables et générant des données de même structure. Plutôt que d’adapter le code-source des rapports à chaque nouvelle version du modèle, ou encore de ne jamais l’adapter et donc d’être tributaire de la structure des toutes premières versions du modèle, cela constitue un choix intermédiaire qui permet de limi- ter le nombre de variantes de rapports. Cette approche suit les grandes lignes du développement logiciel général. Les itérations successives d’un logiciel sont constituées de versions « majeures » – les générations de modèles dans notre cas –, qui n’assurent pas nécessairement de compatibilité avec les versions ma- jeures précédentes, et de versions « mineures », dans lesquelles la compatibilité est assurée 20.

Pour revenir aux rapports voués à l’évaluation d’un modèle, en inscrivant les différentes versions du modèle – et des programmes générant les rapport cor- respondant – dans des sous-ensemble de versions, les « générations », les diffé- rents rapports peuvent être considérés comme reproductibles et automatiques au sein de ces générations. Pour SimFeodal, cela implique d’organiser les diffé-

rentes versions du modèle – résultant des étapes de paramétrage,

– au

sein de grandes générations, à chaque changements structurels des mécanismes

ou données produites par le modèle (voir

sions).

**Les rapports, des instruments suffisants ?** A l’issue de la conception

et de l’implémentation de ces rapports automatiques, on dispose donc, pour chaque expérience, d’un document aisément partageable et lisible. Ces docu- ments s’enrichissent, au fur et à mesure des générations de modèles, de nou- veaux indicateurs, et sont comparables au sein de ces générations. Cela pourrait constituer la dernière étape de la création d’outils d’évaluation d’un modèle, dans la limite d’un nombre de versions ou de génération de modèles assez restreint.

SimFeodal, comme c’est souvent le cas dans les modèles à base d’agent, a tou-

tefois été caractérisé par une forte quantité d’allers-retours entre le modèle et ses résultats, entraînant à chaque fois de nouvelles expérimentations (cf. sec- tion 5.1).

On a vu que la manipulation d’un grand nombre d’indicateurs, même pour une

quantité restreinte d’expériences, disqualifiait l’usage de fichiers individuels et poussait à l’usage de rapports structurés. Avec un grand nombre d’expériences, les mêmes limites apparaissent pour les rapports : la masse d’expériences rend partiellement caduque l’utilisation unique des rapports automatiques. Il est en

20. Par exemple, un fichier de dessin vectoriel créé avec le logiciel Adobe Illustrator 15.0 ne sera pas lu correctement avec une version 14.0. Ce fichier présentera toutefois un compatibilité parfaite avec les versions 15.1 à 15.*n* du logiciel.

22

***lue***

*2019-11-13 21:24:08*

--------------------------------------------

expression pas claire

***lue***

*2019-11-13 21:27:07*

--------------------------------------------

assembler, rassembler plutôt

***lue***

*2019-11-13 21:27:38*

--------------------------------------------

"le travail par générations"

***lue***

*2019-11-13 21:29:13*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-13 21:30:00*

--------------------------------------------

éventuellement là aussi pas de saut de paragraphe

sous-ver

chapitre 4, tableau des versions et

cf. chap4

Chapitre 5

5.2. EXPLORER LES SORTIES

effet aisé de comparer, sur un même écran d’ordinateur, deux ou trois rapports,

mais dès lors qu’il faut en comparer un plus grand nombre, la manipulation conjointe des rapports devient complexe, tout autant que d’avoir une vision globale des résultats principaux de chaque expérience.

**5.2.4**

**Organiser les rapports : les *dashboards***

Pour être en mesure de comparer de nombreux éléments, il est nécessaire

de passer d’une exploration linéaire, fondée sur le visualisation successive de chacun des indicateurs, à une exploration globale et interactive. En pratique, plutôt que de faire défiler visuellement les nombreuses pages d’indicateurs, mieux vaut une interface présentant les points clefs de l’évaluation et qui per- mette d’entrer dans le détail de chacun des indicateurs dans un second temps, sur demande. Comme le résume le « mantra » de l’analyse visuelle (« *Visual information-seeking mantra* ») de Ben Shneiderman :

« Overview first, zoom and filter, then details on demand »

Shneiderman 1996, p. 2

**5.2.4.1 Les *dashboards***

Cette logique, assez universelle désormais, est celle qui préside à la création des nombreux « tableaux de bord », ou « *dashboards* » que l’ont voit émerger depuis la fin des années 1990. Rob Kitchin et ses co-auteurs définissent ainsi les *dashboards*, en s’appuyant sur les travaux de Few ~~n otamment~~ :

« For Few [(Few 2006a, p.34)] a ‘dashboard is a visual display

of the most important information needed to achieve one or more objectives ; consolidated and arranged on a single screen so the in- formation can be monitored at a glance’. Just as a car dashboard provides critical information needed to operate the vehicle at a glance, indicator dashboards provide key information for running companies or cities (Dubriwny & Rivard, 2004 [Rivard et Cog-

swell 2004]). »

Kitchin, Lauriault et McArdle 2015, p. 11

Très répandus dans le monde de l’informatique décisionnelle (*Business Intel-*

*ligence, BI*), ces outils permettent d’explorer des données d’entreprises, par exemple des résultats financiers. Pour ce faire, ils mettent en avant, dans une interface unique, des indicateurs clés (*Key Performance Indicators, KPI*), qu’il est ensuite possible de filtrer et d’affiner, par exemple par sélection de différents intervalles temporels.

Les *KPI* jouent le rôle d’indicateurs synthétiques, c’est-à-dire qu’ils s’adressent à

des gestionnaires, par exemple des *managers*, qui ont une expertise importante sur les résultats produits. Les utilisateurs des *dashboards* ne sont donc pas des analystes, à même d’explorer eux-mêmes les données mobilisées, mais plutôt des thématiciens qui se fondent sur les indicateurs présentés pour prendre des décisions.

23

***lue***

*2019-11-13 21:39:04*

--------------------------------------------

utiliser

***lue***

*2019-11-13 21:39:34*

--------------------------------------------

.

***lue***

*2019-11-13 21:40:10*

--------------------------------------------

notamment

***lue***

*2019-11-13 21:43:19*

--------------------------------------------

"importante mais non technique (par exemple mathématique)" ? pour faire le lien explicitement avec les thématiciens.

Chapitre 5

5.2. EXPLORER LES SORTIES

La dichotomie « analyste/décisionnaire » s’exprime aussi dans le domaine de

la recherche, et notamment dans la recherche en géographie urbaine à visée applicative. Avec l’avènement des données massives et de leur prise en compte pour la gestion des villes (*smart cities*), les géographes se sont aussi penchés sur des outils de ce type (Laurini 2018). En résulte une utilisation de plus en plus fréquente de *dashboards* en géographie urbaine (« *city-dashboards* », Roumpani, O’Brien et Hudson-Smith 2013 ; Kitchin, Lauriault et McArdle 2015 ; BattY 2015).

Le parallèle avec le monde de l’informatique décisionnelle est en effet présent

dans les types d’utilisateurs et de producteurs de ces outils. Il s’agit de mettre à disposition d’experts thématiques (les décideurs publiques) des indicateurs clés, issus de calculs parfois complexes. Cela afin de leur permettre d’évaluer une situation donnée et de prendre les décisions politiques adéquates.

Le constat ayant mené à l’apparition des premiers *dashboards*, tant en informa-

tique décisionnelle qu’en géographie urbaine, est identique. Les informations nécessaires à l’évaluation d’une situation (financière, relative aux politiques publiques…) sont de plus en plus nombreuses et hétérogènes. Les indicateurs permettant de mener ces évaluations, pensés pour les décideurs qui en feront usage (*managers*, acteurs publiques…), se démultiplient et se diversifient aussi par conséquence.

Inspiré autant par l’usage *BI* que par l’usage géographique, nous considérons

que ces outils peuvent se révéler utiles dans l’évaluation de modèles de simula- tions complexes. Les enjeux sont en effet les mêmes : permettre à des thémati- ciens de comprendre et d’évaluer les sorties d’un modèle, à l’aide d’indicateurs nombreux et complexes présentés de manière transparente relativement à cette complexité.

Ce positionnement méthodologique s’inscrit pleinement dans la démarche de

co-construction interdisciplinaire de SimFeodal. On y retrouve ~~a insi~~ la même logique qui anime les dashboards : une évaluation menée par des thématiciens

qui s’appuient pour cela sur des indicateurs clefs (

3) et précisent leur analyse à l’aide d’indicateurs se-

condaires présentés sous forme d’un panel varié de visualisations. Nous avons

donc choisi de ré-organiser les rapports initialement produits pour leur donner une forme plus adaptée à ces enjeux, sous forme de *dashboards*.

**5.2.4.2 SimVADB**

Les *dashboards* font souvent usage~~s~~ de représentations graphiques très méta- phoriques des tableaux de bords automobiles. On y retrouve fréquemment une forte mise en valeur d’indicateurs numériques simples au travers de représen- tations « skeuomorphes », c’est-à-dire qui reprennent l’apparence physique des objets symbolisés 21. On retrouve communément, par exemple, des indicateurs représentés sous forme de jauges (*gauge charts*), de thermomètres (*thermometer charts*), ou encore de voyants d’alerte et autres témoins lumineux (figure 5.4)~~).~~

21. Voir par exemple la page Wikipédia consacrée : https ://fr.wikipedia.org/wiki/Skeuomorphisme

24

***lue***

*2019-11-13 21:44:12*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-13 21:45:08*

--------------------------------------------

", cela" (en tous cas pas de phrase nominative)

***lue***

*2019-11-13 21:47:04*

--------------------------------------------

en ?

***lue***

*2019-11-13 21:47:37*

--------------------------------------------

"En nous inspirant" ? Ou "Inspirés"

***lue***

*2019-11-13 21:48:02*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-13 21:48:40*

--------------------------------------------

"en effet dans cette co-construction"

***lue***

*2019-11-13 21:54:26*

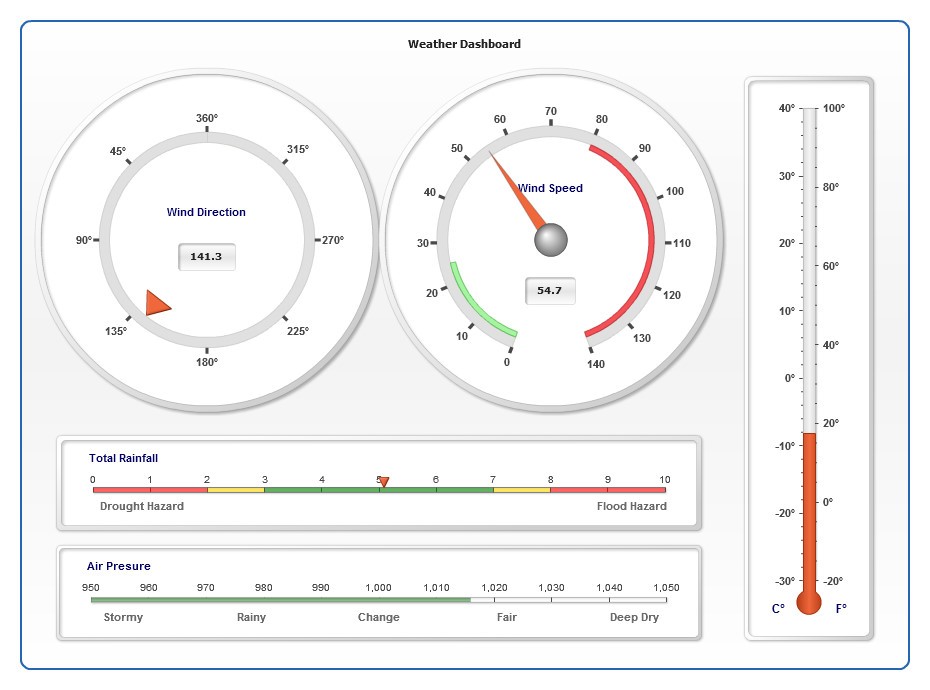
--------------------------------------------

un peu répétitif avec le paragraphe du dessus même si le propos est légèrement différent. Proposition pour fusionner les 2 :

"Inspiré autant par l’usage BI que par l’usage géographique, nous considérons que ces outils peuvent se révéler utiles dans l’évaluation de modèles de simulations complexes, et en particulier dans une démarche de co-construction interdisciplinaire. Les enjeux sont en effet les mêmes : permettre à des thématiciens de comprendre des données complexes. Dès lors, la logique qui anime les dashbords répond à nos besoin : permettre une évaluation et une analyse du modèle par des thématiciens par le biais d'indicateurs clefs et d'indicateurs secondaires (reférence), présentés sous la forme d'un panel varié de visualisations. Nous avons donc choisi de ré-organiser les rapports initialement produits pour leur donner une forme plus adaptée à ces enjeux, sous forme de dashboards."

importants dans le chap

ref. aux indicateurs les plus



Chapitre 5

5.2. EXPLORER LES SORTIES

Figure 5.4 – Un exemple de représentations visuelles courantes dans les *dash-*

*boards*. Tiré de Pandre (2011)

Pour SimFeodal, les indicateurs étant assez fortement conçus et structurés

(

3), nous n’avons pas ressenti le besoin de faire appel à ce type de repré-

sentation. Nous avons donc emprunté aux *dashboards* la logique d’organisation

visuelle des indicateurs plutôt que les modes de visualisation en eux-mêmes. Pour faciliter la transition pour l’utilisateur, on cherchait ~~a insi~~ à produire un *dashboard* au plus proche, visuellement, des rapports automatiques qui les pré- cédaient.

On a pour cela développé un *dashboard* adapté à SimFeodal, nommé Sim-

VADB 22. Dans un premier temps, on souhaitait simplement ré-organiser le code-source produisant les rapports automatiques, afin de convertir ces rap- ports en *dashboards*. Cela a été effectué au moyen d’outils permettant de gé- nérer des applications en ligne, sans changer de langage de programmation. Dans ce cas, on s’est appuyé sur la librairie logicielle Flexdashboard (Iannone, Allaire et Borges 2018).

Le passage du rapport automatique au *dashboard* illustre l’un des grands inté-

rêts des outils de type *CLI* : dans le cas de SimVADB, il a suffit de ré-organiser le code, sans modifier à aucun moment les fonctions de calcul et de création des indicateurs de sortie de simulation. Les codes 5.1 et 5.2 illustrent la facilité de cette modification. Il s’agit véritablement de placer les différentes fonctions dans des blocs graphiques. Ces modifications minimes augmentent toutefois considérablement la convivialité et la facilité de l’analyse de résultats de sortie d’un modèle.

22. **S**imulation **V**isual **A**nalysis **D**ash**B**oard.

Cette application a rapidement été remplacée par l’itération suivante (SimEDB, voir sec- tion 5.2.5), et n’a donc dans les faits jamais été complètement finalisée. On en trouve une trace, fonctionnelle mais incomplète (les versions ultérieures n’ont pas été enregistrées dans l’outil de versionnement), dans ce dépôt logiciel : https ://github.com/RCura/SimEDB/tree/2cd22c7c 25

***lue***

*2019-11-13 21:56:38*

--------------------------------------------

tu ne mets pas juste la ref en biblio ?

***lue***

*2019-11-13 22:00:40*

--------------------------------------------

passage nous > on ?

***lue***

*2019-11-13 22:01:56*

--------------------------------------------

de quel langage s'agit-il du coup ? Ici mérite explicitation (d'autant que je ne crois pas que tu dises en quoi tu programmes les rapports)

***lue***

*2019-11-13 22:02:39*

--------------------------------------------

"un premier"

***lue***

*2019-11-13 22:03:21*

--------------------------------------------

uniquement

***lue***

*2019-11-13 22:03:36*

--------------------------------------------

": il"

***lue***

*2019-11-13 22:03:47*

--------------------------------------------

pas de saut de paragraphe

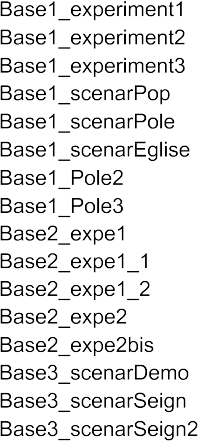
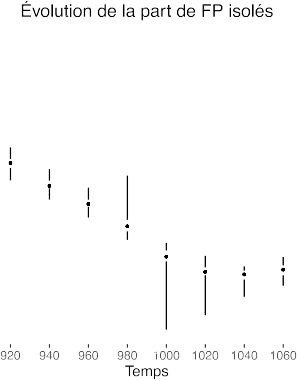
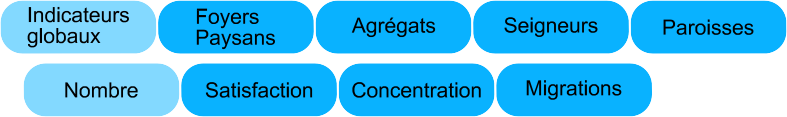
***lue***

*2019-11-13 22:04:13*

--------------------------------------------

pourquoi ??? tu veux dire dans le code ?

chap.



Chapitre 5

5.2. EXPLORER LES SORTIES

Code 5.1 – Pseudo-code du rapport

automatique.

Code 5.2 – Pseudo-code du dashboard.

Comme dans l’exemple de code, on a préféré organiser les indicateurs au seins

d’onglets plutôt que de les présenter dans des pages successives. Les onglets de premier niveau représentaient les types d’agent, et des onglets de second ni- veau permettaient de visualiser l’ensemble des indicateurs associés à ces agents (figure 5.5).

Figure 5.5 – Un *mock-up* 23de la première interface de SimVADB, un *dashboard*

dédié à la visualisation des indicateurs de sorties de simulation de SimFeodal.

Au niveau de l’interface utilisateur, SimVADB permettait de choisir, via un menu de sélection (partie de gauche dans la figure 5.5), les expériences passées dont on voulait visualiser les indicateurs de sortie (les deux niveaux d’onglets de la partie de droite).

**Les limites du *dashboard*.** Avec la multiplication des valeurs de para-

mètres testées, il est devenu plus efficace de regrouper les expériences au sein d’expérimentations. Celles-ci voient varier plusieurs paramètres, potentielle- ment avec de multiples valeurs de paramètres pour chacun. Elles constituent

23. Un *mock-up* est une « maquette à l’échelle », ou un prototype, permettant de représenter une interface graphique de manière schématique.

26

***lue***

*2019-11-13 22:04:35*

--------------------------------------------

une seule légende pour les deux non ? cça fait bizarre comme ça

***lue***

*2019-11-13 22:05:24*

--------------------------------------------

mettre au présent

***lue***

*2019-11-13 22:05:29*

--------------------------------------------

mettre au présent

***lue***

*2019-11-13 22:06:27*

--------------------------------------------

"Par ailleurs, on a choisi d'organiser"

***lue***

*2019-11-13 22:06:55*

--------------------------------------------

pourquoi tu n'utilises pas simplement "prototype" dans la légende ?

***lue***

*2019-11-13 22:08:16*

--------------------------------------------

mettre au présent - ou bien pour toute cette série de verbe vérifier dès le début de la sous-partie le temps employé. Je pense que si tu veux utiliser l'imparfait, il faut préciser directement dans le texte qu'il s'agit d'une première version de dashboard "finalement abandonnée" (là je crois que ça ne vient qu'en note)

***lue***

*2019-11-13 22:10:31*

--------------------------------------------

je mettrai ce paragraphe en premier, puis celui sur les 2 niveaux d'onglets (et du coup pas vraiment la peine de parler de choix d'onglets plutôt que de pages, etc. je pense que la précision est pas tellement utile ici) : "dont on voulait/veut visualiser les indicateurs de sortie (partie de droite). Ces derniers sont organisés au seins d'onglets : les onglets de premiers niveau etc."

***lue***

*2019-11-13 22:15:27*

--------------------------------------------

supprimer ce sous paragraphe et intégrer le texte au début du suivant ("une interaction trop simple")

# Agent de type A afficher (' Agent de type A ')

print (' Indicateur 1 ') calcul\_indicateur\_1 {...} affichage\_indicateur\_1 {...}

afficher (' Indicateur 2 ') calcul\_indicateur\_2 {...} affichage\_indicateur\_2 {...}

# Agent de type B afficher (' Agent de type B ') [...]

onglet { titre = ' Agent de type A ', sous\_onglet { titre = ' Indicateur 1',

calcul\_indicateur\_1 {...} affichage\_indicateur\_1 {...}

},

sous\_onglet { titre = ' Indicateur 2', calcul\_indicateur\_2 {...} affichage\_indicateur\_2 {...}

}

},

onglet { titre = ' Agent de type B ', [...]

}

Chapitre 5

5.2. EXPLORER LES SORTIES

donc un ensemble d’expériences

paramètre par défaut communs.

qui partagent des mécanismes et un jeu de

**Une interaction trop simple.** Avec le mode de sélection choisi dans Sim-

VADB, basé sur le nom des expériences, il devenait plus difficile de sélectionner rapidement des ensembles d’expériences membres d’une même phase d’expé-

rimentation (cf.

4). En effet, et comme

illustré dans la figure 5.5, les noms d’expériences tendent à s’allonger, et avec

leur masse augmentant, il est peu commode d’avoir à parcourir tout un long menu de sélection pour trouver les expériences souhaitées. De plus, malgré des tentatives de nommage régulières et explicites, la multiplication des ex- périences et expérimentations implique aussi une certaine confusion dans les types de mécanismes et valeurs de paramètres associés. Sans table de corres- pondance complète entre les noms des expériences et leurs valeurs de para- mètres, il devenait impraticable de retrouver les différentes expériences met- tant en avant, par exemple, des attractivités fortes par les pôles, une plus forte hiérarchisation des attracteurs ou encore des migrations lointaines facilitées. Le choix méthodologique d’interaction avec la plate-forme d’affichage des in- dicateurs, basé sur une sélection des expériences depuis leur nom, s’est donc révélé inadapté à la sélection et à l’exploration des sorties de SimFeodal.

**Dépasser le « présentoir » de données.** Au delà du souci du mode d’interac-

tion, qui aurait pu être amélioré, un autre problème apparaissait. Pour évaluer visuellement différentes configurations du modèle, on ne pouvait se conten- ter d’un simple affichage des données, au sein d’un outil de type présentoir

interactif tel que SimVADB. Comme dit dans

4, le paramétrage de

SimFeodal a ~~a insi~~ reposé sur de nombreuses étapes d’évaluation des différentes

version du modèle. L’approche d’analyse principale était donc la comparaison, point par point, entre les résultats des indicateurs de sortie de simulation des versions successives de SimFeodal. Un outil de présentation dynamique de ré- sultats de sorties de simulations est certes plus adapté qu’un rapport statique, mais il ne constitue pas pour autant un outil adapté à la comparaison. S’il suffit pour de la restitution, par exemple dans le cadre du rapport systématique des résultats de SimFeodal, on ne peut s’appuyer uniquement sur une succession d’évaluations visuelles pour appréhender l’étendue des changements apportées par une modification des valeurs de paramètres.

**5.2.5**

**Interagir avec les rapports : exploration interactive**

Face à la démultiplication des expérimentations, consécutive aux nombreuses

étapes de paramétrage de SimFeodal (

4), il a fallu repenser la plate-

forme d’évaluation des résultats. Pour cela, considérant que les simulations ne

pouvaient être aisément appréhendées et sélectionnées par leur nom, numéro d’étape ou de version, il a été décidé d’adopter une posture plus proche de l’exploration du modèle en elle-même. C’est-à-dire de ne pas caractériser les

~~simulations par un identifiant quelconque, mais plutôt~~ par leur spécificité in-

27

***lue***

*2019-11-13 22:14:56*

--------------------------------------------

je ne comprends pas ces différents "sous paragraphes", il y a beaucoup de niveaux déjà... et là ces sous-niveaux apparaissent alors que jusque là il n'y en avait pas. Si tu tiens absolument à garder ceux là je dirais qu'il faut que tu fasses :

/Premier titre générique sur les dashboard ???/

/Un premier dahsboard : SimVADB/

/Une interaction trop simple/

/Dépasser le présentoir de données/

***lue***

*2019-11-13 22:15:55*

--------------------------------------------

l'augmentation de leur masse

***lue***

*2019-11-13 22:17:39*

--------------------------------------------

"est devenu" > dans ce paragraphe tout mettre au passé (et dans ce cas choisir : passé simple, imparfait?) ou tout mettre au présent

***lue***

*2019-11-13 22:18:23*

--------------------------------------------

des limites

***lue***

*2019-11-13 22:18:55*

--------------------------------------------

qu'avait été construit

***lue***

*2019-11-13 22:30:47*

--------------------------------------------

un peu long et répétitif, y compris avec des choses déjà dites, notamment sur le fait qu'il y a comparaisons répétées. Proposition :

"Au delà des limites du mode d'interaction, qui aurait pu être amélioré, l'approche comparative nécessaire au paramétrage, à l'évaluation et à l'analyse des différentes versions du modèle apportait un autre problème. Si l'utilisation d'un outil de présentation dynamique tel que SimVADB était plus adapté que la multiplication de rapports linéaires et statiques, il restait insuffisant pour mener à bien de multiples comparaisons. La succession d'évaluation visuelle reste très restreinte pour appréhender l'étendue des changements etc." (la partie sur "ok pour restitution etc." me semble inutile)

***lue***

*2019-11-13 22:31:05*

--------------------------------------------

ainsi/donc ?

***lue***

*2019-11-13 22:31:22*

--------------------------------------------

"nous avons décidé"

***lue***

*2019-11-13 22:32:42*

--------------------------------------------

Cette seconde approche a consisté à caractériser les simulations par...

cf. chap

le chapitre

tableau versions et sous-version, chap

Chapitre 5

5.2. EXPLORER LES SORTIES

trinsèque, c’est-à-dire la combinaison de valeurs de paramètres qui les rendent

uniques. Ce faisant, au sein de la plate-forme d’exploration SimVADB, l’enjeu devenait plutôt la compréhension des effets des valeurs de paramètres sur les indicateurs que l’évaluation d’une simulation en particulier. Il fallait passer du descriptif, quelle qu’en soit la méthode, à du comparatif.

Du point de vue de l’interface utilisateur, cela implique que la sélection ne se

fasse plus par un unique critère (le nom de la simulation), mais au contraire sur du multi-critère. Par une succession de sélections, chaque paramètre pou- vait constituer un nouveau filtre dans lequel on avait à choisir les valeurs à interroger (voir la figure 5.6 **(E)** de l’encadré 5.1).

La quantité de paramètres en entrée était importante et pouvait dès lors donner

lieu à un mode de sélection complexe et fastidieux – définir une par une les valeurs voulues pour chacun des 45 paramètres –. Nous avons choisi encore une fois de nous appuyer sur l’aspect visuel afin de permettre aux utilisateurs de SimVADB de choisir la ou les expérimentations à analyser.

**Visualiser avec des coordonnées parallèles.** Pour cela, on a choisi de

représenter les combinaisons de paramètres dans un graphique en « coordon- nées parallèles » (*parallel coordinates*, d’après Inselberg et Dimsdale 1987, voir Few 2006b par exemple pour une description plus succincte, illustrée et pratique). Ce type de graphique est en effet extrêmement pertinent pour re- présenter une information multi-dimensionnelle en ce qu’il permet de détecter graphiquement des *clusters* d’individus statistiques 24 (Heinrich et Weiskopf 2013, p. 2), c’est-à-dire de faire ressortir visuellement les expériences dont les valeurs de paramètre sont proches. Notons bien que l’on parle ici des valeurs de paramètres, c’est-à-dire des conditions des expériences, et non des indica- teurs de sortie. L’approche va ainsi des paramètres aux résultats : depuis des valeurs de paramètres choisies, on analyse la diversité des résultats.

**Interagir avec des coordonnées parallèles.** De plus, en matière d’in-

teraction, on utilise fréquemment les graphiques en coordonnées parallèles en vue de filtrage. Cette opération est le plus souvent menée par des actions de *brushing* (« brossage »), c’est-à-dire de sélection graphique d’une zone en dessi- nant son étendue à la souris (voir encadré 5.1). Ce type de sélection se révèle en effet souvent plus efficace et intuitive qu’une sélection textuelle ~~p lus systé-~~

~~matique~~ :

« Filtering is an operation that removes signals from its input. A filter reduces the number of lines to be rendered. In this sense, dy- namic querying [...] is a filter, if implemented with brushing [...], which reduces clutter by putting the filtered lines in focus using

24. Ici, chaque expérience est un individu statistique. Il est caractérisé par un ensemble de variables, les différents paramètres, et les modalités de ces variables ~~q ue cet individu em-~~  ~~prunte (l~~es valeurs de paramètres). Quand le nombre d’individus est important, les différentes

« courbes », qui correspondent au profil des individus sur le graphique en coordonnées paral- lèles, peuvent se superposer et montrer des tendances similaires. Avec ce type de représenta- tion, il est facile de visualiser les grandes classes d’individus constituées par ces « tendances », et donc de constater des distinctions entre les individus (les expériences) de manière visuelle.

28

***lue***

*2019-11-13 22:38:03*

--------------------------------------------

"En d'autres termes" ?

***lue***

*2019-11-13 22:38:36*

--------------------------------------------

présent ? passé ? à revérifier quand tu auras tout repris

***lue***

*2019-11-13 22:38:55*

--------------------------------------------

"par des critères multiples"

***lue***

*2019-11-13 22:40:24*

--------------------------------------------

je trouve ça bizarre d'appeler une sous-partie de l'encadré alors que lui-même n'a pas encore été appelé - est-ce nécessaire ici ? est-ce que tu ne peux pas juste appeler l'encadrer ?

***lue***

*2019-11-13 22:41:41*

--------------------------------------------

pas de saut de paragraphe OU mettre ces deux lignes en dessous après le titre "Visualiser avec..."

***lue***

*2019-11-13 22:43:23*

--------------------------------------------

appelle l'encadré direct ici (ou après la phrase suivante), On veut absolument avoir une image dès que tu en parles :p !

***lue***

*2019-11-13 22:43:50*

--------------------------------------------

coooool

***lue***

*2019-11-13 22:44:04*

--------------------------------------------

Ce dernier

***lue***

*2019-11-13 22:44:22*

--------------------------------------------

de l'expérience,

***lue***

*2019-11-13 22:45:11*

--------------------------------------------

,

***lue***

*2019-11-13 22:46:45*

--------------------------------------------

peut-être mettre ici le (des experiences) puis remplacer partout individus par expériences

***lue***

*2019-11-13 22:48:31*

--------------------------------------------

si tu veux garder, remplacer par "qui est par nature plus systématique"

Chapitre 5

5.2. EXPLORER LES SORTIES

some highlighting mechanism. Combining simple brushes using lo-

gical operators [...] further allows the user to formulate rather com- plex queries that might even achieve faster and more accurate re- sults using parallel coordinates than using a Structured Query Lan- guage (SQL) [...]. »

Heinrich et Weiskopf 2013, p. 13

Cette utilisation est courante dans les champs de la visualisation d’informa-

tion (*InfoVis* et *Visual Analytics*), et ~~e n~~ a été reprise en géographie quantitative. On la retrouve par exemple chez l’un des représentants de l’analyse spatiale des années 1990, Stewart Fotheringham. Cet auteur indique même l’usage du graphique en coordonnées parallèles en tant que filtre pour identifier des infor- mations dans une autre dimension, spatiale ici : « the data being displayed in parallel coordinates can be linked to a map and then brushed to highlight the locations of interesting lines displayed in *m*-space on the parallel co-ordinates. » (Fotheringham 1999).

*Encadré 5.1 : Construction et utilisation interactive d’un graphique en coor-*

*données parallèles*

La figure 5.6 illustre les étapes successives de construction d’un gra- phique en coordonnées parallèles, depuis le tableau statistique (**A**) jus- qu’au graphique final (**D**).

Pour cela, on projette les valeurs des variables sur des axes représentant chacune des variables (**B**). En normalisant la taille de ces axes et en les plaçant en parallèle (**C**), on peut alors tracer les « profils » des variables en reliant les positions de chacun des individus statistiques sur chacun des axes (**D**).

La seconde partie de la figure représente le mode d’interaction par *bru- shing* : on « brosse » sur chaque axe une sélection de valeurs à conserver (**E**). La sélection graphique est convertie en intervalles numériques et formalisée sous une forme classique (**F**) qui permet de filtrer les données sous-jacentes. Au final, cette opération renvoie le seul individu statis- tique répondant aux deux sélections graphiques (**G**).

29

***lue***

*2019-11-13 22:49:56*

--------------------------------------------

"ici spatiale" ou "spatiale, ici"

***lue***

*2019-11-13 22:50:33*

--------------------------------------------

voulu ?

***lue***

*2019-11-13 22:51:24*

--------------------------------------------

",dans cet exemple,"

Chapitre 5

5.2.

EXPLORER LES SORTIES

**(A)**

i4 i1

i3 i2

**(B)**

i4

i3

i2

i1

i1

i2

i3

i4

1

2

3

6

0

1

**Var2**

2

2,5

A

B

**Var1**

**Var3**

**(C)** 1

2

i1

i4

i1 i4

**(D)**

0

A

1

0

A

i2

2

i3

i3

1

3

3

1

i2

2

2

i2 i3

i4

i1

6

2,5

B

6

2,5

B

**Var1**

**Var2**

**Var3**

**Var1**

**Var2**

**Var3**

**(E)** 1

2

1

3

2

6

**Var1**

**Var2**

**Var3**

**(F)**

**AND**

**(G)**

Figure 5.6 – Construction d’un graphique en coordonnées parallèles et

sélection interactive.

Appliqué aux données de SimFeodal, cette interface (figure 5.7) se révèle par-

ticulièrement efficace pour sélectionner les configurations de paramètres à ex- plorer. Ainsi, en « brossant » quelques filtres manuellement (figure 5.7 - **A**), on arrive rapidement à isoler une expérience spécifique.

30

***lue***

*2019-11-13 22:52:24*

--------------------------------------------

vu que ton encadré est de toutes façons sur deux pages, si tu peux introduire plus d'espace entre les différentes lignes de la figure je pense que ça serait un petit peu plus agréable. Mais sinon c'est très clair !

***lue***

*2019-11-13 22:53:07*

--------------------------------------------

"mode de visualisation" ?

**Var1**

**Var2**

**Var3**

i2

2

2

B

**Var3 == "B"**

**1** ⩽ **Var1** ⩽ **2**

0

A

2,5

B

**Var1**

**Var2**

**Var3**

i1

1

2,5

A

i2

2

2

B

i3

3

1

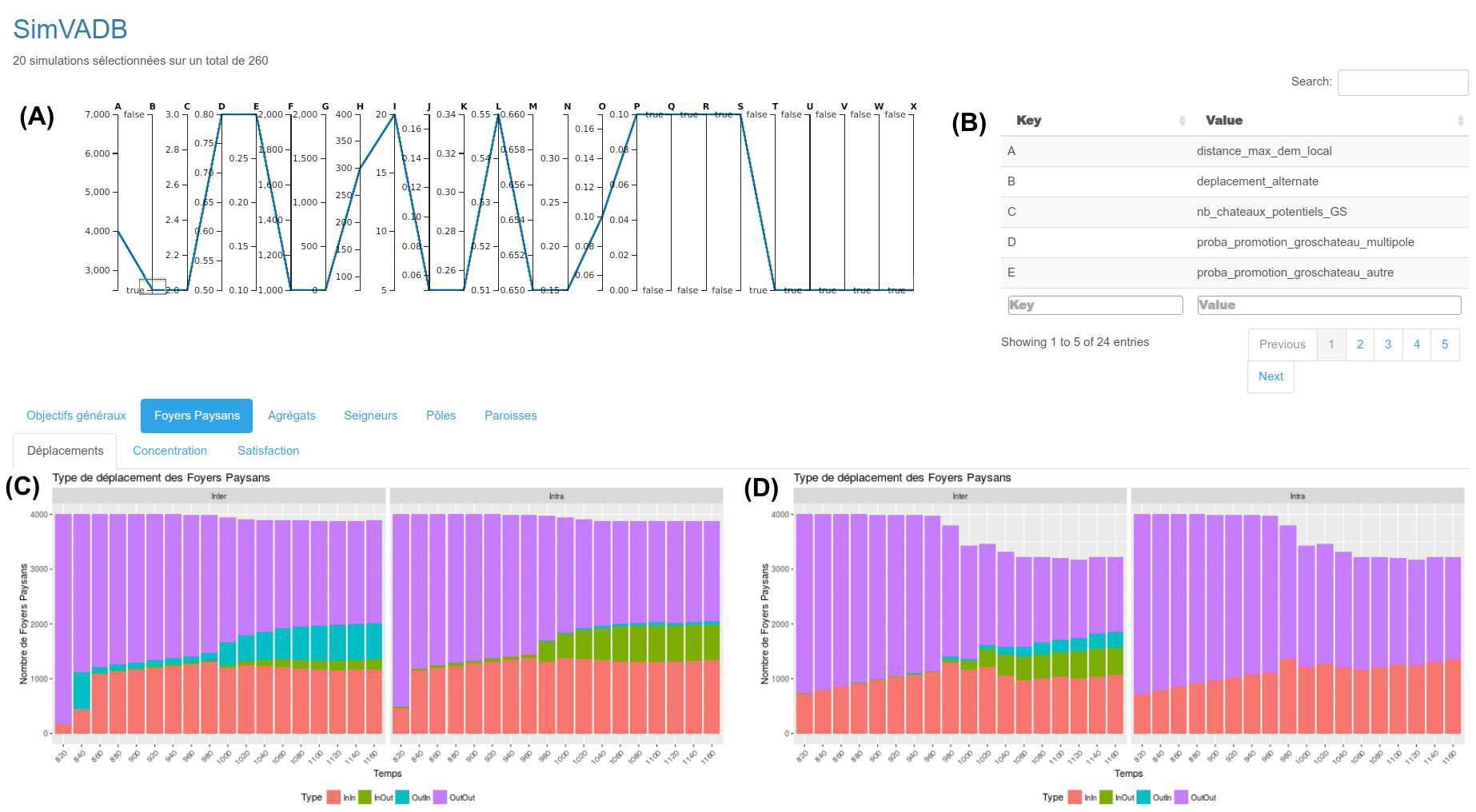
B

i4

6

0

A



Chapitre 5

5.2. EXPLORER LES SORTIES

Figure 5.7 – SimVADB (Simulation Visual Analysis DashBoard), un *dashboard*

d’exploration visuelle des indicateurs de sortie de simulation de SimFeodal. La sélection des simulations à explorer se fait dans le graphique en coordonnées parallèles (**A**), en « brossant » des filtres graphiques sur les « dimensions » du graphique, dimensions dont les intitulés sont explicités dans le tableau **B**. Les graphiques **C** et **D** indiquent l’évolution des types de déplacements des foyers paysans au cours des simulations.

- Le graphique **C** représente, pour cet indicateur, une moyenne de l’ensemble des simulations intégrées dans la base de données (260 ici), recouvrant donc plusieurs valeurs de paramètres.

- Le graphique **D** représente cet indicateur calculé depuis un sous-ensemble de 20 simulations (donc une expérience composée de 20 réplications), pour lesquelles le paramètre « B » (deplacement\_alternate) vaut true.

Afin de permettre aux utilisateurs de remarquer les particularités des simula- tions explorées, nous avons choisi de mettre en emphase les différences entre la tendance générale des indicateurs, en calculant des moyennes de l’ensemble des simulations (figure 5.7 - **C**), et les valeurs spécifiques des indicateurs de l’expérience choisie (figure 5.7 - **D**). Cela permet, visuellement, d’être en me- sure d’évaluer les sorties de simulation d’une expérience tout en ayant un ré- férentiel visible. Les différentes expériences produisent des résultats sensible- ment similaires 25, et on ne peut alors plus les comprendre sans les confronter à d’autres résultats similaires. Le choix d’une agrégation de l’ensemble des simu- lations effectuées est discutable, en ce qu’on aurait par exemple pu plutôt iso- ler des simulations « de référence » afin de diminuer l’effet « d’aplatissement » engendré par l’agrégation de résultats nombreux et hétérogènes. Toutefois, la variabilité des résultats étant encore assez restreinte, au moment de la création et de l’utilisation de SimVADB, ce référentiel agrégé permettait déjà une com- préhension plus fine des sorties de simulations, en particulier dans l’analyse de l’impact de variations fines de valeurs de paramètres.

25. Si chaque expérience, et chaque réplication, produisent des résultats uniques, le choix d’une évaluation par des indicateurs visuels peut prêter à confusion si l’on n’a pas de repère

précis. Les critères attendus, présentés dans

sont ainsi assez précis pour départager

une simulation très éloignée des attentes et une autre simulation plus conforme. Pour autant, par exemple quand les valeurs de paramètres varient faiblement, les résultats produits peuvent être assez similaires dans les grandes tendances qu’ils font ressortir.

31

***lue***

*2019-11-13 22:53:46*

--------------------------------------------

on veut la figure en plus grand !!!

***lue***

*2019-11-13 22:54:24*

--------------------------------------------

préciser qu'il s'agit de la 2e version ? pour bien distinguer de la première version illustrée plus haut...

***lue***

*2019-11-13 22:58:53*

--------------------------------------------

définitivement il faut que la figure soit plus grande, car là on ne peut rien lire des dimensions...

donc soit mettre en paysage pour garder le format en longueur, soit faire 3 lignes (A/B/C+D) ou 4 lignes (une pour C et une pour D)

+ Je trouve le titre trop générique, proposition :

"Utilisation des graphiques en coordonnées parallèles dans SimVADB (pas besoin de la parenthèse tu l'as déjà explicité dans le texte) pour explorer visuellement les indicateurs etc."

+ je trouverai ça plus lisible si tu avais une différences entre le titre de la figure et son commentaire en "légende"

***lue***

*2019-11-13 23:00:42*

--------------------------------------------

plutôt préciser qu'il s'agit des valeurs correspondant à une sélection sur le paramètre B = pas besoin d'avoir le détail, ça floute le message

***lue***

*2019-11-13 23:02:11*

--------------------------------------------

précise ? 'pour éviter la répétition

le chapitre 3

Chapitre 5

5.2. EXPLORER LES SORTIES

**5.2.6 Explorer en comparant : la plateforme SimEDB** 26

Après un travail de paramétrage grossier qui permet de stabiliser les méca- nismes, il est souvent nécessaire de passer à une phase plus fine. On vise à ce moment à mieux calibrer un modèle à l’aide de variations de valeurs de para- mètres de granularité inférieures. En vue d’évaluer les simulations, et donc de les différencier les unes des autres à l’aune des indicateurs générés, la compa- raison d’une expérience spécifique avec un référentiel constitué de toutes les expériences précédentes ne permet plus de mener ce travail de comparaison fine. Les variations entre simulations sont trop fines pour être distinguables les unes des autres par le biais d’une comparaison avec un référentiel unique 27. Cela s’entend quelque soit la manière dont ce référentiel est constitué, qu’il résulte d’une agrégation de simulations ou encore d’une version « de base » du modèle (par exemple, dans le cas de SimFeodal, les versions principales

identifiées dans le

4).

Pour pouvoir correctement évaluer les apports d’un nouveau jeu de valeurs de

paramètres, et donc, dans une démarche itérative, pouvoir différencier deux expériences successives, il est nécessaire d’être en mesure de comparer direc- tement les expériences les unes avec les autres, ou encore avec un référentiel facilement ajustable. On peut énumérer ~~l es~~ quelques exemples de cas de figure

~~suivants~~ :

—

Comparaison entre une expérience spécifique et une autre expérience

spécifique de même « importance ». Par exemple, comparer deux expé- riences qui font varier légèrement différement une valeur de paramètre. Comparaison entre une expérience spécifique et une autre expérience spécifique d’« importance » différente. Cette différence de niveau peut

—

~~être constituée~~ par exemple ~~p ar~~ une version « de base » que l’on compa-

re~~rait~~ à une variante de celle-ci.

Comparaison entre deux expérimentations, par l’agrégation de leurs ré- sultats. Si l’on a mené des expérimentations faisant varier de manière systématique deux paramètres différents, il peut être intéressant de les comparer en bloc, c’est-à-dire par exemple en prenant les moyennes de chacune des expériences composant ces expérimentations.

—

Ce faisant, on fait en fait varier la notion de « simulation de référence », qui

peut alors revêtir plusieurs formes. Pour cela, il n’est plus possible de mener

26. La plate-forme d’exploration SimVADB (SimFeodal Visual Analysis Dashboard) a été renommée SimEDB ([...] Exploration Dashboard) par soucis de simplicité, le terme « Explora- tion » nous semblant plus ~~c ompréhensible et~~ explicite que celui de Visual Analysis. Ce nom apporte de plus une cohérence sémantique entre plusieurs productions de l’auteur – Time- LineEDB - (Cura 2017b) ; RoadTrafficEDB et CitiBikeEDB - (Cura 2017a). Cela inscrit cette plate-forme d’exploration de données issues de simulation dans une « famille » d’outils d’ex- ploration de données spatio-temporelles.

27. ~~O n peut prendre l’exemple de SimFeodal.~~ En faisant varier le nombre de foyers paysans de 4000 à 4200 (5% de variation), les résultats du modèle changent peu : de faibles variations de valeurs de paramètres entraînent le plus souvent de faibles variations dans les indicateurs de sortie~~s~~ observés. On peut de plus noter que les répercussions d’un changement de valeur de paramètre peuvent être très différentes de l’ordre de grandeur de ce changement de valeur. Cela s’explique par la non-linéarité de l’influence des paramètres. Dans l’exemple pris, pour ces 5% de variation dans le nombre de foyers paysans, la majorité des indicateurs variera ainsi de moins de 1%.

32

***Lucigares du pharaon***

*2019-11-14 17:50:18*

--------------------------------------------

format et place des tirets dans cette phrase ?

***Lucigares du pharaon***

*2019-11-14 17:52:31*

--------------------------------------------

", fondé sur de grandes variations des valeurs," (pour mieux comprendre par rapport à quoi la granularité est inférieure)

***Lucigares du pharaon***

*2019-11-14 17:53:21*

--------------------------------------------

"plus poussé/plus précis" ?

***Lucigares du pharaon***

*2019-11-14 17:53:54*

--------------------------------------------

"Dans SimFeodal, en faisant..."

***Lucigares du pharaon***

*2019-11-14 18:18:57*

--------------------------------------------

"du fait de" ou "ce qui s'explique" (pour faire une seule phrase)

***Lucigares du pharaon***

*2019-11-14 18:20:46*

--------------------------------------------

A chaque fois je mettrai des articles "La comparaison"

***Lucigares du pharaon***

*2019-11-14 18:25:36*

--------------------------------------------

,

***Lucigares du pharaon***

*2019-11-14 18:25:45*

--------------------------------------------

, (sinon phrase d'après nominale)

***Lucigares du pharaon***

*2019-11-14 18:26:11*

--------------------------------------------

"en comparant" ?

***Lucigares du pharaon***

*2019-11-15 13:22:31*

--------------------------------------------

"Il s'agit en fait de" ?

chapitre

Chapitre 5

5.2. EXPLORER LES SORTIES

une comparaison visuelle entre un référentiel commun et une expérience spéci-

fique, mais bien de baser l’évaluation sur la comparaison entre deux ensembles spécifiques qui doivent pouvoir être spécifiés. D’un point de vue méthodolo- gique, cela requiert de pouvoir afficher conjointement les indicateurs de sor- ties de deux expériences (ou ensembles d’expériences). Cela implique aussi de laisser à l’utilisateur la responsabilité d’un choix supplémentaire puisqu’il faut désormais effectuer deux sélections : une pour chacun des points de comparai- son. La sélection d’une expérience via l’usage de *brushing* sur un graphique en coordonnées parallèles des valeurs de paramètres ayant montré son efficacité, il a été choisi d’étendre ce principe d’interactivité au choix du référentiel.

Dans cette version remaniée de la plate-forme d’exploration (voir figure 5.8),

renommée SimEDB (**Sim**Feodal **E**xploration **D**ash**B**oard) 28, l’accent est mis sur la comparaison de deux ensembles de résultats, chacun répondant à une sélec- tion propre. L’utilisateur doit ainsi « paramétrer » interactivement, via *brushing*, les expériences à afficher pour le référentiel et pour les expériences à compa- rer. On dispose pour cela de deux outils de filtrage des simulations (partie de gauche dans la figure 5.8), qui peuvent être utilisés de concert, pour compa- raison visuelle, ou par étapes successives 29.

En superposant les graphiques et tableaux des indicateurs, la comparaison vi- suelle est facilitée. On peut alors comparer deux variations fines d’un méca- nisme du modèle, en sélectionnant par exemple une unique différence dans les valeurs de paramètres du modèle (par exemple un paramètre relatif à la pro- motion des paroisses dans la figure 5.8). De manière générale, ce choix d’outil d’interrogation des données permet de répondre à l’ensemble des cas de figures identifiés dans les paragraphes précédents.

28. Voir la note de bas de page 26.

29. En menant par exemple une première comparaison entre une expérience « A » en haut et « B » en bas, puis en sélectionnant « C » en haut, puis « D » en bas etc. On compare ainsi A avec B, puis B avec C, et enfin C avec D.

33

***Lucigares du pharaon***

*2019-11-15 14:04:01*

--------------------------------------------

je mettrai ça en première conséquence

***Lucigares du pharaon***

*2019-11-15 14:15:16*

--------------------------------------------

supprimer cette note OU déplacer la note 25 ici OU mettre une petite phrase en dessous du titre "Une nouvelle version blabla renommée" et mettre la note là

***Lucigares du pharaon***

*2019-11-15 14:20:43*

--------------------------------------------

"donc" ?

***Lucigares du pharaon***

*2019-11-15 14:20:56*

--------------------------------------------

"peut"

***Lucigares du pharaon***

*2019-11-15 14:21:32*

--------------------------------------------

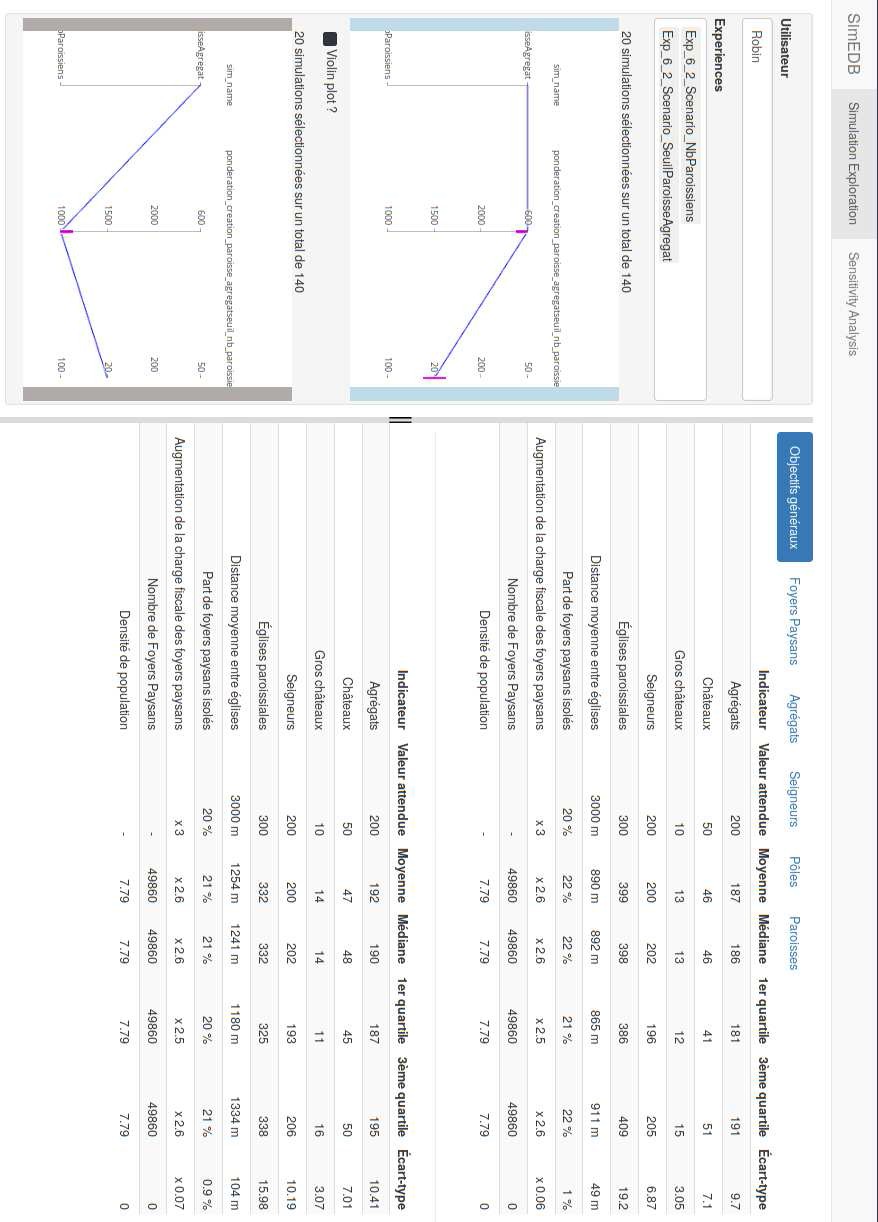
"la comparaison"

***Lucigares du pharaon***

*2019-11-15 14:25:20*

--------------------------------------------

mettre à la fin de la phrase



Chapitre 5

5.2.

EXPLORER LES SORTIES

Figure 5.8 – SimEDB

Nous reviendrons plus précisément et longuement sur la description de Si-

mEDB dans les parties suivantes (section 5.4, p. 57), mais après en avoir décrit les étapes de construction et les besoins auxquelles ces évolutions répondaient, il est maintenant nécessaire de revenir sur les données manipulées par cette plate-forme d’exploration. Le type, la structure et la masse de ces données (sec- tion 5.1) sont en effet indissociables des choix méthodologiques effectués pour SimEDB. Il est donc important de présenter les choix et contraintes de ces don- nées avant d’entrer dans une description approfondie de la plate-forme.

34

***Lucigares du pharaon***

*2019-11-15 14:26:55*

--------------------------------------------

mettre un titre plus explicite : "La comparaison d'expériences dans SimEDB"

***Lucigares du pharaon***

*2019-11-15 14:28:00*

--------------------------------------------

nécessaire

***Lucigares du pharaon***

*2019-11-15 14:28:46*

--------------------------------------------

"nous proposons maintenant" ?

Chapitre 5

5.3. ORGANISER LES DONNÉES

**5.3**

**Organiser les données**

On a vu dans la première partie de ce chapitre (section 5.1) que les données

produites par SimFeodal étaient nombreuses, diverses et massives. La seconde partie (section 5.2) a montré les types de problèmes que de telles données, une fois exploitées pour en tirer les indicateurs de sortie, peuvent poser en matière d’exploration.

Nous souhaitons ici revenir sur une corollaire indispensable à l’utilisation effi-

cace de ces données. De la même manière que la multiplicité des indicateurs, des expériences et des expérimentations requiert des outils d’exploration adap- tés, la multiplicité des données requiert des outils de stockage et d’interroga- tion eux aussi adaptés. Là encore, on peut noter une succession de contraintes liées à ces données et à leur massification, contraintes qui limitent et guident les modes d’organisation de ces données. Sans structuration adéquate, l’acqui- sition, l’archivage, l’interrogation ou encore la sauvegarde des données géné- rées par le modèle ne peuvent être garantis, et encore moins de manière effi- cace. Le choix d’une méthode d’organisation des données en sortie de simula- tions ne relève donc pas d’une quelconque coquetterie technique. Au contraire, conditionne et contraint fortement aussi bien les modalités de création des in- dicateurs que les possibilités de la plate-forme d’exploration de les afficher de manière interactive.

**5.3.1**

**Assurer la capacité d’interrogation des données**

Avant de se soucier du « schéma » de la base de données 30, du choix du Système de Gestion de Base de Données (SGBD), ou encore des performances de ce dernier, il convient de se fixer sur la manière dont on souhaite entreposer les données.

De la myriade de fichiers issus de tableurs organisés dans une multitude de

dossiers spécifiques à l’entrepôt de données décentralisé orienté documents, en passant par les traditionnelles bases de données relationnelles, les possibi- lités de stockage et d’organisation des données issues de simulations sont ainsi innombrables. Plusieurs contraintes successives permettent de limiter le choix à un sous-ensemble de solution adaptées parmi lesquelles on peut alors mener une étude plus approfondie, passant notamment par des comparaisons entre ces candidats.

Une des premières contraintes est constituée par la nécessité d’interroger fré-

quemment et de manière répétée les données. C’est l’une des contre-parties du passage d’un rapport automatique à un outil d’exploration interactif et dyna- mique (section 5.2.4). Dans un rapport automatique, on interroge une fois les données pour en tirer les indicateurs de sortie, et ceux-ci ne sont plus ame- nés à changer, sauf re-calcul par exemple suite à des ajouts d’indicateurs. Au

30. On utilise souvent le terme de « schéma » pour désigner la version implémentée, dans un SGBD spécifique, du Modèle Conceptuel de Données (MCD). Contrairement au MCD, qui donne une version conceptuelle et générique d’une base de données, le schéma est donc tributaire du SGBD dans lequel il est intégré.

35

***Lucigares du pharaon***

*2019-11-15 14:33:36*

--------------------------------------------

"cette méthode"

***Lucigares du pharaon***

*2019-11-15 14:34:59*

--------------------------------------------

"d'un affinement technique superflu" ? la "coquetterie" fait bizarre

***Lucigares du pharaon***

*2019-11-15 14:37:50*

--------------------------------------------

"que l'on peut alors comparer de manière approfondie" - mais la précision me semble limite pas utile (ça me dérange moins si, comme dans ma proposition, la phrase est + courte)

Chapitre 5

5.3. ORGANISER LES DONNÉES

contraire, dans un outil dynamique et interactif, les indicateurs sont calculés

depuis les données « à la volée », c’est-à-dire à chaque qu’un indicateur doit être affiché. On peut mettre en place un système de cache, pour conserver les cal- culs déjà effectués, mais avec les dernières itérations de SimEDB où le nombre de possibilités de sélections est extrêmement important, ce n’est plus possible. Il est nécessaire de procéder aux calculs à chaque affichage des graphiques et tableaux de résultats. L’interrogation des données est donc extrêmement fréquente et répétitive. Elle doit alors être aussi simple (en termes de mode d’interrogation) qu’efficace (en termes de rapidité d’interrogation).

Dans le cadre des données issues de SimFeodal, en vue de leur mobilisation

dans SimEDB, nous avons ainsi eu à sélectionner quelques SGBD candidats parmi une foule de solution possibles. Afin de guider ce choix, trois critères ont été définie~~s, e t forment, selon une hiérarchie propre à leur ordre, un en-~~

~~semble de filtre ayant permis la réduction des SGBD possibles à un nombre~~

~~appréhendable.~~ Ces critères, dont l’énonciation guidera cette partie, sont ~~a insi~~

(1) l’universalité, ou « agnosticité », des SGBD aux outils de requête ; (2) la pé- rennité et stabilité des solutions disponibles et (3) les performances des SGBD considérés.

**5.3.1.1 Interroger de manière universelle et indépendante**

Lors de la conception d’un outil faisant appel à des données, qui plus est mas-

sives, il convient de se positionner tôt sur la manière d’interroger ces données. Par interrogation, on entend ici, comme souvent dans le domaine des bases de données, la manière de faire appel, concrètement, aux données, pour en tirer les sous-ensembles, agrégations et autres résultats synthétiques résultant du traitement des données brutes. Si l’on considère des données stockées dans un tableur, alors les « formules », les tableaux croisés dynamiques ou encore les graphiques issus du tableur sont des interrogations des données, qui s’ex- priment dans ce cas via un ensemble de langages, écrits – les formules, qui font appel à des fonctions spécifiques des tableurs – ou visuels – les tableaux croisés dynamiques, construits en faisant glisser des intitulés de colonnes dans un tableau.

**Stockage distribué ou centralisé.** Avant même de s’intéresser aux spé-

cificités de ces langages, un premier choix réside dans le mode de stockage des données qui doivent être mis à disposition d’une plate-forme. Doit-on lais- ser l’utilisateur intégrer lui-même les données, et ainsi, en faire un stockage

« distribué », dans le sens où chaque utilisateur de l’application possèderait physiquement une copie locale des données ? Ou, au contraire, les données doivent-elles être centralisées, c’est-à-dire enregistrées en une seule copie à la- quelle les utilisateurs accèderaient à distance ? Pour reprendre l’exemple des tableurs, doit-on privilégier une solution locale – chacun ayant une copie du fichier tableur et menant ses propres modifications dessus – ou une approche de type centralisée, par exemple en privilégiant des tableurs collaboratifs en ligne (*Google Docs/Sheets* par exemple) ?

Habituellement, c’est-à-dire dans une grande majorité d’applications, les don- 36

***Lucigares du pharaon***

*2019-11-15 14:38:27*

--------------------------------------------

fois

***Lucigares du pharaon***

*2019-11-15 14:49:27*

--------------------------------------------

"cette solution" ?

***Lucigares du pharaon***

*2019-11-15 14:49:42*

--------------------------------------------

"donc"

***Lucigares du pharaon***

*2019-11-15 14:51:09*

--------------------------------------------

pas utile je trouve

***Lucigares du pharaon***

*2019-11-15 14:53:00*

--------------------------------------------

"mises"

Chapitre 5

5.3. ORGANISER LES DONNÉES

nées sont stockées localement : cela permet, en particulier, de ne pas dépendre

d’une connexion internet pour interroger des données qui seraient hébergées sur internet. Dans le cas de SimFeodal, cette solution est rendue difficile, sinon impossible, par la masse de données en sortie de simulations. Si chaque utili- sateur de SimEDB devait posséder une copie des données, voir plusieurs en cas d’utilisations depuis différents ordinateurs, cela occuperait plusieurs gigaoctets de données à chaque fois. De plus, en cas de mise à jour des données, c’est- à-dire d’insertions de nouvelles sorties de simulations, il faudrait distribuer à nouveau l’ensemble du jeu de données, à chaque fois.

Pour ces raisons, nous avons fait le choix d’un stockage centralisé, sous forme

d’une architecture « client-serveur », hébergé sur un serveur internet dédié, ce qui permet à la plate-forme de travailler à chaque fois sur les données les plus à jour, réduit la taille du stockage physique associé, et dispense d’une configu- ration sur chaque poste utilisateur : si le lien entre l’application et les données fonctionne correctement pour un utilisateur, il fonctionnera à l’identique pour tous les autres. Ce choix présente un dernier avantage, non négligeable : en stockant les données en un seul lieu, c’est-à-dire sur un serveur informatique, on peut faire en sorte de rendre ce serveur aussi performant que possible, et accélérer ainsi l’interrogation des données pour tous les utilisateurs.

**Interrogation spécifique ou générique.** De nombreuses solutions inté-

grées de gestion de données proposent leurs propres modes d’interaction avec les données, c’est-à-dire un langage spécifique permettant d’interroger les don- nées contenues dans le système 31. Au contraire, les SGBD les plus classiques s’appuient plutôt sur des langages de requêtes aussi standardisés que possibles, afin de faciliter l’adoption de leur propre solution à des utilisateurs d’autres plate-formes. La spécificité présente l’avantage de langages plus adaptés aux données manipulées, et donc souvent plus intuitifs dans l’interrogation des spé- cificités des données. De plus, la spécificité permet aussi une optimisation des requêtes, et est donc souvent plus performante que les solutions plus géné- riques.

De manière générale et dans le cas de SimFeodal, nous avons préféré privi-

légier une approche plus générique, faisant appel à des solutions de SGBD plus standardisées. La raison tient principalement à une volonté de généricité du stockage des données : au cours des différentes étapes de construction de SimFeodal, les besoins en matières d’interrogation des données ont évolué. Cette évolution était prévisible et prévue, et nous avons donc choisi dès le départ d’adopter uniquement des solutions modulaires, garantissant une évo- lutivité facilitée de la base de données, aussi bien regardant sa structure (enre- gistrement de nouvelles variables ou de nouveaux agents du modèle) que son contenu (massification des données en sortie au fur et à mesure de l’exploration du modèle).

De plus, dans la perspective de ce travail de thèse, où l’on cherche à rendre

31. Souvent, cette interrogation se fait par appel à des *API* (*Application Programming Interface*, ou Interface de Programmation Applicative en français). Ces interfaces sont propres à chacune des plate-formes, et demandent donc un langage et un formalisme de requête spécifique.

37

***Lucigares du pharaon***

*2019-11-15 14:54:51*

--------------------------------------------

mettre plutôt après "il faudrait" (ligne du dessus)

***Lucigares du pharaon***

*2019-11-15 14:55:37*

--------------------------------------------

tiret ? + je dirai "plateforme d'exploration" et "les données issues des simulations" pour bien expliciter

***Lucigares du pharaon***

*2019-11-15 14:56:09*

--------------------------------------------

". Si"

***Lucigares du pharaon***

*2019-11-15 15:00:02*

--------------------------------------------

répétition. "particularité" ?

Chapitre 5

5.3. ORGANISER LES DONNÉES

les productions aussi reproductibles et génériques que possible, il était indis-

pensable de disposer d’un SGBD aussi standard que possible pour en faciliter l’adoption et l’adaptation à d’autres modèles de simulations par exemple.

**Bases de données relationnelles ou NoSQL.** Même une fois arrêté sur

le choix de ne faire appel qu’à des outils standards pour stocker les données, le nombre de solutions disponibles demeure très important. Afin de réduire ce nombre, on peut déjà choisir les grands types de SGBD auxquels faire appel. Les SGBD sont ~~ainsi~~ souvent classés selon les grands traits de la méthode dont ils or- ganisent les données. Les deux grands types 32 sont les SGBD « relationnels » et les SGBD « NoSQL ». Si la distinction est sujette à de très nombreux débats, sou- vent virulents 33 , on se contentera ici de définir les SGBD relationnels comme les SGBD~~, les plus fréquemment utilisés,~~ où l’information est stockée dans des tables composées de champs – les colonnes, correspondant aux variables – et de lignes – les entités décrites par les variables. Le format des données est donc rectangulaire et n’accepte pas, comme dans un tableau statistique, que les entités possèdent un nombre de variables différent, ou encore des types de valeurs différents de celles des autres entités (une même colonne ne peut donc contenir conjointement un nombre et un texte dans des lignes différentes). On nomme ces bases de données relationnelles via la manière qu’elles ont de faire communiquer des éléments hétérogènes, et donc contenus dans des tables dif- férentes : si les tables ont une colonne en commun, on pourra alors effectuer une opération de jointure permettant de mettre en commun les informations de ces tables dans une unique table résultante.

A l’inverse, les SGBD NoSQL se définissent de manière opposée à ce mode de

stockage 34, rompant par exemple la contrainte d’unicité de type des colonnes, ou de nombre identique de colonnes renseignées pour chaque entité. Pour sim- plifier le discours, on se contentera de caractériser les SGBD NoSQL comme des SGBD non relationnels. Les SGBD NoSQL ont, en général, de bien meilleures performances et une plus grande flexibilité que les SGBD relationnels. Dans le cas de SimEDB/SimFeodal, où l’on est confronté à des données massives, cela présente un avantage non négligeable.

Toutefois, leur flexibilité est associée à une contrainte majeure en termes de généricité : alors que les SGBD relationnels partagent un langage d’interro-

32. Il en existe d’autres, comme les SGBD orientés objets (quasiment disparus aujourd’hui), orientés graphes (Neo4j…), les SGBD pensés pour le stockage et l’interrogation d’ontologies (*Triplestores RDF*, interrogeables en langage SPARQL) ou encore les nouveaux SGBD de type

« NewSQL » (Apache Ignite, CockRoachDB…) pensés pour une parallélisation massive des don- nées. Ces types de SGBD ne correspondent toutefois pas du tout aux besoins identifiés pour SimEDB/SimFeodal, et sont en général dédiés à des problèmes et marchés de niches. Nous ne les décrirons donc pas plus en détail ici.

33. À l’instar des violentes querelles qui agitent régulièrement les informaticiens : Vim vs Emacs, Programmation Orientée-Objet vs Programmation Fonctionnelle, R vs Python…

34. À l’origine, c’était le sens fort du nom « NoSQL » : Non SQL, le SQL faisant ici références aux SGBD pré-existants, majoritairement relationnels, dont la mouvance NoSQL, portée par l’apparition des « *big data* » a voulu se distinguer. Sans entrer dans le détail, notons tout de mêmes que de nombreux SGBD NoSQL, qui traduisent désormais cet acronyme par « Not only SQL », sont maintenant relationnels, mais mettent en avant d’autres types d’approches.

38

***Lucigares du pharaon***

*2019-11-15 15:02:03*

--------------------------------------------

virgule

***Lucigares du pharaon***

*2019-11-15 15:02:46*

--------------------------------------------

"principaux grands" ?

***Lucigares du pharaon***

*2019-11-15 15:07:58*

--------------------------------------------

", qui sont par ailleurs les plus fréquemment utilisés,"

***Lucigares du pharaon***

*2019-11-15 15:08:32*

--------------------------------------------

"des"

***Lucigares du pharaon***

*2019-11-15 15:09:17*

--------------------------------------------

"par exemple,"

***Lucigares du pharaon***

*2019-11-15 15:09:39*

--------------------------------------------

du fait de

***Lucigares du pharaon***

*2019-11-15 15:13:17*

--------------------------------------------

je ne trouve ça pas très clair exprimé comme ça... proposition :

"On nomme ces SGBD "relationnels" parce qu'ils permettent de mettre en relation des tables différentes, c'est-à-dire des contenus hétérogènes, par le biais de champs identiques. On peut ainsi faire communiquer différentes tables possédant une colonne identique par le biais d'une jointure, qui permet de mettre en commun etc."

***Lucigares du pharaon***

*2019-11-15 15:14:01*

--------------------------------------------

"au langage classiquement utilisé dans les etc."

***Lucigares du pharaon***

*2019-11-15 15:14:48*

--------------------------------------------

"également" ?

Chapitre 5

5.3. ORGANISER LES DONNÉES

gation commun, le SQL (*Structured Query Language*) 35, les SGBD NoSQL font plus souvent appels à des langages spécifiques à chaque SGBD. Pour SimEDB, cela impliquerait une fort dépendance au SGBD choisi : en cas de changement de SGBD, toutes les requêtes seraient à reformuler dans le nouveau langage, parfois même selon des logiques extrêmement différentes les unes des autres (dérivés du SQL, interrogations via des objets JSON, via des langages de par- cours de graphes…) etc. Au contraire, avec les SGBD relationnels, le langage de requête étant commun, une fois le code d’interrogation généré, il est très aisé de changer de SGBD ~~c ible.~~ Cela garantie une forte capacité d’évolutivité aux outils d’interrogation de données tels que SimEDB. Puisque les fournis- seurs de données sont interchangeables, on peut en changer au fur et à mesure de l’apparition de nouveaux besoins.

En raison de la généricité de ces solutions relationnelles, qui vient s’ancrer

dans la recherche de reproductibilité et de généricité de notre démarche d’en- semble, nous avons donc choisi de faire reposer le stockage et l’interrogation des données sur des SGBD relationnels. Cette décision s’est montrée d’autant plus heureuse que, au cours de la construction et de l’évolution de SimEDB, le SGBD choisi pour héberger les données a changé plusieurs fois. La géné- ricité des outils choisis a permis de minimiser, voire quasiment d’éviter, les changements à apporter au code-source de SimEDB relatif à l’interrogation de données en vue de produire les indicateurs de sortie depuis les données brutes en sortie de SimFeodal.

**Entrepôts de données et interrogation directe.** En parallèle des SGBD,

des solutions « intermédiaires » permettent de s’abstraire des SGBD en eux- mêmes pour mener les requêtes. Ces solutions, que l’on nomme « Entrepôts de Données » (*Data Warehouses*), se comportent comme une surcouche faisant l’interface entre un ou plusieurs SGBD et la requête émise par le client final. Elles se placent ~~d onc~~ comme intermédiaire entre les SGBD mobilisés et les applications qui les interrogent. Les entrepôts de données jouent aussi bien le rôle d’agrégateur~~s~~ de données 36 que d’environnement~~s~~ de manipulation et de restructuration de données (on les nomme alors « ETL » – *Extract-Transform- Load*).

Le grand intérêt de ces outils d’interface est d’abstraire la complexité de cha-

cune des bases de données manipulées en générant une interface d’interro- gation unique et générique, souvent performante grâce à des optimisations spécifiques (pré-calcul des requêtes possibles par exemple).

35. Ce langage d’interrogation est omniprésent dans l’interaction avec les SGBD, mais aussi, avec de légères variantes, au sein de nombreux logiciels reposant des sélections de données, par exemple les logiciels SIG qui se basent sur la syntaxe du SQL (le fameux triptyque **SELECT**

… **FROM** … **WHERE** …)

36. C’est-à-dire qu’ils permettent d’agréger des sources de données composites, provenant potentiellement de différentes sources (plusieurs bases de données relationnelles) et de diffé- rents types de sources (différents SGBD, relationnels ou non par exemple).

39

***Lucigares du pharaon***

*2019-11-15 15:31:55*

--------------------------------------------

"adaptée" ? heureuse fait un peut comme si au final t'avais choisi un peu au hasard

***Lucigares du pharaon***

*2019-11-15 15:35:16*

--------------------------------------------

"la réécriture" (pour simplifier un peu la phrase) ?

***Lucigares du pharaon***

*2019-11-15 15:43:18*

--------------------------------------------

pas de saut de paragraphe

Chapitre 5

5.3. ORGANISER LES DONNÉES

**Environnements OLAP.** Dans le domaine de la visualisation interactive de

données, ces outils sont beaucoup utilisés, en particulier dans le monde de l’in- formatique décisionnelle. Ils se révèlent en effet extrêmement utiles quand les données sources ne peuvent être modifiées (par exemple quand elles sont issues de chaînes de collecte complexes, ou encore quand leur volumétrie et leur dé- bit est important), puisqu’ils permettent de constituer une surcouche rendant l’interrogation et la visualisation de ces données accessibles à des analystes non spécialistes de la manipulation de données. Toujours en informatique dé- cisionnelle, il est courant de faire appel à des entrepôts de données d’un type spécifique, les « traitements analytiques en lignes », ou environnements OLAP (***O****n****L****ine* ***A****nalytical* ***P****rocessing*), qui permettent de structurer, par exemple sous formes de cubes de données, des sources de données hétérogènes présentant de nombreuses dimensions.

**Environnements SOLAP.** Les environnements OLAP ont été utilisés, promus

et adoptés dans le champs scientifique de la géomatique, en ce qu’ils per- mettent de mettre en place rapidement des environnements d’analyse visuelle de données multi-dimensionnelles spatiales et temporelles. Dans ce cadre, où ces outils sont appelés « SOLAP » (*Spatial OLAP*), les données spatiales s’in- tègrent extrêmement bien en raison de leur capacité à s’emboîter selon les

échelles, ouvrant dès lors la voie à des analyses multi-échelles et multi-dimensionnelles complexes.

Dans la communauté géomatique francophone, les solutions SOLAP sont bien

représentées (par exemple autour de Sandro Bimonte et de son travail de visua- lisation de données spatiales environnementales, (Bimonte 2007 ; Bimonte, Tchounikine et Miquel 2005 ; Zaamoune et al. 2013), et sont couramment employées pour répondre à des questionnements méthodologiques proches de ceux développés dans cet ouvrage. En lien avec les besoins de performances identifiés plus haut, on notera que certaines solutions OLAP permettent aussi d’optimiser la vitesse d’interrogation de bases de données, et visent ainsi à ga- rantir une réponse rapide pour des outils d’interrogation de données interactifs (Zeng, Agarwal et Stoica 2016).

**Performance ou généricité.** Nous avons cependant choisi de ne pas faire

usage de ces outils pour les mêmes raisons que pour les SGBD NoSQL : les avantages qu’ils présentent ne suffisent pas à contre-balancer la perte de gé- néricité qu’ils impliquent. Pour profiter au mieux de ces environnements, il est en effet nécessaire de faire appel à un nouveau langage d’interrogation des données (le « MDX », de « *Multidimensional Expressions* »). Les différentes solu- tions OLAP/SOLAP, de plus, présentent les mêmes inconvénients que les SGBD NoSQL : chacune interagit de manière propre aux différents SGBD, et ces outils sont donc difficilement interchangeables.

De la même manière, on se restreindra, parmi les SGBD relationnels interro-

geables en SQL, à ceux qui disposent d’une méthode d’interrogation standard : si tous ces SGBD acceptent le SQL, certains demandent par exemple des pro- 40

***Lucigares du pharaon***

*2019-11-15 15:46:25*

--------------------------------------------

les data warehouses ?

***Lucigares du pharaon***

*2019-11-15 15:48:45*

--------------------------------------------

c'est un peu trop découpé... remonter ce paragraphe avec le précédent et titrer "les environnements olap et solap"

***Lucigares du pharaon***

*2019-11-15 15:49:28*

--------------------------------------------

"notre travail / ce travail"

Chapitre 5

5.3. ORGANISER LES DONNÉES

tocoles spécifiques pour recevoir établir la connexion au SGBD, recevoir la re-

quête et renvoyer les données correspondantes. Pour ce même choix « d’agnos- ticité » de la plate-forme d’interrogation face à la solution de stockage choisie, on ne conservera que les SGBD acceptant les connexion standardisées (ODBC et JDBC).

**SGBD et données spatiales.** On a mentionné le fait que les entrepôts

de données étaient fortement utilisés, en particulier dans la communauté géo- matique, car très appropriés aux données spatiales. De prime abord, ce point peut paraître critique : jusque là, on s’est contenté de mentionner les capaci- tés organisationnelles de SGBD, et non leur aptitude à manipuler des données spatiales. Ce point, dans les SGBD relationnels, constitue un filtre important : sur la centaine de solutions disponibles, seule une poignée est en mesure de stocker efficacement et d’interroger de l’information spatiale 37.

Pourtant, ~~a u regard des indicateurs de sortie de simulation sur lesquels on s’ap-~~  ~~puie pour évaluer le comportement de SimFeodal,~~ une grande majorité est non spatiale, en raison de la difficulté à agréger des données spatiales théoriques. Dans SimFeodal, la plupart des indicateurs de sortie sont non spatiaux, c’est- à-dire qu’ils mobilisent plus la dimension attributaire des données que leur dimension géographique 38.

La gestion de données spatiales ne constitue donc pas une absolue nécessité, contrairement aux points évoqués auparavant. Elle peut toutefois se révéler avantageuse, ne serait-ce que pour permettre l’observation des configurations spatiales simulées. Cela constitue une approche idiographique~~,~~ visant à exem- plifier plus qu’à synthétiser, mais permet tout de même une compréhension ra- pide des changements de structures spatiales. Toutes choses égales par ailleurs, on privilégiera donc des solutions de stockage ayant une capacité à gérer les données spatiales.

Pour héberger et organiser les données produites par SimFeodal, en vue de

leur interrogation dans SimEDB, nous avons choisi de restreindre la myriade de solutions disponibles grâce à plusieurs filtres successifs. En premier lieu, on a choisit de faire appel à des solutions centralisées (Stockage distribué ou centralisé), au sein de Systèmes de Gestion de Base de Données (SGBD). Ces SGBD permettent une interrogation standardisée (Interrogation spécifique ou générique) via un langage de requête universel, le SQL (Bases de données re- lationnelles ou NoSQL). On a ensuite décidé d’interroger ces SGBD sans passer par l’intermédiaire d’entrepôts de données, et au travers de connexion aussi

37. Les données spatiales peuvent être stockées dans tous les SGBD si l’on attribue une re- présentation textuelle, en chaînes de caractères, par exemple en utilisant le format *Well-Known Text* (WKT). Pour autant, ce format est lourd, inadapté à une indexation, et ne peut permettre à un SGBD de mener des requêtes spatiales directement depuis ces entités. Il est ainsi, par exemple, impossible de calculer le centroïde d’un polygone directement depuis une représen- tation WKT, alors que c’est aisé avec un stockage géométrique.

38. Les raisons en sont multiples et on y reviendra largement dans le

7. Notons tout

de même que les indicateurs résultent de l’agrégation des réplications, et que cette agréga- tion est extrêmement complexe sinon impossible sur des données spatiales majoritairement

aléatoires (voir

2.2.2.1).

41

***Lucigares du pharaon***

*2019-11-15 15:50:55*

--------------------------------------------

"à la base de données"

***Lucigares du pharaon***

*2019-11-15 15:54:33*

--------------------------------------------

je trouve que tu devrais définir plus explicitement le rapport généricité - agnosticité car tu l'annonces avant cette partie, puis ici mais seulement de manière "faible". Juste qqpart un "ce caractère universel/générique, que l'on peut aussi qualifier d'agnostique"

+ je vois pas trop pq l'usage de cette expression que tu n'emploies finalement que 2 fois en plus...

***Lucigares du pharaon***

*2019-11-15 15:55:07*

--------------------------------------------

"De plus, toujours dans l'optique de conserver une généricité/universalité...."

***Lucigares du pharaon***

*2019-11-15 15:56:42*

--------------------------------------------

"par exemple/ de type" ? bizarre sinon car tu nous définies tout, mais pas ces termes là...

***Lucigares du pharaon***

*2019-11-15 15:57:43*

--------------------------------------------

"en fait"

***Lucigares du pharaon***

*2019-11-15 15:58:18*

--------------------------------------------

"ce calcul"

***Lucigares du pharaon***

*2019-11-15 16:00:49*

--------------------------------------------

"une grande majorité des indicateurs de simulation de SimFeodal" (pour simplifier, sinon la phrase va bien, mais je pense qu'il n'est pas besoin de redire à chaque fois "pour évaluer le comportement" "sur lesquels on s'appuie", etc.)

***Lucigares du pharaon***

*2019-11-15 16:01:09*

--------------------------------------------

"Ainsi, dans". Sinon tu ne fais que te répéter.

***Lucigares du pharaon***

*2019-11-15 16:02:27*

--------------------------------------------

"n'est donc pas une priorité dans nos facteurs de choix de SBD"

***Lucigares du pharaon***

*2019-11-15 16:03:00*

--------------------------------------------

"cet usage de la dimension spatiale relève alors d'une approche etc."

***Lucigares du pharaon***

*2019-11-15 16:03:32*

--------------------------------------------

"la"

***Lucigares du pharaon***

*2019-11-15 16:03:38*

--------------------------------------------

"de"

***Lucigares du pharaon***

*2019-11-15 16:03:59*

--------------------------------------------

donc

***Lucigares du pharaon***

*2019-11-15 16:04:23*

--------------------------------------------

majuscule ?

chapitre 2, section

chapitre

Chapitre 5

5.3. ORGANISER LES DONNÉES

standardisées que possible (Entrepôts de données et interrogation directe). Les

SGBD répondant à ces critères sont les SGBD ~~«~~ relationnels ~~» ,~~ dont certains pos- sèdent qui plus est une capacité intéressante à stocker et interroger des données spatiales (SGBD et données spatiales), ce qui constitue notre dernier filtre.

**5.3.1.2 Interroger de manière robuste et performante**

En dépit de l’accumulation de critères exposée précédemment, une quantité

importante de SGBD demeurent en lice. Afin de les différencier, nous avons choisi d’ajouter des critères qui ne portent plus sur les grands types de SGBD, mais plutôt sur une différenciation des SGBD relationnels existants. Ces deux

~~critères, précisés par la suite,~~ sont d’une part la robustesse des SGBD, et d’autre part leurs performances. Ces critères sont pas des « prétextes » à une hiérarchi- sation quantifiée des SGBD, mais ont une importance prépondérante dans notre cas d’utilisation.

**Robustesse des SGBD.** Le premier critère ajouté est celui de la robus-

tesse, c’est-à-dire, ici, de la capacité du SGBD à être interrogé de manière (1) stable et (2) pérenne dans le temps. Une même requête sur les mêmes données doit systématiquement renvoyer le même résultat (stabilité), quelle que soit la durée séparant ces requêtes (pérennité). Si la base de données n’est plus inter- rogeable quelques mois après sa configuration, ou qu’elle renvoie des résultats différents, alors elle ne peut constituer une solution crédible à l’exploration d’un modèle sur une période longue.

La **stabilité** des bases de données est principalement due à la manière de sto-

cker l’information d’un point de vue informatique.

En premier lieu, l’information peut être contenue « en clair » ou alors de ma- nière archivée. Un stockage « en clair » est plus facilement accessible, puisqu’on peut le consulter avec n’importe quel éditeur de texte. Un stockage archivé est moins universel, mais occupe généralement un espace disque inférieur et com- porte des mécanismes de vérification de la cohérence des données. Il est donc plus stable.

Une second différenciation tient à l’emplacement lieu du stockage. Celui-ci peut être effectué dans un unique fichier, ce qui a l’immense avantage de la portabilité des données : pour faire migrer ou sauvegarder la base de données, il suffit de copier le fichier. La plupart des SGBD adoptent toutefois un mode d’organisation en plusieurs fichiers, notamment pour des questions de redon- dance et de vérification de l’intégrité des données : en multipliant les fichiers, on minimise le risque d’erreur critique sur l’ensemble des fichiers à la fois. On notera enfin un dernier type de SGBD, où l’information n’est pas stockée sur un disque dur, mais est entièrement contenue dans la mémoire vive de l’ordi- nateur : les SGBD « *in-memory* ». Ces SGBD sont les plus rapides et stables, mais il faut les re-constituer à chaque redémarrage du serveur qui les héberge, ce qui peut prendre un temps important.

L’enjeu du choix est de se prémunir de « corruptions » de la base de données :

quand le SGBD ne comporte que pas ou peu de mécanismes de vérification de l’intégrité ou de la cohérence des données, il peut arriver qu’une base de 42

***Lucigares du pharaon***

*2019-11-15 16:05:05*

--------------------------------------------

"des"

***Lucigares du pharaon***

*2019-11-15 16:05:25*

--------------------------------------------

"la capacité de"

***Lucigares du pharaon***

*2019-11-15 16:06:34*

--------------------------------------------

"qui portent sur"

***Lucigares du pharaon***

*2019-11-15 16:08:37*

--------------------------------------------

gras inutile (là et le prochain : c en début de paragraphe on se repère très bien sur ces deux mots-clefs)

Chapitre 5

5.3. ORGANISER LES DONNÉES

données se corrompe. On peut prendre l’exemple de l’exécution d’une requête

demandant un calcul complexe et long. Cette requête pourrait être interrom- pue en cours d’exécution par faute d’un *bug* ou d’une expiration de session~~s~~  (*timeout*). Dans ce cas, il se peut que la base de données s’arrête dans un état muté – avec une nouvelle table ajoutée pour moitié par exemple – et ne soit donc plus intègre. C’est très fréquent pour les SGBD basés sur un unique fi- chier, ou encore stockés en clair, puisque les nouvelles informations de la base de données y sont ajoutées au fur et à mesure, plutôt que d’être intégrées dans un fichier annexe que l’on pourrait réinitialiser en cas d’erreur.

Avec la volumétrie des données produites par SimFeodal, les requêtes peuvent s’avérer très longues, et une erreur dans une requête peut fréquemment cor- rompre la base de données. En termes de stabilité, on se tournera donc plutôt vers des SGBD relationnels stables, basés sur une redondance des données et donc sur des architectures archivées et multi-fichiers.

La **pérennité** des SGBD est un sujet proche, tenant aussi à la capacité à in- terroger les données contenues dans une base de données, mais cette fois-ci du point de vue de l’interrogation en elle-même plutôt de des données sur lesquelles elle s’applique : si le SQL est un langage standard 39, les types de données intégrées varient cependant d’un SGBD à un autre (champs textuels ou d’entiers « courts » par exemple). SQL étant un langage typé, selon la ma- nière (bas niveau) dont sont intégrées les données, certaines requêtes iden- tiques peuvent renvoyer des résultats différents selon les SGBD. Plus gênant, les normes implémentées peuvent varier d’une version à l’autre d’un SGBD. Un SGBD relationnel respectant strictement la norme SQL pourrait ainsi évoluer pour supporter plus de fonctionnalités, par exemple en ajoutant des fonctions plus récentes (fenêtres glissantes, ajouts en masse etc.), et renverrait dès lors des résultats différents selon les versions. Pour les SGBD les plus employés, le nombre d’utilisateurs garantit une rétro-compatibilité des requêtes. Pour les SGBD de moindre envergure cependant, par exemple les plus performants et récents issus de la recherche en informatique, cette rétro-compatibilité n’est pas ~~d u tout~~ garantie.

Comme souvent en matière d’infrastructure informatique, il est donc nécessaire

de tenir compte d’un compromis entre l’ancienneté et la forte adoption de certains SGBD d’une part, et les facilités et gains de performances amenées par les plus récents d’autre part. Dans le cas des données de SimFeodal, en tenant compte de cet inévitable compromis, nous avons choisi de privilégier des SGBD reconnus, soient-ils anciens et fortement adoptés ou plus récents mais utilisés par des acteurs d’envergure 40. Ce faisant, on se coupe immanquablement de solutions ~~extrêmement~~ intéressantes et performantes 41. Ce choix est toutefois

39. Dans les faits, on notera tout de même qu’il existe plusieurs normes successives, des

« révisions » du SQL, qui apportent chacune leur lot de subtilités dans l’usage du langage. Les SGBD interrogeables en SQL ne disposent donc pas toutes des mêmes fonctionnalités, selon la révision du SQL qu’elles respectent.

40. La liste des solutions envisagées, ensuite comparées à l’aube de leurs performances, est visible dans l’axe des ordonnées de la figure 5.9.

41. Par exemple BlinkDB (Agarwal et al. 2013), qui permet de limiter une requête à un temps maximal d’exécution donné : quand la requête n’est pas complète, le SGBD renvoi une

43

***Lucigares du pharaon***

*2019-11-15 16:08:12*

--------------------------------------------

gras inutile

***Lucigares du pharaon***

*2019-11-15 16:11:45*

--------------------------------------------

". Si"

***Lucigares du pharaon***

*2019-11-15 16:12:15*

--------------------------------------------

version ?

***Lucigares du pharaon***

*2019-11-15 16:12:41*

--------------------------------------------

?

***Lucigares du pharaon***

*2019-11-15 16:13:24*

--------------------------------------------

"d'un SGBD à l'autre" (pour éviter la répétition)

***Lucigares du pharaon***

*2019-11-15 16:14:49*

--------------------------------------------

"du fait de leur grand nombre d'utilisateur, la rétro-compatibilité des requêtes est garantie" ?

***Lucigares du pharaon***

*2019-11-15 16:16:09*

--------------------------------------------

"utilisation"

***Lucigares du pharaon***

*2019-11-15 16:16:57*

--------------------------------------------

"l'aune"

Chapitre 5

5.3. ORGANISER LES DONNÉES

en la large faveur d’une meilleure garantie de pérennité, et de robustesse en

générale, des données de SimFeodal.

**Performance des SGBD.** Une fois que les solutions disponibles ont été

discriminées par leur type, par leur interface avec les requêtes et par leur ro- bustesse, la quantité de SGBD restant demeure de l’ordre de la dizaine. Pour choisir, parmi ceux-là, le SGBD qui sera le plus adapté aux besoins identi- fiés, il est donc nécessaire d’établir des critères plus précis et quantifiables. Dans le cas d’une application interactive, c’est-à-dire où le nombre de requêtes émises au cours d’une session d’utilisation peut être importante, les perfor- mances des SGBD constituent un critère majeur pour départager l’ensemble des SGBD considérés.

Il est difficile de qualifier les « performances » d’un SGBD : on entend en fait par

ce terme un vaste ensemble hétérogènes de propriétés. On peut par exemple juger les performances par le filtre de la mémoire occupée par le stockage d’une base de données, ou encore par le nombre de requêtes concurrentes que peut gérer un SGBD, ou encore par la capacité à paralléliser le stockage sur plusieurs serveurs. Dans notre cas, ces points sont assez peu significatifs : en dépit de la quantité de sorties, l’ordre de grandeur – quelques gigaoctets de données – reste largement entreposable sur un environnement classique, sans besoin de parallélisation. De la même manière, SimEDB est un environnement dédié à des utilisateurs experts, en petit nombre : les chercheurs travaillant autour de SimFeodal. La quantité de requêtes simultanées ne peut donc pas dépasser la dizaine, ce qui constitue une trivialité pour l’ensemble des SGBD relationnels classiques. On s’attachera donc à juger les performances en matière de rapidité d’exécution des requêtes. Il ne s’agit pas ici de choisir un SGBD qui ferait gagner quelques millisecondes par rapport à un autre, mais plutôt de discriminer les SGBD présentant une durée de réponse trop importante pour notre usage.

En effet, plus les données sont massives, plus le temps d’exécution d’une re-

quête augmente, souvent sous la forme d’une fonction puissance. Si tous les SGBD présentent des vitesses acceptables et proches sur des bases de données de faible volume, l’écart s’accroît considérablement à mesure que les données s’accumulent. La figure 5.9 42 montre les différences ~~i ncontestables~~ qui existent entre les SGBD étudiés. On peut y constater que l’écart est gigantesque, par exemple vis-à-vis du temps nécessaire à une jointure, entre les 4 secondes de MonetDB et les 300 secondes (5 minutes…) de SQLite. Le choix d’un SGBD se- lon ses performances a donc un impact majeur sur la fluidité d’une application d’exploration de données massives. Pour départager les SGBD, on comparera

estimation du résultat, estimation qui gagne en précision quand on augmente la limite tem- porelle. Un SGBD de ce type serait extrêmement précieux en *visual analytics*, mais la jeunesse de cet outil ainsi que sa nature de projet de recherche rendent incertain la continuité de son développement dans le temps. En 2018, le projet semble d’ailleurs avoir été abandonné~~…~~

42. Dans cette figure, on compare la rapidité de différentes requêtes sur un jeu de données identique selon les SGBD. Ce type de test de performance permettant la comparaison de solu- tions techniques diverses est appelé *benchmark*. Ce jeu de données, composé de 100 Millions de lignes et de deux colonnes numériques, présente une volumétrie comparable (franchement inférieure en nombre de colonnes toutefois) à celle des données issues de SimFeodal qui sont interrogées dans SimEDB.

44

***Lucigares du pharaon***

*2019-11-15 16:26:30*

--------------------------------------------

"Nous" vraiment ici pour marquer le retour à tes enjeux à toi (j'aurai mis "ici" en fait mais tu l'utilises dans la phrase du dessous ^^)

***Lucigares du pharaon***

*2019-11-15 16:27:27*

--------------------------------------------

"d'éliminer" ? j'ai plus internet pour vérifier l'usage du verbe discriminer ><

***Lucigares du pharaon***

*2019-11-15 16:29:04*

--------------------------------------------

"fortes"

***Lucigares du pharaon***

*2019-11-15 16:30:10*

--------------------------------------------

"de ce point de vue"

***Lucigares du pharaon***

*2019-11-15 16:30:31*

--------------------------------------------

"très important" (j'ai du mal avec les superlatifs ^^)

***Lucigares du pharaon***

*2019-11-15 16:36:30*

--------------------------------------------

proposition de changement d'ordre pour la phrase :

"Par exemple, concernant le temps nécessaire à une jointure, on peut y constater, etc." + pas de "..." après "5min"

***Lucigares du pharaon***

*2019-11-15 16:48:05*

--------------------------------------------

"nous avons comparé ?" ou au présent ? en tous cas le futur est chelou car tu ne l'as jamais utilisé jusque là

***Lucigares du pharaon***

*2019-11-15 16:47:33*

--------------------------------------------

"largement" ?

***Lucigares du pharaon***

*2019-11-15 16:50:24*

--------------------------------------------

un peu bizarre que tu mettes cet exemple ici alors que tu présentes et introduit la figure véritablement en dessous - et y compris la note du coup... est-ce que tu ne peux pas ici mettre "(voir ci dessous le commentaire de la figure 5.9)" et replacer cet exemple + loin ?

Chapitre 5

5.3. ORGANISER LES DONNÉES

leurs performances selon les différents types d’opérations demandées, qu’elles

concernent l’écriture dans la base (insertion) ou des types de lecture (agréga- tion et jointure).

**Performances en écriture et en lecture.** La figure 5.9 affiche des résul-

tats qui semblent globalement ordonnés (les quatre premiers SGBD sont par exemple quasiment toujours plus lents que les 2 derniers), mais fluctuent ce- pendant à la marge selon les opérations demandées. La première colonne du graphique montre ainsi le temps nécessaire à l’insertion du jeu de données exemple ~~( voir la note de bas de page 42)~~ dans le SGBD depuis un fichier CSV. Les deux colonnes suivantes exposent le temps nécessaire au traitement d’une requête, donc à une interrogation des données ~~u ne fois archivées dans les bases~~

~~de données.~~ On peut constater que le classement des SGBD varie faiblement en lecture, et de manière assez faible en insertion : les deuxièmes et troisièmes colonnes respectent un ordre globalement similaire, assez différent de celui de la première colonne. Dans un environnement classique, la performance d’in- sertion de données est un facteur prépondérant : quand de nouvelles données sont ajoutées constamment, par exemple pour stocker des données issues de capteurs automatiques, l’insertion peut vite constituer le goulot d’étranglement de la solution.

Pour SimFeodal, l’insertion n’est pas véritablement un enjeu : les données sont

ajoutées par bloc, manuellement, une fois que des nouvelles simulations ont été exécutées. C’est donc au pire un acte quotidien, mais dans ce cas, que la requête demande 10 secondes (MapD) ou 10 minutes (MySQL InnoDB), cela n’a que peu d’impact. La première colonne est donc un indicateur de performance peu adapté dans notre cas.

Les deux colonnes suivantes, relatives à l’interrogation de données, se révèlent

au contraire extrêmement importantes : à chaque action de l’utilisateur de SimEDB, une nouvelle requête est envoyée pour calculer un nouvel indica- teur correspondant au jeu de données filtré manuellement (cf. section 5.2.5). À chaque affichage d’onglet, une nouvelle requête est donc émise et traitée. Même si tous les indicateurs ne sont pas systématiquement mobilisés – et donc

calculés – (cf.

3), cela signifie tout de même que pour chaque sélection,

une bonne dizaine d’indicateurs seront observés, et donc, autant de requêtes.

Quand une requête demande 60 secondes (par exemple PostgreSQL en « join- ture »), cela implique que chaque indicateur requiert au moins une minute avant de s’afficher. Pour observer une dizaine indicateurs, l’utilisateur devra donc attendre ~~p atiemment~~ une dizaine de minutes, sans même tenir compte du temps qu’il passera à les analyser visuellement.

Pour noircir le trait, notons de plus que les résultats communiqués dans la

figure 5.9 correspondent à des requêtes simples qui ont valeur d’exemples de base. Dans le cas de SimEDB, le calcul des indicateurs requiert des requêtes plus complexes, faisant appel à des agrégations et à des jointures en même temps, et les délais affichés dans ce *benchmark* sont donc bien inférieures aux durées éprouvées en conditions réelles au sein de SimEDB.

45

***Lucigares du pharaon***

*2019-11-15 16:57:28*

--------------------------------------------

En fait je pense qu'il faut effectivement retirer l'exemple explicite du dessus, et présenter ici formellement la figure 5.9 , en reprenant//réarrangeant des parties du paragraphe du dessous (voir en commençant le paragraphe du dessous avec cette présentation de la figure) en mode "Les résultats de la comparaison de la vitesse d'exécution de X SGBD candidats sont présentés sur la figure 5.9 // note// . La première colonne du graphique montre //etc description//. Globalement, ont lit sur la figure des résultats qui semblent ordonnés etc." puis tu enchaîne avec l'interprétation des différentes résultats (lecture/insertion) auquel tu peux ajouter l'exemple de la jointure.

***Lucigares du pharaon***

*2019-11-15 16:58:27*

--------------------------------------------

inutile, d'autant plus avec ma proposition de changement

***Lucigares du pharaon***

*2019-11-15 17:00:03*

--------------------------------------------

inutile (mais si tu gardes il faut rajouter "une fois ces dernières archivées")

***Lucigares du pharaon***

*2019-11-15 17:01:27*

--------------------------------------------

pas de saut de paragraphe

***Lucigares du pharaon***

*2019-11-15 17:01:34*

--------------------------------------------

"en revanche"

***Lucigares du pharaon***

*2019-11-15 17:02:54*

--------------------------------------------

pourquoi l'avoir fait alors ? Pour que ça "choque moins", plutôt

"La première colonne est donc un indicateur de performance mineur/de faible importance dans notre cas"

***Lucigares du pharaon***

*2019-11-15 17:04:45*

--------------------------------------------

"avant même de pouvoir commencer à les analyser visuellement" (pas la peine je pense de parler du "temps d'analyse visuelle" ici, qui est plus incompréhensible et logiquement acceptable)

***Lucigares du pharaon***

*2019-11-15 17:05:20*

--------------------------------------------

"basiques" ? "de base" m'a troublée car j'attendais le "de données" derrière :p

***Lucigares du pharaon***

*2019-11-15 17:05:44*

--------------------------------------------

"en fait"

chap

Chapitre 5

5.3.

ORGANISER LES DONNÉES

Performances comparées de SGBD

Temps nécessaires à l'execution de requêtes

SQLite (file + index)\* (SQL)

**●**

SQLite (memory)\* (SQL)

MySQL InnoDB (SQL)

**●**

PostgreSQL (SQL)

**●**

MySQL MyISAM (SQL)

**●**

R dplyr (SQL)

**●**

Spark (SQL)

**●**

Python pandas (non SQL)

**●**

R DT

(non SQL)

Spark−cached (SQL)

MonetDB (SQL)

MapD (CPU)\* (SQL)

MapD (GPU) (SQL)

Durée de la requête [secondes] (Échelle logarithmique)

Types de Bases de Données

Analytiques Massives Relationelles Statistiques

R. Cura (2018), d'après S. Pafka (2017)

\* MapD (CPU) et SQLite : Benchmark mené sur un système moins puissant que les autres

Figure 5.9 – Comparaison de la performance de différents SGBD sur un jeu

de données test de 100 millions de lignes. Résultats tirés de Pafka (2017) et complétés par l’auteur.

Les « types de Bases de Données » correspondent aux usages les plus fré- quents des SGBD comparés :

- Analytiques : SGBD optimisés pour les traitements de type agrégation, via une architecture orientée colonne plutôt qu’orientée ligne comme dans les SBGD Relationnels. Ils sont optimisés pour la rapidité d’exécution.

- Massives : SGBD pensés pour la gestion et l’interrogation de données mas- sives (big data), permettant notamment une parallélisation des requêtes. Ils sont optimisés pour la capacité à gérer des volumes gigantesques de données.

- Statistiques : SGBD internes aux environnements de traitement de don- nées statistiques, reposant sur une gestion en mémoire vive. Souvent inté- grés d’office dans les environnements décrits (R, Python), ce sont les SGBD les plus simples à mettre en place et à manipuler.

46

Systèmes de Gestion de Bases de Données

***Lucigares du pharaon***

*2019-11-15 16:28:35*

--------------------------------------------

idem qu'avant sur différence entre titre et légende de la figure

***Lucigares du pharaon***

*2019-11-15 17:08:16*

--------------------------------------------

je ne m'attendais à voir sur cette figure que des bdd relationnelles puisque que tu dis que c'est celles-ci que tu as choisi d'utiliser avant même les questions de performances... Du coup je pense qu'il faut que tu précise, par exemple en note de bas de page au moment où je te conseille e présenter la figure, que d'autres types de SGBD sont aussi testés (massives, analytiques, statistiques) "à titre d'information".

***Lucigares du pharaon***

*2019-11-15 17:19:13*

--------------------------------------------

l'explication de la différence gpu/cpu arrive très tard... elle est par ailleurs intéressante mais peut-être pas utile sur la figure ? (pas centrale en fait comme tu n'en parles que en note de bas de page)

**●**

**●**

**●**

**●**

Jointure

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

Aggrégation de données

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

Insertion de données

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

Chapitre 5

5.3. ORGANISER LES DONNÉES

**De l’intérêt de gagner quelques secondes.** La figure 5.9 permet d’isoler un

sous-ensemble de quatre SGBD ayant, avec le jeu de données testé, des réponses inférieures à une dizaine de secondes : Spark avec cache, MonetDB et MapD sur CPU ou GPU. On pourrait se contenter de choisir le SGBD le plus complet parmi ces quatre solutions.

Pourtant, un autre domaine d’étude appuie l’importance relative des écarts,

mêmes faibles, dans les durées de requête. Ce domaine est celui des sites in- ternet, où les requêtes servent à générer le contenu de différentes pages en interrogeant des base de données de contenu. La consultation d’un site inter- net consiste à charger plusieurs pages, pour l’utilisateur. Du point de vue du serveur, chacune des pages demandées par l’utilisateur requiert différentes re- quêtes à des bases de données. La navigation dans un site est donc assez com- parable à l’utilisation d’une application d’exploration de données : des requêtes hétérogènes, plus ou moins lourdes, s’y succèdent et visent à filtrer et mettre en forme, de manière explicite, des extraits d’informations stockées dans des bases de données. Plusieurs études ont montré que la durée d’affichage d’une page web jouait de manière considérable sur l’usage d’un site, composé de plusieurs de ces pages. L’étude la plus parlante est décrite par Neil Patel qui relate une expérience vécue au sein du moteur de recherche Google :

« Google did an interesting experiment with regard to load times. Google Vice President Marissa Mayer asked web surfers – would you rather see 10 or 30 results for your Google search ? The users agreed that 30 results per page sounded like a good idea. So Google implemented it on some results pages. Then the shock came.

Pages that displayed 30 results each had traffic to them drop an astounding 20%. Google tested the loading difference between the 10 and 30 results pages and found that it was just **half of a second**. If half of a second made that much of a difference in how long users were willing to wait, how much of a difference could it make to your site if you carved a second or two off of load time ? »

(Patel 2011)

Si l’environnement et les conditions décrites ne sont pas directement compa- rables avec celles de SimEDB, il demeure qu’une différence même faible dans un temps de chargement, ou, pour SimEDB, dans un temps d’affichage d’un in- dicateur de sortie, pourrait avoir des conséquences négatives pour l’utilisation de la plate-forme.

Un autre exemple appuie ce raisonnement et répond à la dernière interrogation

de Patel, dans un cadre un peu plus proche de SimEDB. Roxana Elliott, employée d’une société qui propose des solutions d’accélération de sites web, a réalisé un rapport sur les perte d’audience des sites webs en fonction du temps de chargement des pages (Elliott 2017). Les résultats de son étude sont présentés dans la figure 5.10, et permettent de quantifier un effet bien connu, qui veut que l’utilisateur quitte plus rapidement un site (et en visite donc moins de pages) quand les pages sont plus longues à charger.

47

***Lucigares du pharaon***

*2019-11-15 17:10:03*

--------------------------------------------

mais donc finalement pas les relationnelles ????

***Lucigares du pharaon***

*2019-11-15 17:11:38*

--------------------------------------------

"Cependant" me semble plus adapté ou "toutefois"

Chapitre 5

5.3. ORGANISER LES DONNÉES

Nombre de pages visitées par site selon leur durée d'affichage

Plus le temps de chargement est long, moins le visiteur consultera de pages

10

8

6

4

2

2

4

6

8

Temps de chargement d'une page [secondes]

Sources : D'après section.io [(www.section.io/blog/page](http://www.section.io/blog/page)−load−time−bounce−rate/)

Figure 5.10 – Impact du temps de chargement d’un site web sur sa consulta- tion. D’après Elliott (2017).

Cet exemple est plus directement comparables aux contraintes de SimEDB.

Chaque consultation d’un indicateur de sortie correspond ainsi à la vue d’une page dans cet exemple. Ces chiffres renforcent l’importance à accorder au temps de chargement des indicateurs, et donc à la durée de l’exécution des requêtes qui les génèrent. Quelques secondes de différence dans le chargement suffisent ainsi à réduire drastiquement le nombre d’indicateurs que l’utilisateur acceptera d’analyser.

On peut toutefois pondérer ces comparaisons et minimiser l’importance d’écarts

de l’ordre de grandeur de la seconde. Dans le cas de SimEDB, contrairement à celui d’un site web ou d’un moteur de recherche, l’utilisateur est « captif ». Cela signifie qu’un thématicien souhaitant explorer les résultats produits par SimFeodal n’aura d’autre choix que de passer par SimEDB. De même, sachant que la plate-forme présente pour lui un intérêt professionnel, le thématicien sera bien plus patient que face à un quelconque site de courses en lignes.

Dans le cadre d’environnements de type *visual analysis*, il a été montré que

les utilisateurs d’environnement d’exploration étaient toutefois fortement af- fectés par l’accroissement de délais. Zhicheng Liu et Jeffrey Heer (Liu et Heer 2014) montrent ainsi qu’en introduisant une latence supplémentaire de 500 ms dans une application interactive d’exploration de données spatio-temporelles, le nombre d’interactions chute fortement, quand bien certains utilisateurs de cette application ne remarquent même pas la différence de délai.

**Choisir un SGBD adapté à SimEDB.** L’ensemble de filtres successifs a

permis de réduire progressivement la quantité de solutions logicielles appro- priées à l’organisation et à l’interrogation des données issues de SimFeodal. Depuis les centaines de solutions disponibles, on parvient ainsi dans un pre- mier temps à isoler les grands types de SGBD correspondant aux besoins iden- tifier : les SGBD relationnels, basés sur une interrogation standardisée en SQL. Ces outils sont ensuite départagés au prisme de leur robustesse, intrinsèque (stabilité) et sur la base de leur niveau d’adoption (pérennité). Un *benchmark* finit de restreindre la liste des possibles à quelques solutions envisageables en fonction des besoins soulevés par SimEDB.

48

Nombre de pages vues

***Lucigares du pharaon***

*2019-11-15 17:12:47*

--------------------------------------------

pas sûre de ce sous-titre qui est un commentaire de la figure

***Lucigares du pharaon***

*2019-11-15 17:13:16*

--------------------------------------------

"peut être rapporté" ?

***Lucigares du pharaon***

*2019-11-15 17:13:34*

--------------------------------------------

"de plus" dans ce cas non ?

***Lucigares du pharaon***

*2019-11-15 17:14:18*

--------------------------------------------

"site web", l'exemple est bizarre ça peut être un truc commerçant ou des courses de chevaux ^^

***Lucigares du pharaon***

*2019-11-15 17:14:43*

--------------------------------------------

pas de saut de paragraphe

***Lucigares du pharaon***

*2019-11-15 17:16:19*

--------------------------------------------

bon du coup si tous les sgbd de ton benchmark sont relationnels avec différents sous-types, il faut le préciser. dans ta légende de la figure 5.9 comme il y a relationnel / et d'autres c pas clair du tout :(

●

●

●

●

●

●

●

Chapitre 5

5.3. ORGANISER LES DONNÉES

Au regard des performances de chacun des SGBD, MapD (Root et Mostak

2016) présente l’avantage indéniable de la vitesse de traitement des requêtes, tout en étant compatible avec les standards de l’interrogation de données (lan- gage de requête SQL, interfaçable via JDBC). Même exécuté sur un infrastruc- ture informatique n’est pas optimisée pour cette solution 43, MapD est incon- testablement plus performant que les autres SGBD.

On notera toutefois que le SGBD MonetDB (Vermeij et al. 2008), dans son

implémentation intégrée MonetDBLite (Raasveldt et Mühleisen 2018), af- fiche aussi des performances très compétitives, et aurait pu être choisi pour SimEDB, présentant notamment l’avantage d’être plus utilisé et ancien 44. Un des ouvrages de référence en *visual analytics* s’interrogeait d’ailleurs sur les nou- velles possibilités et l’adéquation offertes par ce SGBD (Fekete 2010, p. 105 in Keim et al. 2010).

Nous avons au final préféré MapD, en particulier parce que les données issues

de SimFeodal sont amenées à augmenter, renforçant donc petit à petit l’écart de performance entre MapD et MonetDB. Par ailleurs, par une heureuse coïnci- dence, MapD a été placé sous licence libre peu avant que nous n’ayons à nous pencher réellement sur les problèmes de performances et de robustesse qui apparaissaient suite à l’augmentation du nombre de simulations effectuées.

**5.3.2**

**Structuration des données de SimFeodal**

Le choix d’un SGBD est une étape indispensable à la mise en place d’une base de

données, mais il ne concerne que le domaine technique, voire méthodologique, mais aucunement le domaine conceptuel. Un SGBD est un support logiciel qui permet le stockage et l’organisation de données. Il n’est utile qu’une fois que le mode d’organisation des données a été décidé. L’organisation a proprement parler des données est explicitée dans un modèle conceptuel, nommé Modèle Conceptuel de Données (MCD). Un MCD est propre à un ensemble de données d’une part 45, et un ensemble de problématiques d’autres part 46. Ce MCD dé- crit donc les « tables », leur composition (attributs) et les liens entre tables qui permettent de mener des interrogations croisées. Par exemple, on peut avoir une table élèves, contenant les informations relatives aux élèves d’un établis- sement, une table enseignants, et une table classe qui permet de faire le lien

43. MapD est ~~a insi~~ un SGBD optimisé pour l’analyse sur processeurs graphiques (les GPU), présents dans les cartes graphiques modernes, contrairement aux SGBD classiques qui s’ap- puient sur les processeurs (CPU) pour effectuer leurs calculs. Dans le cadre de cette thèse, nous n’avions pas accès à un serveur doté de GPU, et MapD est donc installé sur une infra- structure à base de CPU, bien moins performante.

44. Dans les faits, MonetDBLite a été le SGBD utilisé pendant une large partie de la concep- tion de SimEDB. Il s’est toutefois révélé assez instable dans notre cas, faisant preuves à plusieurs reprises de corruptions de données ayant entraîné l’obligation de recréer entièrement les bases de données depuis les fichiers bruts produits par SimFeodal.

45. Le MCD décrit la manière dont les données sont stockées, organisées et mises en relations. Il ne peut donc être générique, et doit être modifié quand la structure des données évolue.

46. Il y a une infinité de possibilité d’organisation d’un même jeu de données. Le MCD permet d’organiser ces données en vue de répondre à des questions, exprimées sous formes de requêtes particulières. Appliquées au même jeu de données, différents MCD permettront de répondre plus ou moins facilement (et de manière plus ou moins performante) à certaines questions.

49

***Lucigares du pharaon***

*2019-11-15 17:16:59*

--------------------------------------------

"une"

***Lucigares du pharaon***

*2019-11-15 17:17:30*

--------------------------------------------

phrase ?

***Lucigares du pharaon***

*2019-11-15 17:26:07*

--------------------------------------------

pq des guillemets ?

***Lucigares du pharaon***

*2019-11-15 18:03:28*

--------------------------------------------

Pour mettre plus en valeur le choix final, je pense qu'il faut réorganiser ces deux derniers paragraphes. En particulier, 1. je pense qu'une partie de la note 44 doit passer dans le corps du texte car c'est un vrai argument pour le choix. 2. A l'inverse, le passage sous licence libre "juste avant" est cool mais pas central, du coup ça le mettre en note de bas de page. Proposition :

"Un des ouvrages blabalbalbalb (Fekete 2010, p. 105 in Keim et al. 2010). "Toutefois, les données issues de SimFeodal sont amenées à augmenter, renforçant donc petit à petit l'écart de performance entre MapD et MonetDB. De plus, dans les faits, nous avons utilisé MonetDBLite pendant une large partie de la conception de SimED, mais ce SGBD s'est montré à plusieurs reprises instable // note de bas de page + courte qui décrit l'instabilité//. Au terme de notre réflexion sur nos besoins et de notre étude comparative des différents SGBD envisageables, nous avons donc choisi d'utiliser le SGBD relationnel MapDB pour stocker les données issues de SimFeodal // note de bas de page sur le fait qu'il est devenu libre juste avant youpi//".

(ou "c'est donc le relationnel MapDB que nous avons choisi d'utiliser pour stocker les données etc.")

Chapitre 5

5.3. ORGANISER LES DONNÉES

entre les élèves d’un enseignant, ou au contraire entre un élève et tous ses

enseignants.

Les choix de conception d’un MCD sont fortement lié aux types de SGBD dans

lesquelles ils doivent être implémentés. On ne peut que difficilement implé- menter un MCD très relationnel dans un SGBD pensé collection (NoSQL par exemple). À l’inverse, le stockage d’informations très hétérogènes sur un en- semble d’individus sera complexe à implémenter au sein d’un SGBD relation- nel.

Pour décider de la manière la plus efficace d’implémenter les données issues

de SimFeodal dans un SGBD, et donc du MCD à suivre, il convient de revenir aux spécificités des données produites par le modèle d’une part, et d’autre part de réfléchir aux modes d’interrogations privilégiés, lesquels orienteront la conception du MCD.

**Pré-traitement des données.** Les données produites par un modèle de

simulation sont des données « brutes », c’est-à-dire qu’elles ne sont pas organi- sées de manière rationnelle, contiennent une quantité non négligeable d’infor- mations incomplètes, superflues ou erronées.

—

Par exemple, quand une simulation est arrêtée en cours, soit volontaire-

ment, soit en raison d’un *bug*, les données générées par le modèle sont **in- complètes** : elles ne concernent qu’une partie des pas de temps attendus. Elles sont pourtant exportées dans les fichiers bruts, rendant ceux-ci hé- térogènes en matière de complétion des informations enregistrées. Pour pouvoir analyser une expérience, il faudra supprimer ces données incom- plètes pour qu’elles n’influence pas l’étude des simulations complétées et donc comparables.

De la même manière, il arrive qu’on exécute, par erreur, plusieurs fois les mêmes simulations. Dans ce cas, le nombre de réplications de chacune des expériences ne sera pas systématiquement le même. Cela pose un problème de comparabilité dû à tailles d’échantillonnage différentes. On fait donc face à un problème de données **superflues** : il faudra supprimer une partie de ces simulations des données avant de pouvoir les traiter. On peut enfin voir subvenir des erreurs d’exécution du modèle au niveau

des agents, par exemple quand, en raison d’un *bug*, un agent en interroge un autre qui a disparu depuis. Il arrive ainsi fréquemment que des foyers paysans déclarent une appartenance à un agrégat qui a disparu ~~d epuis,~~  faute à une mise à jour échouée dans le modèle. Dans ces cas, les données seront aussi inscrites dans les sorties de SimFeodal, quand bien même elles sont **erronées**.

—

—

Les données brutes doivent donc nécessairement être vérifiées, filtrées, net-

toyées et retravaillées avant de pouvoir les exploiter en vue de générer les indicateurs de sortie.

**Organisation des données.** Même pré-traitées, les données brutes conservent

une structure tabulaire assez peu adaptée~~s~~ à un traitement. les attributs de chacun des types d’agent sont enregistré dans des fichier spécifiques. Que 50

***Lucigares du pharaon***

*2019-11-15 17:26:49*

--------------------------------------------

pas de saut de paragraphe

***Lucigares du pharaon***

*2019-11-15 17:27:12*

--------------------------------------------

"donc" ?

***Lucigares du pharaon***

*2019-11-15 17:28:48*

--------------------------------------------

vraiment générique à tous les modèles ? si non tu peux simplement dire "généralement" OU préciser "par le modèle simFeodal"

***Lucigares du pharaon***

*2019-11-15 17:29:49*

--------------------------------------------

"afin"

***Lucigares du pharaon***

*2019-11-15 17:29:44*

--------------------------------------------

"influencent"

***Lucigares du pharaon***

*2019-11-15 17:30:22*

--------------------------------------------

"et, elles, comparables"

***Lucigares du pharaon***

*2019-11-15 17:30:55*

--------------------------------------------

des

***Lucigares du pharaon***

*2019-11-15 17:31:37*

--------------------------------------------

"qui a échoué ?"

***Lucigares du pharaon***

*2019-11-15 17:35:06*

--------------------------------------------

"(voir début du chapitre)" ?

***Lucigares du pharaon***

*2019-11-15 17:35:29*

-------------------------------------------- L

Chapitre 5

5.3. ORGANISER LES DONNÉES

ces fichiers aient été nettoyés ou non, ils demeurent fondamentalement iso-

lés les uns des autres. Une partie des indicateurs repose sur des analyses croi- sant différents types d’agents (dans quels pôles les agrégats s’inscrivent-ils par exemple ?), et il est ~~d onc~~ nécessaire de permettre – et de fluidifier – ces requêtes croisées. On a mentionné le choix de SGBD relationnels, il convient donc de concevoir et d’implémenter, dans le SGBD choisi, les relations entre les diffé- rentes tables individuelles qui proviennent des sorties brutes d’un modèle.

**5.3.2.1 Quel modèle de données ?**

Les MCD sont propres à chaque ensemble de données et questionnement asso-

ciés. Il y a toutefois des grandes tendances dans l’organisation des données. Le MCD peut ainsi être catégorisé, selon sa forme, dans des familles de modèles de données. On nomme ces catégories « modèles logiques » ou « schémas » (*logical schema* en anglais). Ceux-ci décrivent la manière dont les données sont struc- turées et surtout reliées les unes aux autres, d’une manière générique contrai- rement aux MCD.

**Un modèle « en étoile ».** Les bases de données relationnelles peuvent

s’appuyer sur de nombreux schémas différents. Sans entrer dans le détail, no- tons que chacun des schémas existant présente des avantages et des incon- vénients liés aux types de requêtes qui lui seront adressés. Par exemple, un schéma « en étoile » (*Star Schema* 2018) privilégie l’efficacité de requêtes d’agré- gations et de jointures, au détriment de la robustesse des données et de la diversité possible des requêtes. Certains types de requêtes, complexes, seront ainsi difficiles, voire impossibles, à exprimer dans ce type de schéma.

Au contraire, un schéma « en flocons » (*Snowflake Schema* 2018) peut se révéler

plus permissif en terme de capacités de requêtes. L’inconvénient est une plus forte complexité des requêtes de bases (exprimées de manière plus verbeuses et tortueuses) et donc une expressivité moindre.

Pour choisir un schéma, et donc une manière d’organiser la base de données,

il convient donc de savoir – ou de prévoir – le type de requêtes qui lui seront adressées. Dans le cas des données de SimFeodal, les indicateurs avaient été définis avant que le besoin d’une interrogation performante et structurée n’ap- paraisse. ~~O n connaissait déjà les indicateurs nécessaires et le type de requêtes~~

~~associés.~~ Nous savions ainsi qu’une majorité des requêtes seraient des tâches d’agrégations simples (nombre d’agrégats au cours du temps, taux de foyers paysans dispersés au cours du temps etc.), pour lesquelles il fallait minimiser la complexité des requêtes et calculs.

Il a été choisi de partir d’un schéma en étoile, puisque celui-ci se montre ex-

trêmement efficace pour réduire les besoins en jointures – chronophages – et pour des tâches d’agrégations lourdes.

51

***Lucigares du pharaon***

*2019-11-15 17:48:58*

--------------------------------------------

"Or, "

***Lucigares du pharaon***

*2019-11-15 17:48:54*

--------------------------------------------

"reposant"

***Lucigares du pharaon***

*2019-11-15 17:37:45*

--------------------------------------------

"Le choix d'un SGBD relationnel nous permet de concevoir et d'implémenter de manière explicite (???) ces relations entre etc."

la phrase est bizarre sinon, tu n'as pas "mentionné" le choix, tu l'as argumenté et présenté de ouf !! du coup là ça te permets de mettre une fois de + en valeur la justesse de ce choix !

***Lucigares du pharaon***

*2019-11-15 17:39:08*

--------------------------------------------

"Ces modèles logiques"

***Lucigares du pharaon***

*2019-11-15 17:40:37*

--------------------------------------------

",surtout,"

***Lucigares du pharaon***

*2019-11-15 17:48:49*

--------------------------------------------

au présent ?

***Lucigares du pharaon***

*2019-11-15 18:11:00*

--------------------------------------------

"d'indicateurs nécessiteraient des requêtes correspondant à des tâches d'agrégations simples"

***Lucigares du pharaon***

*2019-11-15 18:11:35*

--------------------------------------------

"pour traiter des jointures" ?

***Lucigares du pharaon***

*2019-11-15 18:11:59*

--------------------------------------------

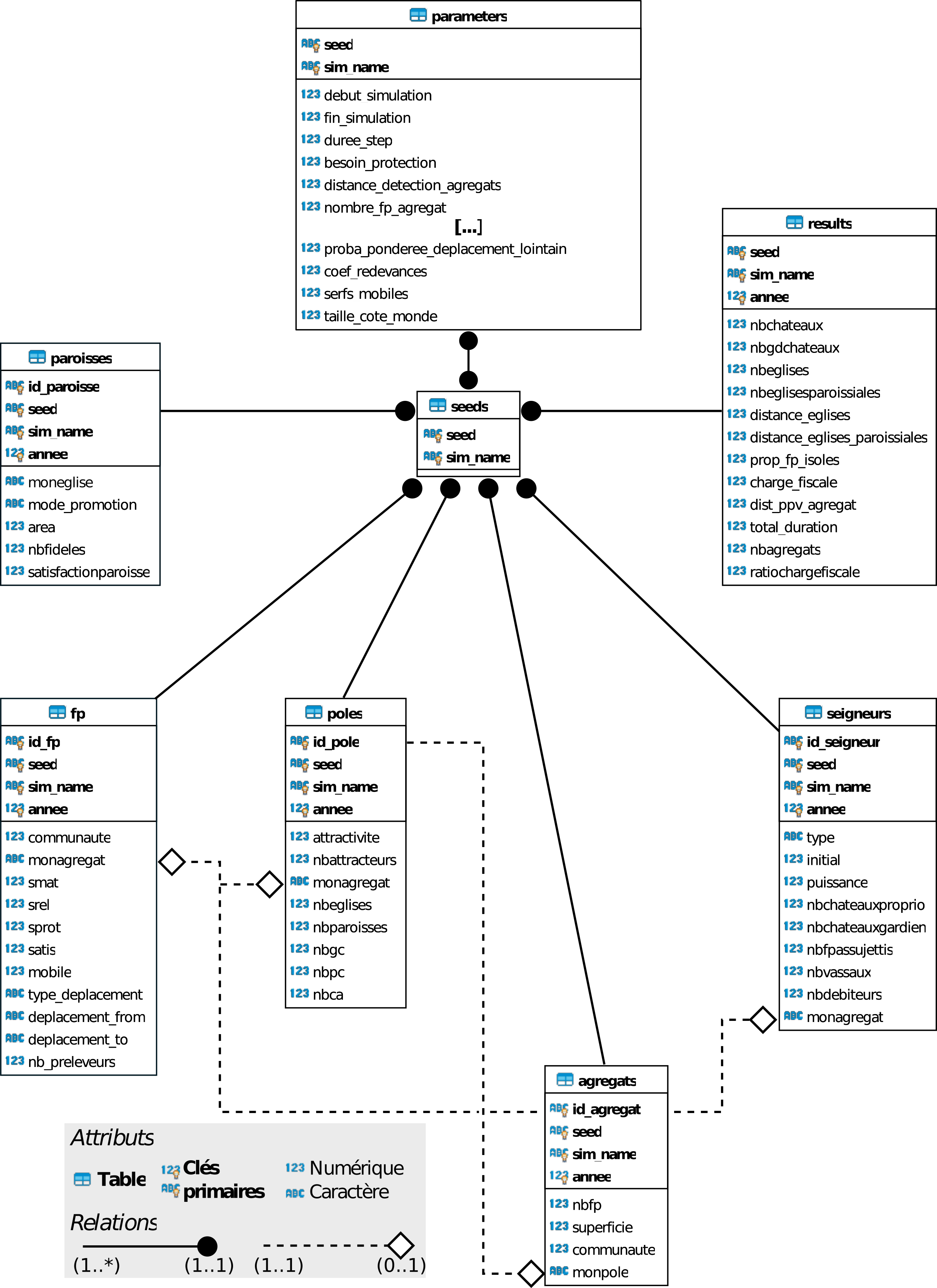
"qui sont par nature chronophages"

***Lucigares du pharaon***

*2019-11-15 18:12:37*

--------------------------------------------

appel de la figure 5.11



Chapitre 5

5.3.

ORGANISER LES DONNÉES

Figure 5.11 – Modèle Conceptuel de Données (MCD) des données en sortie de

simulation de SimFeodal telles qu’implémentées dans SimEDB.

Au centre de cette étoile (voir figure 5.11), il était donc évident de disposer une

table simple, contenant les informations sur lesquelles une majorité des agréga- tions seraient effectuées : les simulations, identifiées par leur nom (sim\_name, qui permet de savoir de quelle expérience ces simulations dépendent) et leur identifiant unique, la graine aléatoire utilisée (seed) 47.

47. La graine aléatoire ne constitue en tant que tel pas un identifiant unique : comme son nom l’indique, elle est aléatoire et présente donc un risque de répétition. Dans Gama, cette graine aléatoire est une valeur qui varie de 0 à 1 et est composé de 19 décimales. Il y a donc potentiellement 1019 graines aléatoires uniques, ce qui est en soi une quasi garantie d’unicité. Notons de plus que dans le MCD de SimEDB, la graine aléatoire est systématiquement associée

52

***Lucigares du pharaon***

*2019-11-15 18:13:01*

--------------------------------------------

peut-être à supprimer

***lue***

*2019-11-21 15:49:07*

--------------------------------------------

"nous avons disposé" : temps (dans la suite tu parles au présent) + le "évident" me gène

***Lucigares du pharaon***

*2019-11-22 16:44:42*

--------------------------------------------

"sont"

Chapitre 5

5.3. ORGANISER LES DONNÉES

**Relier les tables.** Toutes les tables contenant les enregistrements indivi-

duels des agents (fp pour les foyers paysans, paroisses pour les églises parois- siales, etc.) sont donc liées directement à cette table centrale (intitulée seeds ici).

En dehors de ces tables liées aux agents, deux autres tables « globales » sont pré-

sentes : une table « results », qui contient des informations agrégées sur l’état de chaque simulation à chaque pas de temps. Ces informations, par exemple le taux de foyers paysans isolés (champ « prop\_fp\_isoles »), sont redondantes : elles pourraient être calculées directement depuis la table renseignant les foyers paysans, en faisant un ratio entre le nombre de foyers paysans sans agrégat et leur nombre total. Pourtant, pour des raisons d’efficacité autant que de clarté, il a été choisi de dupliquer, en les pré-calculant, ces informations qui sont in- terrogées extrêmement souvent pour calculer les indicateurs de SimFeodal.

Autre table ne répondant pas au schéma classique, la table « parameters » four- nit toutes les méta-données sur les simulations. On y retrouve par exemple les valeurs de paramètres de chacune des simulations, identifiées toujours par le couple sim\_name et seed. Cette table est la seule à être reliée de manière bi- directionnelle à la table centrale (seeds), en particulier en raison de l’usage qui en est fait interactivement (voir l’encadré 5.2).

Notons tout de même que l’on s’éloigne légèrement du classique schéma en

étoile en raison des relations que nous avons choisi d’insérer entre les tables des différents agents (relations notées en pointillées dans la figure 5.11). Intégrer ces relations dans la table centrale aurait considérablement complexifié cette dernière, mais pour autant, elles étaient nécessaires : SimFeodal est un modèle complexe, dans lequel des interactions sont présentes à plusieurs niveaux entre différents types d’agents. La base de données résultant de ce modèle complexe l’est donc nécessairement aussi : on doit implémenter, dans la base de données, des relations entre les tables pour chacune des interactions entre les agents du modèle. ~~I ci,~~ ces relations permettent par exemple d’étudier la composition des pôles autour de chaque agrégat, et ainsi d’étudier le lien entre poids du pôle (en nombre d’attracteurs) et poids de l’agrégat (en nombre de foyers paysans).

Ces indicateurs, situés à l’intersection de différents types d’agents, sont tou-

tefois moins utilisé que les indicateurs plus directs (

teurs). Les requêtes correspondantes, moins fréquentes, ne perturbent pas les

logiques et performances d’ensemble de SimEDB : elles auraient plus facile- ment exprimées dans un schéma « en flocons », mais leur relative rareté ne remet aucunement en cause l’organisation générale du MCD.

**5.3.2.2 Un modèle de données pour favoriser l’interrogation et le fil-**

**trage conjoint**

Le schéma choisi et le Modèle Conceptuel de Données (MCD) associé, per- mettent une interrogation rapide des données en simplifiant les tâches d’agré-

au nom de l’expérience. Même en menant un million de réplications, la probabilité que deux simulations partagent la même graine aléatoire serait largement inférieure à 1%. La graine aléatoire constitue donc un identifiant unique robuste dans notre cas.

53

***Lucigares du pharaon***

*2019-11-22 16:44:38*

--------------------------------------------

"plus" ?

***Lucigares du pharaon***

*2019-11-22 17:19:39*

--------------------------------------------

"l'usage interactif qui est en fait" ?

***Lucigares du pharaon***

*2019-11-22 17:20:28*

--------------------------------------------

je dirais plutôt "là encore" pour appuyer sur le fait que c'est redondant dans cette nouvelle table

***Lucigares du pharaon***

*2019-11-22 17:20:49*

--------------------------------------------

"choisies"

***Lucigares du pharaon***

*2019-11-22 17:21:18*

--------------------------------------------

"figurées"

***Lucigares du pharaon***

*2019-11-22 17:47:12*

--------------------------------------------

"d'examiner" ? pour éviter la répétition (mais pas important)

ref à chapitre 3, indica-

Chapitre 5

5.3. ORGANISER LES DONNÉES

gation et en minimisant la quantité de jointures nécessaires à la génération des

indicateurs de sortie. Le choix de s’écarter légèrement du schéma en étoile pré- sente un autre avantage, extrêmement utile, dans le cadre d’une exploration interactive des indicateurs de SimFeodal. En effet, comme on l’a vu ~~a uparavant~~  (section 5.2.6), dans SimEDB, on compare les simulations en les isolant à partir des valeurs de paramètres qui leur correspondent, via un acte de *brushing* des

~~valeurs de paramètres présentées~~ dans un graphique en coordonnées parallèles interactif. Du côté du MCD, la table correspondante est la table parameters. Quand l’utilisateur sélectionne un sous-ensemble de valeurs de paramètres, la table est filtrée, et ne renvoie donc que les simulations correspondantes.

C’est ici que l’intérêt de la table seeds et de son lien bidirectionnel avec la table parameters apparaît : une fois parameters filtrée, cette sélection est renvoyée à la table seeds, et se répercute donc directement à toutes les autres tables. Avec une unique requête, qui plus est sur une table de faible dimension (seeds ne comporte que deux champs), le filtrage est donc extrêmement véloce, accé- lérant d’autant le filtrage des autres tables et donc la génération des indicateurs de sortie. Ces étapes de filtrage successifs, optimisées par l’architecture choisie pour les données de SimFeodal, sont présenté dans l’encadré 5.2.

54

***Lucigares du pharaon***

*2019-11-22 17:56:22*

--------------------------------------------

"De plus"

***Lucigares du pharaon***

*2019-11-22 17:56:53*

--------------------------------------------

tu viens tout juste de l'expliquer c donc un peu redondant

***Lucigares du pharaon***

*2019-11-22 17:57:22*

--------------------------------------------

pas de saut de paragraphe



Chapitre 5

5.3. ORGANISER LES DONNÉES

*Encadré 5.2 : Un exemple d’interrogation de la base de données de SimEDB.*

La figure 5.12 présente l’ensemble des étapes qui permettent de générer un indicateur de sortie. Cette planche montre un exemple de sélection faite dans l’application SimEDB, et décrit la manière dont cette sélection est répercutée à travers le MCD de SimFeodal (figure 5.11). La démarche aboutit par la sélection d’un ensemble de données, qui répondent à un critère sur deux paramètres du modèle. Cette sélection est ensuite uti-

lisée pour générer un indicateur

d’agrégats au cours du temps.

de

sortie,

ici,

l’évolution du nombre

**(A)**

5

**(B)**

5

**(C)**

true

true

100m

100m

6

6

150m

150m

7

**AND**

8

200m

200m

9

10

250m

distance\_ detection\_ agregats

false serfs\_ mobiles

250m

false

nombre\_ fp\_agregat

nombre\_ fp\_agregat

distance\_ detection\_ agregats

serfs\_ mobiles

**(D)**

- Depuis un ensemble d'expériences aux valeurs de paramètres variées **(A)**,

- l'utilisateur mène une sélection graphique en e�ectuant un brushing des valeurs voulues **(B)**.

- Cette sélection graphique est alors formalisée dans le langage de requet̂ e SQL **(C)**.

**(E)**

- Cette requet̂ e est exécutée sur la table *parameters*.

- On isole alors les attributs *seed* et *sim\_name* de cette table **(D)**.

- Ce �ltrage est répercuté sur la table des *seeds*, qui ne comprend plus à ce moment que le sous-ensemble des *seed* et *sim\_name* correspondant à la requet̂ e de l'utilisateur **(E)**.

- Le �ltrage e�ectué sur la table *seeds* peut alors et̂ re répercuté sur les di�érentes tables liées aux agents.

- Ici, on utilise cette sous-sélection sur la table des *agregats* **(F)**,

- ce qui permet par exemple de compter de nombre moyen d'agrégats à chaque pas de temps et de produire l'indicateur de sortie de simulation souhaité **(G)**.

- Pour produire n'importe quel autre indicateur depuis cette meme sélection, il su�ra de reporter le �ltrage de la table *seeds* dans les tables de chaque type d'agents.

**(G)**

Évolution du nombre d'agrégats

Variabilité : Réplications

200

150

100

50

820 840 860 880 900 920 940 960 980 1000 1020 1040 1060 1080 1100 1120 1140 1160 1180 1200

Temps

Paramètres de la sélection : nombre\_fp\_agregat : [6 ; 7] serfs\_mobiles : ['false']

Figure 5.12 – De la sélection interactive à l’indicateur de sortie.

55

Nombre d'agrégats

***Lucigares du pharaon***

*2019-11-22 17:58:45*

--------------------------------------------

? "Aboutit à " ?

***Lucigares du pharaon***

*2019-11-22 17:59:54*

--------------------------------------------

pas fou d'avoir "d" ici, même si je comprend tes pbm de place sur cette figure... POurquoi ne pas mettre D, E et F sur la même colonne à gauche ou à droite ?

***Lucigares du pharaon***

*2019-11-22 18:03:08*

--------------------------------------------

"pour les lignes correspondant à la sélection"

***Lucigares du pharaon***

*2019-11-22 18:03:55*

--------------------------------------------

"le"

●

●

●

●

●

●

●

●

●

●

●

●

●

●

●

●

●

●

●

●

**fp**

**seigneurs**

[...]

[...]

**poles**

**agregats**

[...]

**id\_agregat seed sim\_name annee**

**(F)**

**parameters**

**seed sim\_name**

[...]

distance\_detection\_agregats nombre\_fp\_agregat

[...]

serfs\_mobiles

**seeds**

**paroisses**

**results**

**seed sim\_name**

[...]

[...]

**parameters**

**seed sim\_name**

[...]

distance\_detection\_agregats nombre\_fp\_agregat

[...]

serfs\_mobiles

**seeds**

**seed sim\_name**

**parameters**

**seed sim\_name**

[...]

distance\_detection\_agregats nombre\_fp\_agregat

[...]

serfs\_mobiles

**serfs\_mobiles IN 'false'**

**nombre\_fp\_agregat ⩾ 6 and**

**nombre\_fp\_agregat ⩽ 7**

7

8

9

10

Chapitre 5

5.3. ORGANISER LES DONNÉES

**Une organisation dédiée à l’exploration interactive**

La présentation des choix d’organisation de données témoigne d’une visée réso- lument applicative, c’est-à-dire visant à penser l’organisation, la structuration et les SGBD d’implémentation, comme au service de la plate-forme d’explo- ration SimEDB. Le SGBD choisi, MapD, est ainsi un logiciel particulièrement adapté aux besoins identifiés, c’est-à-dire à une efficacité et une robustesse d’interrogation des données générées par SimFeodal. MapD est interrogeable de manière universelle, via des protocoles de connexion standards, au moyen d’un langage qui fait office de *lingua franca* de l’interrogation de données, le SQL. Au sein du SGBD, la structure des données, révélée dans le MCD qui adopte une structure « en étoile », vise aussi à faciliter et à optimiser la vitesse des requêtes visant à générer les indicateurs de sortie de SimFeodal. Cette structure de données est enfin pensée, en amont, pour minimiser le nombre de requêtes nécessaires à l’affichage des indicateurs, dans un cadre interactif, correspondant à des sous-ensembles des nombreuses simulations effectuées au cours de la construction, du paramétrage et de la calibration de SimFeodal.

Il est important de noter qu’en l’absence de ces choix de conception de base de

données, de la modélisation conceptuelle jusqu’à l’implémentation technique, la plate-forme d’exploration des données SimEDB, que nous allons maintenant présenter plus en détail, n’aurait pu être conçue, élaborée et bâtie de manière convaincante.

56

***Lucigares du pharaon***

*2019-11-22 18:10:47*

--------------------------------------------

Comme nous venons de le présenter, nos choix d'organisation des données ont été fait dans une visée résolument applicative, c'est-à-dire que nous avons pensé l'organisation, la structuration et les SGBD d'implémentation pour les mettre au service de la plateforme..."

***Lucigares du pharaon***

*2019-11-22 18:10:17*

--------------------------------------------

"aux besoins d'ef... et de vitesse..."

***Lucigares du pharaon***

*2019-11-22 18:12:25*

--------------------------------------------

"suivant un MCD" (le "révélé" est très bizarre)

***Lucigares du pharaon***

*2019-11-22 18:13:39*

--------------------------------------------

et de l'exploration/étude aussi tout simplement non ?? La ça fait comme si c'était important que pour la phase "technique" de construction du modèle

5.4. UNE PLATE-FORME D’EXPLORATION DE DONNÉES DE

SIMULATIONS : SIMEDB

Chapitre 5

**5.4**

**Une plate-forme d’exploration de données de**

**simulations : SimEDB**

La section 5.2 (Comment explorer les sorties de SimFeodal ?) a décrit les étapes

successives d’avancement dans l’exploration des données en sortie de SimFeo- dal, depuis l’observation en direct des simulations (« pré-filtrage ») jusqu’au besoin d’une plate-forme permettant l’exploration et la comparaison interac- tive des sorties de simulation. La plate-forme proposée en réponse à ce besoin, SimEDB 48, dans un objectif de généricité et d’adéquation, se devait aussi de répondre à de nombreuses contraintes, aussi bien liées aux possibilités offertes qu’à l’usage qui en serait fait. Dans cette partie, nous nous attacherons donc à présenter les contraintes qui ont guidé la conception de SimEDB, ainsi que les choix, méthodologiques et techniques, qui en ont résulté.

**5.4.1 Contraintes**

**5.4.1.1 Adapter la complexité aux utilisateurs**

Dans le domaine de l’Interface Homme-Machine (IHM), il est courant de consi-

dérer qu’un outil d’analyse et de représentation doit être adapté à un public. La figure 5.13, emblématique de la conception de géovisualisations par Alan Ma- cEachren, replace ~~a insi~~ les types d’usage d’une plate-forme d’exploration se- lon trois axes : le niveau d’expertise des utilisateurs visés (*users*), le niveau d’in- teraction souhaité (*interaction*) et l’objectif poursuivi par la (géo)visualisation (*task*). D’après l’auteur, à un niveau d’expertise de l’utilisateur correspond un unique degré d’interaction et un unique objectif : dans le cube, seule une

« droite » des usages possibles est présente. L’auteur décompose ces usages en quatre types :

—

Pour le grand public (*users* de type *public*), l’objectif est de transmettre

une information simple (*info sharing*). Le niveau d’interaction avec la géo- visualisation doit donc être faible. Il s’agit d’une tâche de présentation (*present*).

Pour un public légèrement plus connaisseur, on peut augmenter le niveau d’interaction. On entre alors dans un but de synthèse (*synthesize*).

En ciblant un niveau encore supérieur d’expertise chez l’utilisateur, et en visant à de la construction de connaissance plus qu’à une transmission de connaissance, on augmente encore le niveau d’interaction. La géovi- sualisation a alors pour but l’analyse (*analyze*).

Au plus haut niveau d’interaction, d’expertise et de recherche, la géovi- sualisation peut servir d’outil d’exploration (*explore*).

—

—

—

48. **Sim**Feodal **E**xploration **D**ash**B**oard, voir la note de bas de page 26, section 5.2.6.

57

***Lucigares du pharaon***

*2019-11-22 18:15:16*

--------------------------------------------

il n'y a nulle part l'adresse de la plateforme... je dirai qu'il faut la mettre au moins 2 fois (et évidemment si tu coupes les chapitre en 2 les 2 fois seront nécessaires) : une fois quand tu en parles en tout premier, pour qu'on puisse tester ces histoires de brushing direct, ça permettra au lecteur de mieux se rendre compte, et une fois ici (pour le lecteur qui commence par cette partie, parce qu'il n'a pas forcément vu que tu parles de SimEDB avant)

***Lucigares du pharaon***

*2019-11-22 18:27:36*

--------------------------------------------

"qui représente" ?



Chapitre 5

5.4.

SIMEDB

Figure 5.13 – « *An update to Cartography³, 10 years af-*

*ter its conception* », par Çöltekin, JanetZko et Fabri- kant 2018, d’après MacEachren et al. 2004, p. 10.

Roth (2015, p. 16) commente cette figure en effectuant une assimilation entre

niveau d’interaction et complexité de l’interface de l’outil de géovisualisation :

« All participants agreed that user expertise requires increased interface com- plexity, as suggested by the Cartography³ framework ».

La plateforme SimEDB est conçue pour être utilisée par des experts thématiques

(l’équipe de modélisation de SimFeodal), avec un objectif clairement inscrit dans la construction de connaissance. A ce titre, et d’après MacEachren, le niveau d’interaction avec l’outil de géovisualisation devrait être élevé (forte complexité de l’interface pour Roth) et ancrer l’usage dans une dimension exploratoire.

**Des utilisateurs hétérogènes mais captifs.** SimEDB est pourtant pensé

à un niveau intermédiaire, entre l’analyse et la synthèse, dans le cube de la fi- gure 5.13. Il ne s’agit ainsi pas d’explorer des données, au sens de MacEachren, qui sous-entend par cette exploration ~~(~~ *~~explore~~*~~)~~ la recherche d’informations dans un jeu de données inconnu de l’utilisateur. Le besoin identifié consiste à permettre aux utilisateurs d’explorer des sorties de simulation à travers des indicateurs déjà pensés et constitués. Il ne s’agit pas de proposer un outil d’ex- ploration de données brutes, permettant de créer à la volée des nouveaux in- dicateurs, via une approche d’exploration « naïve ». Au contraire, l’exploration est guidée par les indicateurs, et la tâche s’apparente plus à de l’analyse de résultats de simulations, voire à de la synthèse des spécificités des résultats is- sus d’expériences différentes. L’objectif de SimEDB s’écarte donc du « modèle » de MacEachren, puisqu’il ne se situe pas sur la « droite » des usages (voir la figure 5.14).

58

***lue***

*2019-11-23 17:32:54*

--------------------------------------------

pas compris. "En assimilant" tout court ?

***lue***

*2019-11-23 17:33:16*

--------------------------------------------

?

***lue***

*2019-11-23 17:43:17*

--------------------------------------------

si tu tiens à garder la parenthèse, tu peux la mettre après "explorer des données" au dessus, mais je pense que la précision est inutile

***lue***

*2019-11-23 17:43:51*

--------------------------------------------

"consiste dans notre cas"

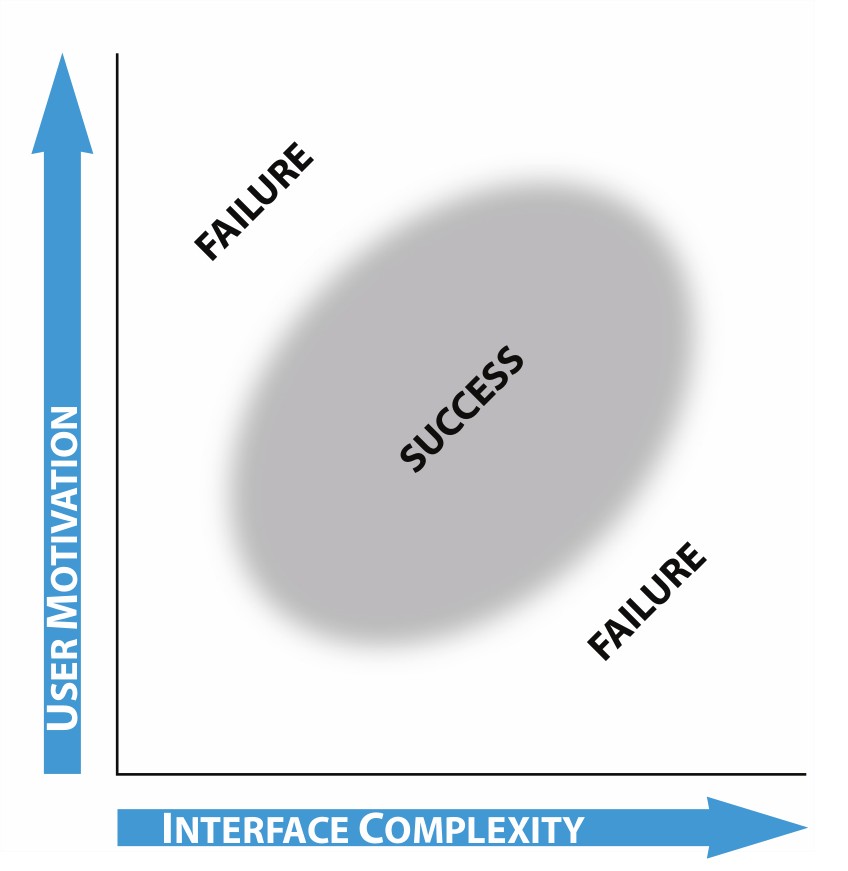
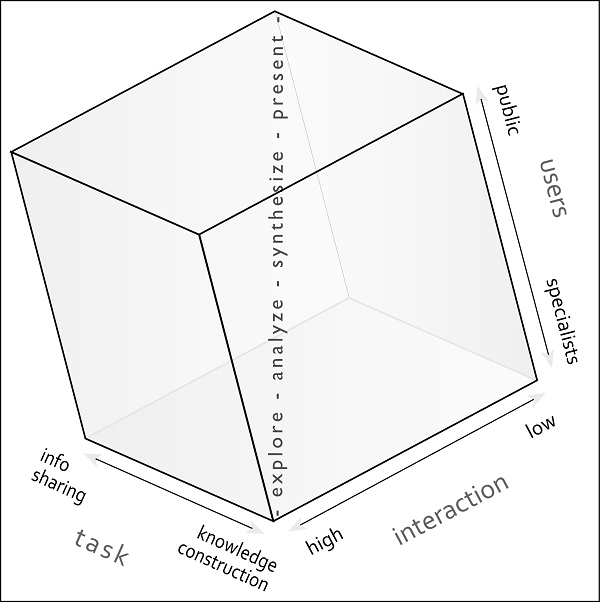
***lue***

*2019-11-23 17:44:40*

--------------------------------------------

"donc"

Sinon ces 2-3 phrases sont un peu répétitives



Chapitre 5

5.4. SIMEDB

Figure 5.14 – Positionnement

de SimEDB dans le cube *Carto- graphy³* de MacEachren.

Figure 5.15 – « *Interface com-*

*plexity versus user motivation.* », Roth 2013, p. 79.

Cet écart au modèle conceptuel de MacEachren s’explique notamment par la

diversité des utilisateurs de SimEDB. Il serait absurde de qualifier un niveau d’expertise général des utilisateurs tant les spécificités de cette expertise sont nombreuses. Entre des profils de spécialiste thématiciens, de modélisateurs ou encore de géomaticiens, l’expertise est présente, mais concernant des champs différents, toutefois tous intéressés par l’exploration des sorties de SimFeodal.

Il est dès lors peu évident de se fixer sur un degré de complexité à atteindre

dans la plate-forme d’exploration : un niveau faible serait frustrant pour les utilisateurs avancés, et un niveau avancé serait source de confusion et donc de perte de motivation pour les utilisateurs moins expérimentés (figure 5.15).

Une spécificité du cas d’usage de SimEDB permet toutefois de miser sur une bonne motivation générale des utilisateurs, et donc sur la possibilité de créer un outil à l’interaction plus complexe qu’un simple présentoir de données. Contrai- rement à une utilisation grand public, qui ne présente aucun engagement vis- à-vis d’une interface d’exploration de données, ou à l’inverse contrairement à des domaines experts où chaque utilisateur dispose de ses propres outils et mé- thodes pour explorer un jeu de données, le public cible de SimEDB est « captif ». On entend par là que les utilisateurs concernés par SimEDB ne disposent pas d’autre solution que de passer par cette plate-forme pour explorer les données issues de sortie de simulation, en particulier en raison des contraintes liées aux caractéristiques de ces données (leur masse par exemple, voir la section 5.1.4, Des données aux indicateurs). On peut dès lors se permettre de développer une interface plus complexe que si l’on visait un plus large public.

**Intuititivé de l’usage au regard des applications traditionnelles.** En

dépit de cette motivation, les utilisateurs de SimEDB demeurent majoritaire- ment des experts thématiciens, potentiellement peu familiarisés à l’exploration de données interactives. Afin que le temps d’exploration des données issues de SimFeodal soit dévoué à la compréhension et à la synthèse de ces données plu- tôt qu’à un apprentissage ou amélioration en exploration de données, il a été choisi de créer une application aussi simple que possible au regard des fonc- 59

***lue***

*2019-11-23 17:45:48*

--------------------------------------------

"concernent"

***lue***

*2019-11-23 17:47:12*

--------------------------------------------

besoin de redonner le titre de la partie ? Tu ne l'as pas fait jusque là...

***lue***

*2019-11-23 17:50:05*

--------------------------------------------

?? quelle motivation ?

"En dépit de ce premier constat?"

Chapitre 5

5.4. SIMEDB

tionnalités principales qu’elle devait

permettre : observer les indicateurs de

sortie de simulation pour des expériences données, et les comparer entre elles

aussi efficacement que possible.

Il n’était donc pas question de construire un nouveau « logiciel expert », doté

de dizaines de fonctionnalités avancées, mais au contraire, de simplifier au maximum l’interface pour ne pas encombrer et complexifier l’utilisation de ces fonctionnalités principales.

On souhaitait une plate-forme aussi épurée que possible, plutôt que de partir

sur la personnalisation et l’adaptation d’une solution existante, nécessairement générique et donc difficile à spécialiser.

**5.4.1.2 Efficacité**

Dans la description du choix du SGBD, on a mentionné une première fois l’in-

térêt de disposer d’une solution d’interrogation de données qui garantisse une certaine rapidité dans l’exécution des requêtes. Sans entrer dans le détail des recherches en IHM, on peut compléter ce besoin de rapidité par deux aspects complémentaires. Une solution interactive qui minimise les latences permet

(1) de motiver l’utilisateur, c’est-à-dire de ne pas le décourager d’utiliser l’ap- plication, et (2) de lui faire conserver sa concentration 49 (*focus*).

Le premier point a été abordé plus haut (section 5.3.1.2), et surtout, en raison de la « captivité » de l’utilisateur évoquée ci-dessus, ne s’applique que margina- lement à notre cas d’étude. Des délais trop importants pourraient décourager l’utilisateur, mais en l’absence d’alternative pour explorer les sorties de simu- lation, cela n’a pas un impact trop important.

**Conserver la concentration.** Le problème de la concentration de l’utili-

sateur demeure, lui, critique. Des études ont montré, depuis longtemps (Mac- KenZie et Ware 1993), qu’il y avait un lien fort entre la performance d’une interrogation visuelle et le délai nécessaire à son obtention. Liu et Heer 2014,

p. 8 montrent ainsi qu’avec un simple délai de 500 millisecondes (ms), la qua- lité des observations, des généralisations qui peuvent être tirées des données, et des hypothèses émises, décroît nettement chez l’utilisateur. Les auteurs in- diquent d’ailleurs que cette diminution est plus importante encore quand l’ex- ploration est effectuée par des actions de *brushing* et de sélections croisées (*linking*), deux méthodes qui sont au cœur de SimEDB : « For example, more aggressive caching or prefetching methods may be employed for operations sensitive to small variations in latency, such as brushing and linking » (ibid., p. 9).

Forch et al. 2017, pour leur part, étudient la perception du délai de réponse

49. Avant de spécifier ce sujet, notons que quand le délai entre une interaction avec un outil informatique et le retour qu’il doit produire (affichage de graphique par exemple) est important, l’utilisateur perd en capacité d’association entre son action et le retour observé. Typiquement, dans un processus d’exploration de données, plus ce délai est faible, plus l’utili- sateur peut mobiliser son intuition pour évaluer, par exemple, les relations entre des variables ou des individus.

60

***lue***

*2019-11-23 17:51:13*

--------------------------------------------

pas de saut de paragraphe + petit mot de liaison ? "Partant" ?

***lue***

*2019-11-23 17:52:52*

--------------------------------------------

j'ai l'impression qu'il y a ici deux idées différentes qui ne sont pas nécessairement contradictoire (ou alors c'est trop implicite) : 1/ faire une plateforme épurée 2/ faire une plateforme spécialisée pour SimFeodal Donc à revoir en 2 phrases ou mieux expliciter pourquoi tes 2 parties de phrases s'opposent...

***lue***

*2019-11-23 17:58:58*

--------------------------------------------

"discuter de ce sujet" ?

***lue***

*2019-11-23 17:59:59*

--------------------------------------------

" : la motivation et la concentration. Une solution etc."

***lue***

*2019-11-23 18:00:26*

--------------------------------------------

"du point de vue de l'expérience utilisateur"

***lue***

*2019-11-23 18:01:27*

--------------------------------------------

parenthèse ? c'est bizarre comme ça au milieu de te phrase

***lue***

*2019-11-23 18:07:36*

--------------------------------------------

je trouve cette note un peu bizarre car j'ai l'impression qu'en termes d'argumentation elle est sur le même plan que ce que tu expliques dans le texte... en fait pq l'avoir mis en "avant de rentrer dans le sujet", car c'est exactement ce que tu décris quand tu parles de la concentration dans les visuals analytics, et en particulier avec tes citations qui parlent explicitement de "cognitive cost", de "train of thoughts", etc.

Chapitre 5

5.4. SIMEDB

lors d’interactions menées avec une souris d’ordinateur. Ils concluent ~~a insi~~ que

les utilisateurs perçoivent des délais d’attente inférieurs à 100 ms, mais notent que les utilisateurs n’en sont pas pour autant perturbés, en particulier ceux qui ont le moins l’habitude de réactions rapides 50.

Concernant le champ, plus spécifique, des *visual analytics*, nous n’avons pas trouvé d’articles de référence permettant d’établir une comparaison de l’effi- cacité des résultats trouvés selon la latence de la réponse. Les auteurs de ce champ recommandent largement de prêter attention à la rapidité de rendu et à son optimisation :

« When simple pattern finding is needed, the importance of having a fast, highly interactive interface cannot be emphasized enough. If a navigation technique is slow, then the cognitive costs can be much greater than just the amount of time lost, because an entire train of thought can become disrupted by the loss of the contents of both visual and nonvisual working memories. »

Ware 2012, tiré de Amirpour Amraii 2018, p. 12.

De manière plus précise, on retrouve une quantification, en termes d’ordres de grandeur, des délais acceptables dans un processus d’exploration de données :

« However, due to human cognitive constraints, exploration needs highly res- ponsive system response times […] : at 500 ms, users change their querying behavior ; past five or ten seconds, users abandon tasks or lose attention » (Fe- kete et al. 2019, p. 2). Ces auteurs font référence à un billet de blog de Jakob Nielsen qui donne une justification plus poussée en matière de capacité de concentration.

« When the computer takes more than 0.1 second but less than 1 second to respond to your input, **it feels like the computer is cau- sing the result** to appear. Although users notice the short delay, they **stay focused on their current train of thought** during the one-second interval.

[…]

After 1 second, users get impatient and notice that they’re waiting for a slow computer to respond. The longer the wait, the more this impatience grows ; after about 10 seconds, the average attention span is maxed out. At that point, the user’s mind starts wandering and doesn’t retain enough information in short-term memory to ea- sily resume the interaction once the computer finally loads the next screen.

More than 10 seconds, and you **break the flow**. Users will often **leave the site** rather than trying to regain the groove once they’ve started thinking about other things. »

Nielsen 2009.

Selon ces différentes considérations, dans le cadre de SimEDB, on doit donc viser à développer une plate-forme aussi rapide que possible. Celle-ci doit donc

50. Ils remarquent ainsi que les utilisateurs plus habitués à des jeux vidéos rapides (« highly dynamic computer games, such as action games, racing games, or first person shooter games [...] », Forch et al. 2017, p. 51) sont plus vite affectés par le délai de réponse que les autres.

61

***lue***

*2019-11-23 18:04:38*

--------------------------------------------

coool :D

***lue***

*2019-11-23 18:05:11*

--------------------------------------------

"néanmoins"

***lue***

*2019-11-23 18:08:11*

--------------------------------------------

le gras ce toi qui le rajoute ? il faut préciser que ce soit le cas ou pas

***lue***

*2019-11-23 18:09:46*

--------------------------------------------

supprimer l'un des deux "donc" (je dirai plutôt le premier... mais même peut-être les deux)

***lue***

*2019-11-23 18:09:53*

--------------------------------------------

"Suivant" ?

Chapitre 5

5.4. SIMEDB

viser des temps de latence maximale de 10 secondes, tout en sachant, dès le

départ, qu’il sera impossible d’arriver aux délais de 100 ms ou 500 ms évoqués précédemment, ne serait-ce que parce que le temps de requête des données – sans compter le temps de rendu graphique – est déjà supérieur d’un ordre de grandeur.

**5.4.1.3 Interopérabilité et évolutivité**

Une autre contrainte forte tient cette fois au choix de l’environnement infor- matique qui accueillera la plate-forme d’exploration. On peut résumer ce choix à deux alternatives : un environnement local, en installant l’application sur l’ordinateur de chaque utilisateur, ou un environnement distant, où l’applica- tion serait donc accessible à distance, par exemple via une interface web 51. Ce choix a des nombreuses répercussions, aussi bien en matière de possibilité d’accès que de facilité à faire évoluer la plate-forme.

Jusqu’à récemment, le choix le plus classique était de développer une appli-

cation installable sur un ordinateur : cela permet de garantir une utilisation à tout moment, sans contrainte d’accès au réseau internet. Cela permet aussi d’obtenir de meilleurs performances, puisque la rapidité de l’application dé- pend à ce moment uniquement de la puissance de l’ordinateur, plutôt que de devoir souffrir du passage par l’intermédiaire d’un serveur.

Comme pour le choix du type de SGBD, nous avons cependant préféré nous

orienter sur une solution de type distante, pour des raisons d’interopérabilité et d’évolutivité ~~q ue nous allons décrivons ici.~~

**Différents supports d’interrogation.** La performance d’une application

locale, par rapport à une application distante, est un atout extrêmement in- téressant, ~~c omme on vient de le montrer plus haut.~~ Pourtant, cela implique une énorme contrainte : l’application doit être interopérable entre les diffé- rents systèmes d’exploitations (*Operating System*, OS) et versions de ceux-ci. Les utilisateurs potentiels de SimEDB, représentation fidèle des acteurs de la recherche, se partagent ainsi entre les trois systèmes d’exploitations majori- taires (Windows, MacOs, Linux).

Pour permettre à chacun d’utiliser SimEDB, il faudrait donc que le dévelop-

pement de cette plate-forme soit compatible avec ces différents OS, ce qui est une contrainte considérable en développement logiciel.

Ne mentionnons même pas les nouveaux OS, centrés autour d’usages tactiles, tels qu’on les retrouve sur les tablettes et autres *smartphones*, qui demandent, eux aussi, de nombreuses spécificités de développement.

En somme, disposer d’une application locale universelle, c’est-à-dire utilisable

quelque soit le support informatique, est une quasi-impossibilité technique, et

51. Cette question était également posé pour le choix du SGBD en section 5.3.1.1. L’outil d’exploration et l’architecture de données sont cependant indépendants, et le choix d’un sto- ckage des données sur un serveur distant n’implique aucunement que l’application suive la même logique. On peut ainsi avoir un SGBD distant qui serait interrogé par une application locale.

62

***lue***

*2019-11-23 18:13:58*

--------------------------------------------

inverser la phrase ça sera plus explicite pour expliquer le choix :

"Dans notre cas, du fait du temps nécessaires de requête des données qui est (hors temps d'affichage et hors plateforme) déjà important, il nous sera impossible d'arriver aux délais de 100 ms ou 500 ms évoqués dans la littérature. Nous avons donc choisi de viser en temps de latence maximale de 10 secondes, qui correspond à des performances acceptables au regard des calculs des affichages à effectuer (ou un truc du genre)"

***lue***

*2019-11-23 23:16:02*

--------------------------------------------

parfait !

***lue***

*2019-11-23 23:22:34*

--------------------------------------------

tu viens JUSTE de le dire une phrase au dessus, pas besoin d'y faire une référence comme ça. Je dirai plutôt :

"Si une application locale offre des performances plus intéressantes qu'une application distance, elle implique également une énorme contrainte : "

***lue***

*2019-11-23 23:22:56*

--------------------------------------------

"et les différentes versions"

***lue***

*2019-11-23 23:23:50*

--------------------------------------------

je mettrai ça en note "A ce titre, ils sont une bonne représentation des acteurs/usages de la recherche"

***lue***

*2019-11-23 23:24:15*

--------------------------------------------

Pas de retour à la ligne

***lue***

*2019-11-23 23:30:02*

--------------------------------------------

Pas de retour à la ligne + phrase sans sujet > si tu veux garder cette idée rajoute un "nous" au début. Mais je pense que tu devrais enlever cette phrase en l'état, je trouve que ça ne rajoute pas grand chose à l'argumentation, que tu construits autour du besoins de tes utilisateurs, et dont tu ne dit pas qu'ils ont ces usages tactiles... Ou alors : "Dans le meilleur des cas, et pour couvrir l'ensemble des utilisations possibles, on pourrait considérer également d'autres OS, en particulier ceux centrés autours d'usages tactiles tels qu'on les retrouvent sur les tablettes et autres smartphones, qui demandent eu aussi de nombreuses spécificités de développement".

Chapitre 5

5.4. SIMEDB

un objectif en soit, que notre travail de recherche ne cherche aucunement à ré-

soudre. Pour garantir la faisabilité d’une plate-forme d’exploration de données locale dédiée aux données de simulation de SimFeodal, il faudrait donc com- mencer par restreindre son champ d’application à un ou deux supports officiels, par exemple l’OS Windows, abandonnant de fait les utilisateurs potentiels ne disposant pas de cette architecture logicielle.

**Gérer les mises à jours et modifications.** Comme pour les bases de don-

nées (« Stockage distribué ou centralisé », section 5.3.1.1), la question de l’ap- plication locale ou distante pose un contrainte supplémentaire en matière de maintenabilité et d’évolutivité de la plate-forme choisie. Dans le cadre d’une application locale ~~( correspondant au distribué en SGBD),~~ la distribution des différentes mises à jour de l’application entraînent nécessairement l’installation locale, à chaque fois. Le risque est alors que tous les utilisateurs ne disposent pas d’une même version, ce qui peut entraîner, par exemple, des contradic- tions dans l’évaluation d’expériences, certains utilisateurs ayant accès à une version proposant des différences dans la manière de calculer ou d’afficher les indicateurs.

Sans aller jusqu’à ces extrêmes, notons qu’avec une application locale, le temps

de répercussion d’une modification du code de la plate-forme est plus impor- tant : il faut en effet réinstaller sur chaque poste le logiciel ainsi modifié. Cela disqualifie de fait des modifications « en direct », par exemple lors d’une session collective d’exploration des résultats où les utilisateurs auraient des proposi- tions de modifications à faire, ne serait-ce que pour des changements aussi infimes que des titres de graphiques ou d’axes.

**Le choix d’une application web.** Au contraire, avec une application dis-

tante, donc basée sur l’accès, par un navigateur internet, à une application centralisée, ces problèmes ne se posent pas : des navigateurs sont disponibles pour tous les OS existants (OS dédiés aux ordinateurs ou aux usages mobiles), et interprètent de la même manière une page web, indépendamment de leur sup- port de consultation. De plus, comme pour les SGBD, l’usage d’une plate-forme distante permet une répercussion instantanée des mises à jour et corrections : un utilisateur n’a qu’à rafraichir sa page pour que la dernière version de l’ap- plication s’affiche. De la même manière, si un utilisateur souhaite étudier un nouvel indicateur, non prévu auparavant, le temps de déploiement peut être suffisamment court pour que cela soit possible au cours d’une même session d’exploration de données.

Il y a toutefois un désavantage vis-à-vis de solutions entièrement locales, puisque

les données permettant l’affichage des indicateurs doivent transiter sur le ré- seau internet. En cas de connexion lente, l’usage de l’application sera particu- lièrement difficile, et même impossible en l’absence d’une connexion.

Cette lenteur relative est toutefois compensée par un avantage de la centra-

lisation de l’application : les calculs, parfois lourds, ne reposent pas sur les capacités individuelles des ordinateurs clients. En installant l’application sur un serveur dédié, il suffit donc d’augmenter les caractéristiques de celui-ci

63

***lue***

*2019-11-23 23:30:50*

--------------------------------------------

ici je préfèrerais "nous aurions du"

***lue***

*2019-11-23 23:31:38*

--------------------------------------------

"une réinstallation " ?

***lue***

*2019-11-23 23:31:54*

--------------------------------------------

"de la même" ?

***lue***

*2019-11-23 23:36:42*

--------------------------------------------

si tu supprimes plus haut la partie ou tu parles des usages tactiles, tu peux réintroduire cette idée ici en note, en mode (et donc supprimer la parenthèse) :

"Une plateforme distante disponible à partir d'un navigateur internet à en outre l'avantage de pouvoir être utilisée sur des OS mobiles (tablettes, smartphones, etc.), permettant notamment des usages tactiles - des fonctionnalités qu'il aurait là aussi fallut développer spécifiquement dans le cadre d'une solution locale)"

***lue***

*2019-11-23 23:37:30*

--------------------------------------------

d'une seule / d'une session unique

***lue***

*2019-11-23 23:38:02*

--------------------------------------------

capacités ?

***lue***

*2019-11-23 23:39:45*

--------------------------------------------

l'exemple était plus clair au dessus avec la séance de travail collaborative... là on ne sait plus trop : qui développe et déploie le nouvel indicateur ? rajouter qqpart un "nous" pour qu'on te replace là dedans.

Chapitre 5

5.4. SIMEDB

pour que les performances soient améliorées pour chacun des utilisateurs de

l’application.

Dans le cas de SimEDB, nous disposons de ressources informatiques largement

suffisantes 52 pour assurer une rapidité de traitement des données et ainsi per- mettre à l’application ~~S imEDB~~ de se dégager de ce « goulot d’étranglement » technique qu’aurait sinon éprouvé~~e~~ la plate-forme.

**5.4.1.4 Généricité de l’interrogation et indépendance vis-à-vis des don-**

**nées**

La dernière contrainte, plus technique, tient au besoin de généricité d’une

plate-forme d’exploration de données vis-à-vis des données qu’elle interroge. On a résumé les possibilités et choix effectués en matière de SGBD (section 5.3.1 :

~~« Assurer la capacité d’interrogation des données »),~~ et décidé de ne retenir que des SGBD permettant une interrogation standardisée via des connecteurs gé- nériques et un langage universel (le SQL).

L’infrastructure de stockage et d’organisation des données a ainsi été conçue

pour être aussi générique que possible. Encore faut-il que la plate-forme d’ex- ploration de données soit elle aussi aussi générique que possible, et donc en mesure de profiter de l’universalité du SGBD choisi.

**Indépendance vis-à-vis du support de données.** Une contrainte forte

est donc constituée par la capacité de la plate-forme a être indépendante de la source des données : quelque soit le SGBD choisi, les requêtes émises par la plate-forme doivent être les mêmes, sans requérir d’adaptations spécifiques en dehors de la désignation du lieu de stockage des données et des pilotes du SGBD.

Dans les faits, lors de la construction de SimEDB (cf. section 5.2 ~~: « Comment~~

~~explorer les sorties de SimFeodal ? »)~~, plusieurs solutions de stockage de don- nées ont été employées successivement, au fur et à mesure des limites rencon- trées chez chacune. Depuis les premières implémentations des rapports auto- matiques jusqu’à l’utilisation de SimEDB dans son état actuellement discuté, les données de sorties de simulation ont ainsi été tour à tour interrogées depuis de simples fichiers CSV ~~a u départ~~ jusqu’au SGBD ultra-performant MapD, en passant par des solutions intermédiaires plus classiques (SQLite et MonetDB notamment).

Il n’était donc aucunement question d’avoir à adapter le code source des pro-

grammes permettant de générer les indicateurs depuis les données, mais au contraire, de s’assurer d’utiliser des bibliothèques logicielles indépendantes des données, c’est-à-dire capables d’exécuter les mêmes chaînes de traitements quelle que soit la provenance des données.

On peut expliciter ce propos à l’aide de l’exemple caricatural des logiciels de

52. En nous appuyant dans un premier temps sur un serveur de calcul interne à l’UMR Géographie-cités, puis sur un serveur de calcul partagé mis à disposition par la « Très Grande Infrastructure de Recherche » Huma-Num ensuite.

64

***lue***

*2019-11-23 23:40:41*

--------------------------------------------

Pas de saut de paragraphe

***lue***

*2019-11-23 23:43:39*

--------------------------------------------

"actuel"

***lue***

*2019-11-23 23:44:19*

--------------------------------------------

"pas" je trouve à plus d'impact

***lue***

*2019-11-23 23:44:39*

--------------------------------------------

ajouter virgule

Chapitre 5

5.4. SIMEDB

type tableurs. Dans ce type d’application, on peut écrire des programmes (les

« macros ») qui permettront l’ouverture d’un fichier CSV et effectueront des calculs dessus pour en tirer par exemple des résumés. Dans ce même tableur, on peut aussi faire appel à des sources de données différentes (bases de données Access ou SQL par exemple), mais les programmes (macros) seront alors à ré- écrire en quasi-totalité pour les adapter aux différences de sources de données.

Dans notre cas, les sources de données ayant très largement évolué au cours

du temps, on ne pouvait faire reposer notre application sur une plateforme qui demande une adaptation forte à la provenance des données, comme c’est le cas des tableurs. Il était donc nécessaire de s’appuyer sur des environnements logiciels (les bibliothèques logicielles) permettant une forte généricité vis-à-vis des sources de données.

**Indépendance vis-à-vis des requêtes et modularité de l’implémenta-**

**tion.**

Pour garantir cette généricité, il est donc nécessaire de s’assurer que

le mode de communication de la plate-forme vers les données soit bien basé

sur un langage universel : le SQL. Il convient donc de choisir un ensemble de technologies permettant de générer des requêtes SQL, quand bien même l’expression de ces requêtes elles-mêmes serait conçue dans un autre langage. Pour les requêtes complexes, le SQL tend en effet a être peu lisible, les opé- rations s’emboîtant les unes dans les autres de manière très linéaires, et donc, souvent verbeuses. En SQL pur, il est donc peu évident de créer une implémen- tation modulaire d’une requête, c’est-à-dire permettant une factorisation des commandes et un paramétrage des entrées.

Les indicateurs de sortie de SimFeodal sont, on l’a vu, assez fréquemment ba-

sés sur le même type d’opération : on observe par exemple souvent l’évolution du nombre d’agents d’un certain type (agrégats, foyers paysans…)au cours du temps. ~~D ans le cas de cet exemple,~~ en SQL, pour spécifier une requête permet- tant de récupérer le nombre de foyers paysans au cours du temps, groupés par année et avec un filtre sur certaines simulations, il ne faut que quelques lignes de code. Pour que cette requête devienne générique, c’est-à-dire indépendante du type d’agent qui en deviendrait un argument, il est nécessaire d’y ajouter de nombreuses lignes de code. Cela revient potentiellement à doubler, pour cha- cun des indicateurs, la longueur du code-source requis pour l’expression des requêtes, et bien sûr à les rendre plus complexe à modifier et corriger. De plus, les modes d’expression qui permettent de modulariser du code SQL peuvent varier fortement selon les SGBD choisis, n’étant pas strictement décrits dans les normes SQL. Par exemple, la déclaration d’une variable, par exemple pour paramétrer le nom de la table contenant les agents, est très différente dans les deux SGBD les plus utilisés (MySQL et PostgreSQL).

Faire appel à un langage intermédiaire, générant du SQL en sortie depuis une

entrée sous forme d’un « *Domain Specific Language* » (DSL) permet ainsi de bé- néficier d’une part de l’universalité du SQL, et d’autre part, d’une syntaxe plus expressive que celle du SQL. En utilisant un DSL, plus adapté à la manipulation de données qu’à la sélection de sous-ensembles, on gagne donc en modularité d’implémentation , et donc en ré-utilisation de fonctions plus génériques, ce 65

***lue***

*2019-11-23 23:45:13*

--------------------------------------------

"résumés statistiques"

***lue***

*2019-11-23 23:53:49*

--------------------------------------------

je pense que tu peux te passer de ces paragraphes, le dernier est très redondant même si je comprends qu'il soit nécessaire après l'exemple. Mais je ne suis pas sûre que l'exemple soit lui-même nécessaire.

***lue***

*2019-11-24 00:00:12*

--------------------------------------------

ajouter espace

***lue***

*2019-11-24 00:00:54*

--------------------------------------------

"par exemple"

***lue***

*2019-11-24 00:01:27*

--------------------------------------------

"néanmoins" ?

***lue***

*2019-11-24 00:02:39*

--------------------------------------------

pas défini > ici peut-être qu'une explicitation serait bienvenue sur la fin de ce paragraphe (factorisation et paramétrage)

***lue***

*2019-11-24 00:03:07*

--------------------------------------------

"Ainsi" (pour éviter répétition)

***lue***

*2019-11-24 00:03:39*

--------------------------------------------

quel est le sujet de ce verbe ?

***lue***

*2019-11-24 00:04:14*

--------------------------------------------

"est une solution qui permet"

***lue***

*2019-11-24 00:05:07*

--------------------------------------------

Peut-être ici "Nous avons donc plutôt choisi de faire appel" pour insister sur toi et sur le choix final (sinon ça fait un peu trop "exposé des options")

***lue***

*2019-11-24 00:07:31*

--------------------------------------------

ici j'enlèverai les virgules parce qu'il ne me semble pas que tu introduits avant explicitement cet équilibre manipulation/sélection, et du coup quand tu le mets entre parenthèse c'est comme si tu l'avais expliqué en ces termes avant... Est-ce que c'est générique à tous les DSl ou propre à celui que tu choisi toi ?

***lue***

*2019-11-24 00:10:13*

--------------------------------------------

Proposition pour éviter les "donc" à répétition :

"et à la sélection de sous ensemble, on gagne en modularités d'implémentation, ce qui permet d'utiliser des fonctions plus génériques adaptables à différentes données en entrée (ou qqchose comme ça d'un peu plus explicite?). Ainsi, le code-source développé est plus robuste, ré-utilisable, et évolutif."

Chapitre 5

5.4. SIMEDB

qui permet de disposer d’un code-source plus robuste, ré-utilisable et évolutif.

**Conclusion : Vers une plate-forme web générique et intuitive**

Dans cette sous-partie, nous avons présenté les principales contraintes qui ont

orienté le choix des cadres méthodologiques et techniques utilisables pour concevoir une plateforme telle que SimEDB.

En premier lieu, on fait le choix de se tourner vers une plate-forme implémen-

tée sous forme d’application web, utilisable depuis un simple navigateur – donc inter-opérable entre les différents supports technologiques –, ce qui exclue de fait quantités d’outils, de logiciels et de bibliothèques logicielles pensées pour l’exploration interactive de données.

On souhaite de plus que la plate-forme utilisée dispose d’une interface aussi

épurée que possible, donc nécessairement très adaptée au cas particulier des données issues de simulation que l’on manipule. Là encore, l’étendue des pos- sibles est restreinte, éliminant l’ensemble de solutions « clefs-en-main », par exemple conçues autour des « webSIG » ou de bibliothèques logicielles de vi- sualisations interactives intégrées.

L’utilisation de la plate-forme doit être aussi efficace que possible, en cher-

chant à minimiser les temps de latence entre sélection interactive et affichage des indicateurs en résultant. On devra donc privilégier des ensembles techno- logiques récents et performants, intrinsèquement dédiés à l’interactivité, au détriment de *frameworks* plus génériques.

Enfin, il faut que cette solution, dans la mesure du possible, soit en mesure de proposer une syntaxes d’interrogation de données modulaire, factorisée, et plus expressive que le SQL sur lequel elle doit toutefois s’appuyer.

Ces contraintes sont des éléments génériques à prendre en compte dans la

conception d’un outil d’exploration de données, et elles dépassent largement notre seul cas d’utilisation. Nous n’avons pour autant pas tenté de brosser un paysage complet des contraintes potentielles, liées aux différents usages possibles, qui peuvent guider les choix techniques et méthodologiques de la conception d’un outil. La relative spécificité de SimEDB tient à la combinaison des contraintes identifiées et à la combinaison des choix effectuée~~s~~ pour les dépasser, que nous allons maintenant expliciter.

66

***lue***

*2019-11-24 00:10:58*

--------------------------------------------

"exclut"

***lue***

*2019-11-24 00:11:28*

--------------------------------------------

encore la relation épurée/adaptée que je ne trouve pas claire/argumentée.

***lue***

*2019-11-24 00:15:56*

--------------------------------------------

"des"

***lue***

*2019-11-24 00:16:16*

--------------------------------------------

"avec un temps de latence minimal"

Chapitre 5

5.4. SIMEDB

**5.4.2**

**Construire une plate-forme interactive pour l’explo-**

**ration de sorties de simulation**

Dans cette dernière sous-partie, nous allons donc présenter les choix – tech-

niques, esthétiques et interactifs – qui ont été adoptés dans la conception et l’implémentation de SimEDB. Nous les présentons ici de manière linéaire, dans l’ordre quasi-chronologique du développement, mais il est important de garder en considération que ces éléments sont intimement intriqués. Un choix tech- nique, par exemple, peut conditionner les types d’interactions possibles, parce que l’utilisation de telle méthode d’interaction peut n’être proposée que dans tels et tels environnements logiciels.

Notons enfin que l’application SimEDB présentée ici, aussi bien dans son usage

que dans sa conception, représente un instantané de développement, qui cor- respond à la période de rédaction du présent chapitre : à l’instar d’un modèle, une plate-forme peut et doit évoluer pour s’adapter aux besoins de ses utilisa- teurs tant qu’elle est utilisée. Les technologies et choix esthétiques introduits n’ont pas toujours été présents, et auront sans doute à évoluer dans la suite de la « durée de vie » de SimEDB. Pour les raisons évoquées en termes de facilité de mise à jour d’une solution distante, cela ne pose toutefois aucun problème vis-à-vis de l’utilisation de la plate-forme, largement indépendante, en matière de temporalités, du présent ouvrage.

**5.4.2.1 Choix des technologies**

Nous présentons ici les technologies mobilisées dans le cadre du développe- ment de SimEDB. Le but n’est pas d’entrer dans les détails de l’implémenta- tion 53, mais bien de justifier et présenter les choix relatifs aux technologies employées, en restant à un niveau assez général 54. Il nous paraît important d’entrer dans ces choix qui relèvent plus de la technique que de la méthodo- logie en ce qu’ils concourent de la volonté de reproductibilité de la thèse, et particulièrement de la reproductibilité de la démarche, conceptuelle et métho- dologique, mise en place. Nous portons la conviction que l’ensemble de tech- nologies assemblées ici dans notre « chaîne de traitement » est très largement ré-utilisable, dans le cadre d’adaptations à d’autres cas d’études, mais aussi et surtout, pour une multitude de problématiques requérant une analyse visuelle

de données massives (

7).

**Technologies webs « natives » et adaptativité.** Au cours de la dernière

décennie, les interfaces physiques de consultation de médias informatiques se sont largement diversifiées. Cela a provoqué une hétérogénéisation importante aussi bien des modes d’interaction (dispositifs « tactiles ») que des modes d’af- fichages (les tailles et résolutions des écrans n’ont jamais été aussi diverses et imprévisibles).

53. Le code source de SimEDB – et l’historique de son versionnement – sont, pour cela, dis- ponibles en ligne sous licence libre, sur la plate-forme Github :  github.com/RCura/SimEDB

54. À ce titre, les quelques lignes de codes présentes par la suite servent un but illustratif et descriptif, et nous semblent remplir ce rôle bien plus efficacement que n’importe quel schéma structurel ne le pourrait.

67

***lue***

*2019-11-24 00:17:28*

--------------------------------------------

"liés"

***lue***

*2019-11-24 00:20:04*

--------------------------------------------

"expliquées" ou pour alléger la phrase :

"Comme nous l'avons expliqué, du fait de la facilité de la mise à jour d'une relation distante, cela ne pose..."

***lue***

*2019-11-24 00:21:16*

--------------------------------------------

à ne pas mettre entre tiret, c'est central ici et pas anectodique !!

***lue***

*2019-11-24 00:21:41*

--------------------------------------------

"et consultables" ?

***lue***

*2019-11-24 17:39:55*

--------------------------------------------

pourquoi entre guillemets ?

on y reviendra dans le chap

Chapitre 5

5.4. SIMEDB

En conséquence, les normes de présentations graphiques ont évolué~~,~~ vers plus

d’« adaptativité », en particulier avec l’avènement du « responsive web design » (« conception de sites web adaptatifs ») qui permet de prévoir efficacement l’agencement d’une page web quelque soit le support de consultation.

Les technologies qui prédominaient dans la réalisation d’applications web in-

teractives il y a quelques années 55 ont largement disparu suite à un manque d’adaptation à ces nouveaux support.

Pendant ce temps, de nouveaux standards du développement web (HTML5

entre autre) ont émergé et atteint un niveau de maturité suffisant pour rem- placer l’ensemble des possibilités (et les étendre) proposées par ces anciens environnements trop monolithiques.

Ces technologies, aujourd’hui indispensables, reposent sur des codes standar-

disés, verbeux et peu explicites 56, mais toutefois assez universellement inter- prétables par les navigateurs. Pour pallier à leur faible expressivité, on peut faire appel à des *frameworks* graphiques qui en simplifient l’usage : comme les DSL évoqués plus haut, ce sont des ensembles de bibliothèques logicielles qui génèrent à l’aide d’instructions courtes et simples les centaines de lignes de codes nécessaires à l’affichage interopérable, universel et constant d’un site ou d’une application web.

Nous avons donc fait le choix de nous concentrer sur des environnements stan-

dardisés, capables de générer du HTML (« *HyperText Markup Language* »), lui- même mise en forme à l’aide de styles CSS (« *Cascading Style Sheets* ») et rendu interactif par du code JavaScript.

À ce titre, le framework Bootstrap 57 s’est révélé extrêmement utile dans le *design* de l’interface de SimEDB (et des versions précédentes), tant il simplifie l’expressivité d’une mise en page à l’aide d’une grille graphique et de compo- sants interactifs ré-utilisables.

**Le choix d’environnements de développement intermédiaires.** 58 Pour construire des applications interactives en lignes, de multiples choix sont pos- sibles, et on peut les catégoriser selon le niveau de développement qu’ils de- mandent. Par exemple, il est tout à fait possible de s’appuyer sur des briques logicielles de bas niveau (ce que l’on appelle communément *framework*), et de développer à partir de celles-ci toute l’interface et le fonctionnement d’une application.

Cette approche, majoritaire dans la construction d’applications actuelles (avec

des *frameworks* basés sur le langage JavaScript tels que ReactJS ou AngularJS,

55. Applications en ~~Flash,~~ *applets* Java…

56. Il suffit de consulter le code-source d’une page web contenant des visualisation interac- tives pour le constater. Les assemblages de langages SVG, CSS et JavaScript sont ainsi assez largement indéchiffrables pour qui n’en est pas un spécialiste.

57. http ://getbootstrap.com/

68

***lue***

*2019-11-24 17:45:26*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-24 17:48:13*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-24 17:48:36*

--------------------------------------------

?

***lue***

*2019-11-24 17:49:57*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-24 17:56:12*

--------------------------------------------

pas de saut de paragraphe

58.

Ne pas oublier, dans le positionnement (chap1) de consacrer au moins un paragraphe

(ou encadré) au choix « militant » de ne se tourner QUE vers des outils libres, sans exception.

Chapitre 5

5.4. SIMEDB

ou encore sur le langage Python tels que Django ou Flask), est extrêmement

flexible et performante, au prix d’un développement important. Un *framework* fournit en effet des « briques » logicielles de base – les composants –, très géné- riques. Ces composants de bases demandent donc une forte personnalisation et un agencement complexe afin d’arriver au résultat souhaité. La communica- tion entre ces composants doit être entièrement prévue et implémentée, et on abouti donc nécessairement sur des projets assez importants, qui demandent une réelle expertise en développement et portent le risque d’être trop com- plexes pour être facilement adaptés et donc rendus génériques.

À l’autre bout du gradient de développement, on peut aussi choisir de bâtir

une application à partir d’un ensemble logiciel intégré, comme Tableau, qui permet d’agencer visuellement et graphiquement des composants graphiques et leurs liens. Ces outils, très usités en informatique décisionnelle, sont extrê- mement simples à prendre en main, y compris pour des « utilisateurs finaux »

– analystes par exemple –. En contre-partie, ils sont moins personnalisables et configurables ~~q ue des solutions plus bas niveau comme les~~ *~~frameworks~~*~~,~~ et ce sont majoritairement des logiciels propriétaires, donc non modifiables.

Entre ces deux extrêmes, quelques *frameworks* intermédiaires, souvent origi-

naires des outils de manipulation de données plus que du monde de l’informa- tique décisionnelle, mettent à disposition de l’utilisateur des composants de plus haut-niveau que les « briques élémentaires ». L’interaction entre les com- posants y est déjà pré-conçue, tout en reposant sur une construction « depuis zéro », donc personnalisable~~s~~ et adaptable~~s.~~

Généralement, chaque *framework* est associé à un langage de programmation (voir tableau 5.2) : le *framework* Shiny s’appuie sur le langage R, Dash sur le langage Python et Escher sur le langage Julia.

2018

D3.js

Tableau 5.2 – Une sélection de différents *frameworks* dédiés à la création d’in-

terfaces d’exploration de données.

Le choix de tel ou tel framework dépend certes de la maturité de chaque projet

– Shiny est à ce titre très en avance –, mais surtout du langage informatique que le concepteur de l’application souhaite utiliser. Dans le cas de SimEDB, le créateur de la plate-forme est adepte du langage R (voir Commenges et al. 2014) et pratique le *framework* Shiny depuis plusieurs années (voir Cura 2015) : le choix d’utiliser ce *framework*, au sein d’un environnement logiciel 69

***lue***

*2019-11-24 18:15:24*

--------------------------------------------

"à"

***lue***

*2019-11-24 18:17:42*

--------------------------------------------

"qui proviennent plus souvent du monde de la manipulation de donnée que de celui de l'informatique décisionnelle"

***lue***

*2019-11-24 18:27:06*

--------------------------------------------

"du"

***lue***

*2019-11-24 18:30:05*

--------------------------------------------

é

***lue***

*2019-11-24 18:30:31*

--------------------------------------------

pourquoi (juste une précision entre parenthèse) ?

***lue***

*2019-11-24 18:31:07*

--------------------------------------------

??? Tu veux pas juste dire "nous" voir "je" ici ?

***lue***

*2019-11-24 18:32:11*

--------------------------------------------

"et maîtrise" ?

***lue***

*2019-11-24 18:31:51*

--------------------------------------------

et

***Framework***

**Shiny**

**Dash**

**Bokeh**

**Escher**

**Observable**

**Référence**

Chang et al.

2015

PlotlY 2017

Bokeh 2014

Gowda 2018, d’après BeZanson et al. 2014

Bostock

**Langage**

R

Python

Python

Julia

JavaScript /

**Maturité**

 

(2012)



(2017)



(2014)



(2016)



(2017)

(D3 : 2011)

**Communauté**

 









**Connaissance personnelle**

 









Chapitre 5

5.4. SIMEDB

basé sur le langage R, était donc assez évident.

**Manipuler les données avec R et dplyr.** Les langages de programma-

tion, et en particulier les plus utilisés en analyse de données, reposent souvent sur une architecture logicielle modulaire. Le langage constitue un cœur, au- tour duquel des bibliothèques logicielles (des *packages* en R) viennent ajouter des fonctionnalités. Parmi ces bibliothèques logicielles, en Python comme en R, certaines sont entièrement dédiées à la manipulation de données tabulaires

– on parle alors de « Data Manipulation Language » (DML) – et permettent d’ef- fectuer des traitements avec des approches fonctionnelles, plutôt qu’avec les structures impératives plus fréquemment utilisées en programmation. En R, ces *packages* constituent de véritables écosystèmes, dotés de leur propre DSL (voir p. 65) et donc d’une grammaire de manipulation de données propre.

L’un de ces *packages*, dplyr (Wickham et al. 2015), s’inscrit dans un écosystème dénommé tidyverse (Wickham 2017), et permet ~~a insi~~ de chaîner des opérations de manipulation de données en une chaîne de traitement complète, plutôt que de faire appel aux habituelles boucles de parcours de matrices propres aux lan- gages de programmation classiques. Ce faisant, avec des opérations chaînées, qui reposent sur des « verbes » permettant d’effectuer des traitements de re- structurations, de modification, de filtrage ou d’enrichissement d’une donnée tabulaire 59, on obtient un ensemble d’instructions qui forment une « phrase » de manipulation de données, exprimées ~~d onc~~ dans la « grammaire de traite- ment de données » fournie par dplyr.

Cette « grammaire » s’inspire notamment du SQL, bien que beaucoup plus com- plète, et peut en particulier être « convertie » en SQL (figure 5.16), c’est-à-dire qu’une suite d’instructions exprimées via dplyr en R (figure 5.16a) peut être traduite en SQL (figure 5.16b), et donc envoyée et exécutée sur un SGBD.

En matière de performance, l’approche de dplyr est intéressante : toutes les opérations sont effectuées par le SGBD directement, et seul le résultat final est renvoyé à R (instruction collect()). Le traitement de données bénéficie donc de la rapidité d’exécution du SGBD MapD, tout en profitant de la syntaxe ex- pressive de dplyr. De plus, cela permet de minimiser les transferts de données : en exécutant les calculs dans le SGBD, il n’est besoin que d’en renvoyer le résul- tat à l’utilisateur. Et ce résultat est nécessairement moins lourd que les données dont il provient. On optimise ainsi l’utilisation de bande-passante internet.

59. Les fonctions de base sont ~~d onc~~ des « verbes », au sens où elles définissent les opérations qui seront effectuées sur les données. On peut ainsi isoler des colonnes avec le « verbe » select, filtrer les lignes avec filter, modifier une colonne avec mutate etc. La figure 5.16a en donne un exemple commenté et concret.

70

***lue***

*2019-11-24 18:37:25*

--------------------------------------------

"dans le cas de R"

***lue***

*2019-11-24 18:40:54*

--------------------------------------------

la phrase ou les instructions ? peut-être (si la phrase) :

"construite selon une "grammaire etc.""

***lue***

*2019-11-24 18:43:58*

--------------------------------------------

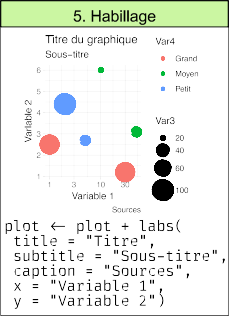
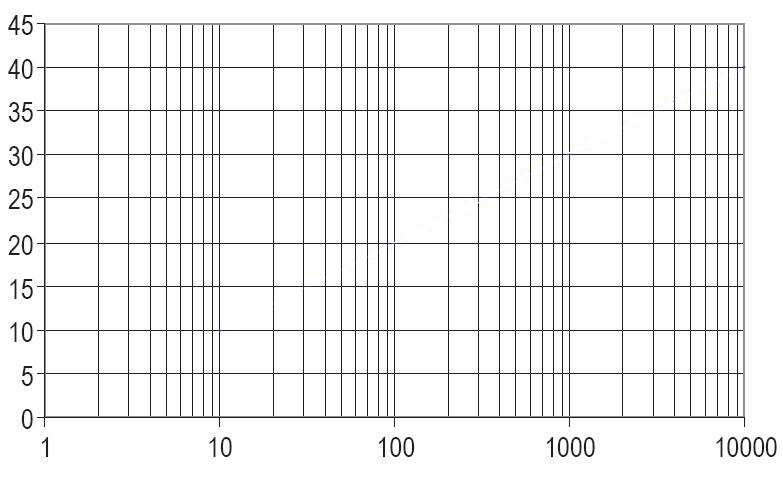
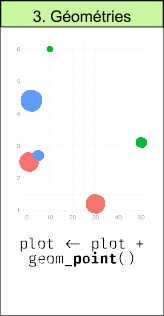
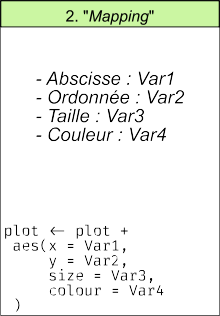
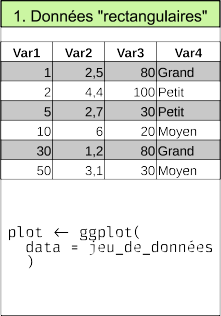
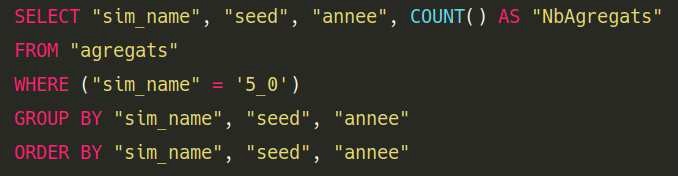
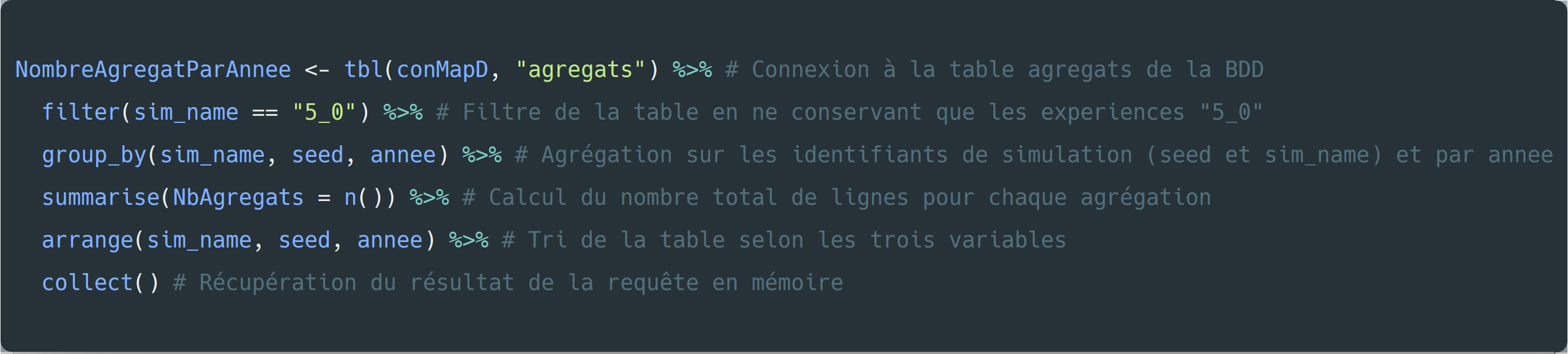
niiice

***lue***

*2019-11-24 18:48:12*

--------------------------------------------

", et"



Chapitre 5

5.4. SIMEDB

(a) Code source R avec le *package* dplyr

(b) Traduction du code source dplyr en SQL

Figure 5.16 – Un exemple de manipulation de données stockées dans un SGBD

depuis R. On y interroge la table des agrégats de population pour calculer le nombre moyen d’agrégats par année de simulation.

**Création de graphiques avec ggplot2 et la « *grammar of graphics* ».** En interrogeant le SGBD avec des outils adaptés, on obtient un jeu de données qui servira de base à la représentation graphique des indicateurs (figure 5.17, étape 1). On peut passer alors passer à l’étape de construction graphique des indicateurs. Il existe pour cela, toujours en restant de l’environnement (et du langage) R, de nombreux *packages* dédiés.

Figure 5.17 – Représentation des éléments de grammaire de ggplot2, d’après une idée de HealY (2018).

L’un des *packages* les plus utilisés, ggplot2 (Wickham 2016), met en œuvre une syntaxe assez adaptée à nos contraintes : ce *package* est conceptuellement fondé sur la « *grammar of graphics* », c’est-à-dire une vision modulaire et très structurée de la conception graphique, pensée par Leland Wilkison (Wilkinson 2006). La logique, assez familière pour un utilisateur de Systèmes d’Information Géo- graphique (SIG), consiste à penser une représentation graphique comme un ensemble de couches (*layers*), qui se superposent, se complètent, et sont toutes basées sur une source de données. Les différentes composantes des données (variables par exemple) sont associées à des composantes graphiques de base (abscisse, ordonnée, taille, couleur …), formant ainsi une mise en correspon- dance (*mapping*) des données avec les composants graphiques (voir figure 5.17, étape 2).

Dans notre cas, cette grammaire est porteuse d’un avantage majeur. Elle est

extrêmement structurée et modulaire. Cela permet de ré-utiliser largement les codes-sources écrits pour un indicateur et de les adapter aisément à d’autres 71

***lue***

*2019-11-24 18:49:48*

--------------------------------------------

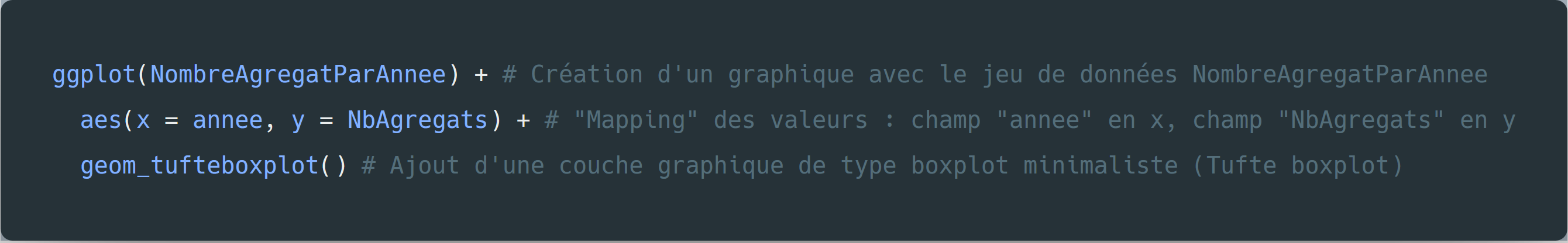
? phrase ? "dans le cadre de l'environnement" ?

***lue***

*2019-11-24 18:59:22*

--------------------------------------------

peut-être un peu petit (je ne me rend pas compte sur l'ordi là) pour lire les lignes de codes/voir les graphiques



Chapitre 5

5.4. SIMEDB

indicateurs. Si la grammaire du graphique est bien définie, elle sera ainsi très

indépendante du contenu des données que l’on y insère.

Par exemple, de nombreux indicateurs de sortie de SimFeodal décrivent l’évo-

lution du nombre d’agents au cours des années de simulation (les agrégats dans la figure 5.18). Ce type de graphique est d’une part rapide à produire avec gg- plot2 : il ne requiert que quelques lignes de code (figure 5.18a). D’autre part, en changeant le tableau de données en entrée (créé dans figure 5.16a), on re- produit exactement le même type de graphique pour, par exemple, un autre type d’agent (le nombre de foyers paysans, d’églises…).

Le *package* ggplot2 répond tout à fait aux contraintes de modularité exposées plus haut, et permet de factoriser le code-source, ce qui garantit une main- tenance plus rapide et une meilleure robustesse de l’application dans son en- semble.

(a) Code source R avec le *package* ggplot2

125

100

75

50

25

820

840 860 880 900 920 940 960 980 1000 1020 1040 1060 1080 1100 1120 1140 1160

annee

(b) Graphique généré

Figure 5.18 – Un exemple de manipulation de données stockées dans un SGBD depuis R.

**Fluidifier les étapes de rendu : le « pipeline de visualisation ».** Dos

Santos et Brodlie (2004) ont conceptualisé et schématisé l’ensemble des étapes nécessaires à la construction d’une visualisation, depuis les données brutes jusqu’à l’image finale, au sein d’un « *pipeline* » de la visualisation (fi- gure 5.19). Ils y décrivent les différents états des données en entrée et en sortie (ligne supérieure), ainsi que les traitements que ces données subissent (ligne inférieure).

72

NbAgregats

***lue***

*2019-11-24 19:02:34*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-24 19:07:39*

--------------------------------------------

ça commence à être un peu répétitif... Proposition :

"Par exemple, pour les nombreux indicateurs de sortie de SimFeodal qui décrivent etc., ce type de graphique est, d'une part, rapide à produire avec ggplot2 parce qu'il ne requiert que quelques lignes de codes. "

***lue***

*2019-11-24 19:08:34*

--------------------------------------------

y'avait pas de "premier type d'agent", du coup :

"on reproduit exactement le même type de graphique pour tous les types d'agents voulus ()."

***lue***

*2019-11-24 19:08:42*

--------------------------------------------

"répond ainsi"

***lue***

*2019-11-24 19:09:02*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-24 19:19:02*

--------------------------------------------

pas le bon titre de figure

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

**●**

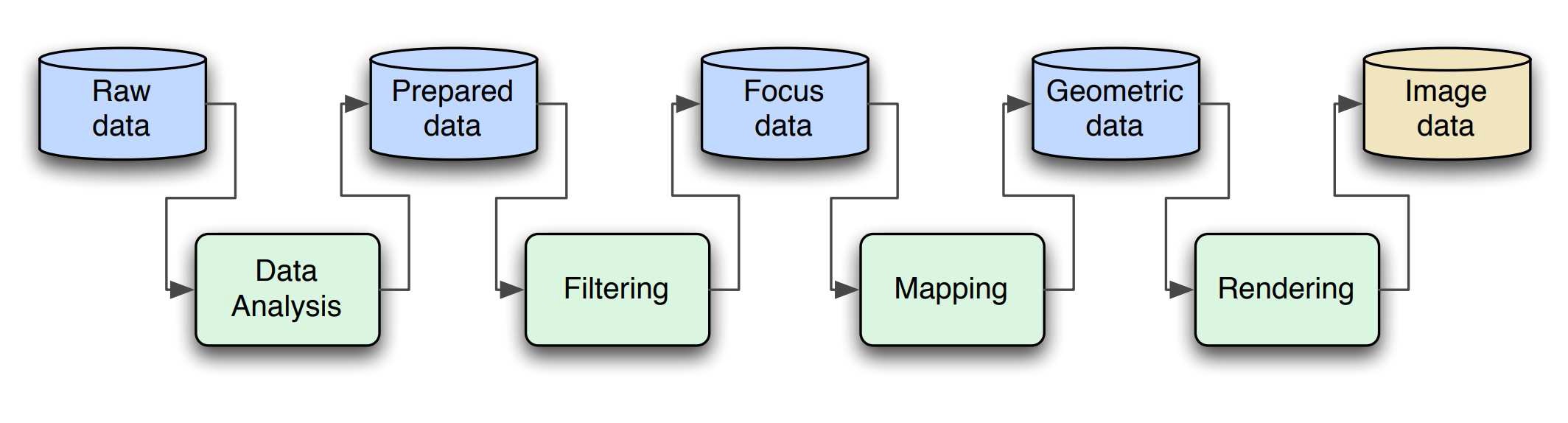
**●**

**●**

**●**

**●**

**●**



Chapitre 5

5.4. SIMEDB

Figure 5.19 – « *The Visualisation Pipeline* », de Keim et al. (2010), p.92, d’après

Dos Santos et Brodlie (2004), p. 314.

Ce *pipeline* débute par des données brutes (*raw data*) auxquelles ont fait su-

bir un traitement (*data analysis*, par exemple une agrégation) pour obtenir des données prêtes à l’utilisation (*prepared data*). Il s’agit ensuite de filtrer ces don- nées (choix des expériences à conserver par exemple), dont l’on conserve donc uniquement les éléments nécessaires (*focus data*). Par une étape de mise en correspondance des variables et des primitives graphiques (*mapping*, voir pa- ragraphe précédent), on obtient un jeu de données « géométriques » (*geometric data*). Cette « géométrie » est à entendre au sens de l’espace de la représenta- tion graphique, qui comprend par exemple les coordonnées des points, lignes, la couleur des cercles et autres éléments mobilisés dans la construction d’un graphique. Il n’est donc aucunement question ici de données géographiques ou spatiales. La dernière étape est plus technique : il s’agit du « rendu gra- phique » (*rendering*), qui convertit un ensemble de spécifications géométriques (textuelles) en une image affichable, faite de pixels (*image data*).

Dans la chaîne de traitement la plus classique, ces étapes s’effectuent au sein

de différents logiciels, chacun dédiés à une tâche. Dans le domaine des uti- lisateurs de SIG, on retrouve par exemple fréquemment une préparation des données dans un tableur, un import dans un logiciel SIG qui va être chargé de la cartographie, puis un export vers un logiciel de dessin vectoriel afin de réaliser la mise en page. À chaque changement de logiciel, il est nécessaire d’exporter les données produites, puis de les ré-importer dans le logiciel suivant.

A contrario, le propre de l’utilisation d’un langage de programmation plutôt

que d’un outil graphique est de pouvoir automatiser et intégrer l’ensemble de ces étapes. L’utilisation de R comme langage de développement de SimEDB nous permet ainsi de développer une unique chaîne de traitement, qui ne re- quiert aucun import/export de données, et peut donc être consolidée, vérifiée et surtout ré-employée *ad libitum*.

L’enchaînement des *packages* employées dans SimEDB est présenté dans la fi-

gure 5.20a, et le code-source correspondant à l’exemple développé dans cette sous-partie dans la figure 5.20.

73

***lue***

*2019-11-24 19:21:57*

--------------------------------------------

"voir ci-dessus" ?

***lue***

*2019-11-24 19:48:00*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-24 19:57:01*

--------------------------------------------

proposition :

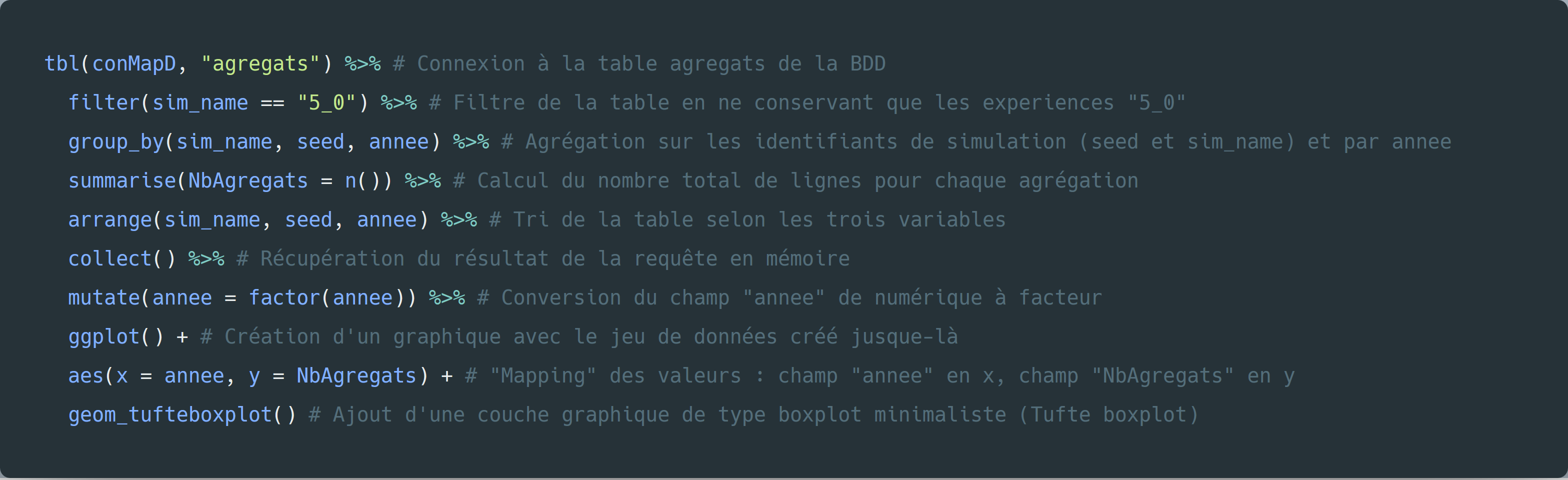
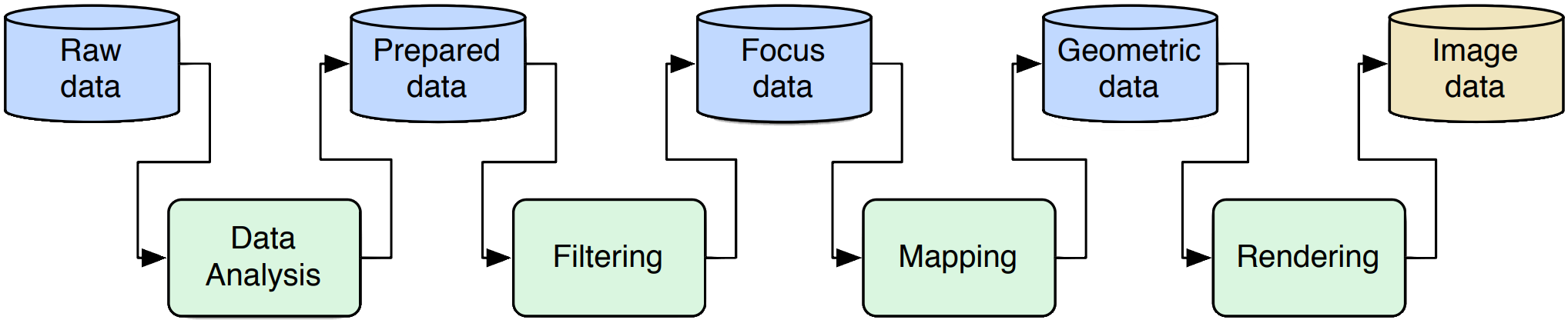
"Les packages utilisés pour construire les chaînes de traitements de données dans SimEDB sont présentés..."

***lue***

*2019-11-24 19:57:24*

--------------------------------------------

", dans"



Chapitre 5

5.4. SIMEDB

SGBD:

**MapD (SQL)**

Traitement: **dplyr (R)**

Visualisation: **ggplot2 (R)**

(a) Technologies utilisées dans SimEDB.

(b) Implémentation d’un exemple de pipeline de visualisation pour construire un indicateur

dans SimEDB

Figure 5.20 – Le « pipeline » de visualisation et son implémentation dans SimEDB. Cette implémentation est obtenue en assemblant les codes des fi- gures 5.16a et 5.18a.

**Modulariser les fonctions.** Shiny, en tant qu’outil de création d’interface

graphique, bénéficie aussi d’un avantage important en matière de conception d’application web : comme ce *package* est basé sur un langage de program- mation modulaire, on peut logiquement créer et ré-utiliser des « briques d’in- terfaces » modulaires. Par l’utilisation de modules 60, il est possible de définir un ensemble d’éléments graphiques adaptatifs et de ré-utiliser tel quel cet en- semble.

Dans l’interface de SimEDB (figure 5.21), ~~p ar exemple,~~ les indicateurs gra- phiques sont toujours présentés de la même manière (encadrés oranges) : dans la partie de droite relative aux indicateurs, l’indicateur à proprement parler est à gauche, et des outils de téléchargement (vectoriel et image) et de notation de l’indicateur (les étoiles) sont placés en haut à droite.

En termes de code-source, la manière de produire les deux indicateurs compa-

rés dans la figure 5.21 est strictement identique : c’est une fonction générique qui prend en entrée des données et un type de graphique à produire. Dans la figure, seul un paramètre varie : le filtre appliqué aux données, qui renvoie ici à différentes expériences. Cela permet donc d’une part de minimiser la taille du code, mais surtout, avec la généricité apportée, de faciliter de manière consi- dérable l’ajout ou la modification d’indicateurs.

60. https ://shiny.rstudio.com/articles/modules.html

74

***lue***

*2019-11-24 20:04:34*

--------------------------------------------

"de tels modules" ?

***lue***

*2019-11-24 20:14:29*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-24 20:14:56*

--------------------------------------------

je dirai plus que c'est un "ainsi" ici : tu montres le résultat de cette modularité

***lue***

*2019-11-24 20:18:24*

--------------------------------------------

ça m'a fait complètement buggué car je ne voyais pas ce qui était "à gauche"... en fait les outils de téléchargements prennent trop peu de place pour que je les considère "à droite", et d'ailleurs tu le dis ils sont "à droite et en haut". Bref, proposition : "les indicateurs graphiques sont toujours représentés de la même manière dans des graphiques à droite (encadrés oranges), qui sont accompagnés d'outils de téléchargements ..... placés à leur droite"

***lue***

*2019-11-24 20:19:35*

--------------------------------------------

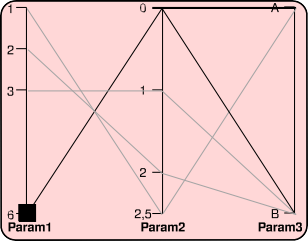
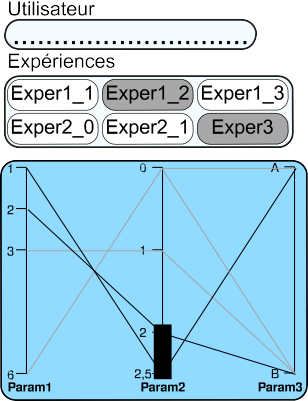
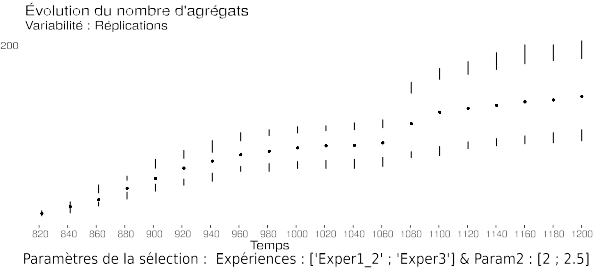
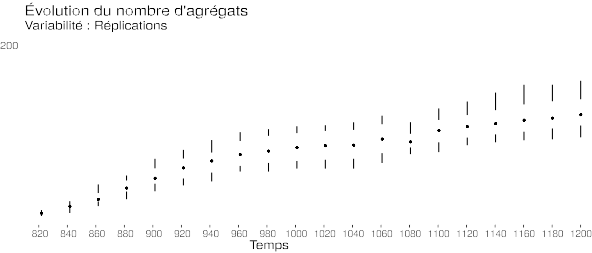
"La modularité" ("cela" ne renvoie à rien de précis ici).

***lue***

*2019-11-24 20:20:07*

--------------------------------------------

"qu'elle apporte" (si tu fais la modif du dessus)



Chapitre 5

5.4.

SIMEDB

Figure 5.21 – Une conception modulaire. Les deux éléments graphiques en-

cadrés sont créés par un même « module » dont les arguments varient. Ici, l’argument est constitué par les expériences sélectionnées : on le visualise dans les graphiques en coordonnées parallèles ~~( gauche)~~ et dans la partie inférieure des ~~des~~ graphiques (texte « Paramètres de la sélection »).

**N.B** : À partir de cette figure, on représente l’interface de SimEDB à l’aide d’un

*mockup* plutôt que de captures d’écrans, pour en simplifier la compréhension.

**5.4.2.2 Choix de l’organisation visuelle**

Les différentes étapes de construction d’une plate-forme d’exploration (sec- tion 5.2 : Comment explorer les sorties de SimFeodal ?) ont conduit à une orga- nisation sous forme de *dashboard* interactif. La forme de ce *dashboard* a évolué tout au long de l’apparition de nouveaux besoins, pour aboutir sur une orga- nisation mono-page, pensée autour de la consultation d’indicateurs de sorties, qui devaient permettre de comparer des expériences différentes sélectionnées au moyen de graphiques en coordonnées parallèles.

Le choix d’un outil dédié à la comparaison, plus qu’à la visualisation des ré-

sultats d’un unique ensemble de simulations, entraîne nécessairement des ré- percussions en matière de présentation visuelle – d’interface graphique – ~~d es~~  ~~éléments permettant de mener cette comparaison.~~ Depuis la première plate- forme aboutie – SimVADB (figure 5.7, reprise ici en figure 5.22a) –, l’interface graphique a fortement évolué par conséquent (figure 5.22b).

75

***lue***

*2019-11-24 20:21:49*

--------------------------------------------

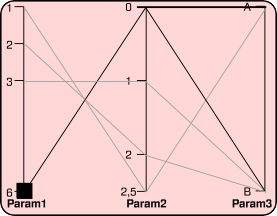
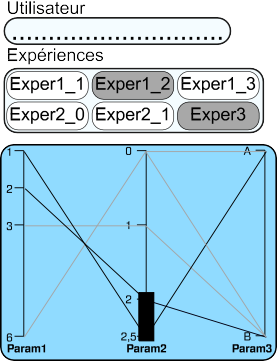
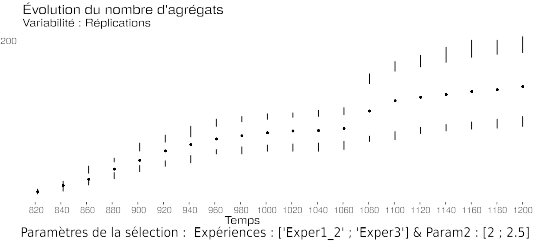
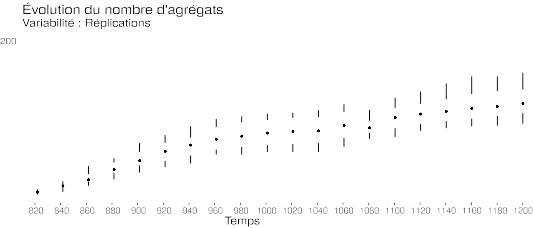
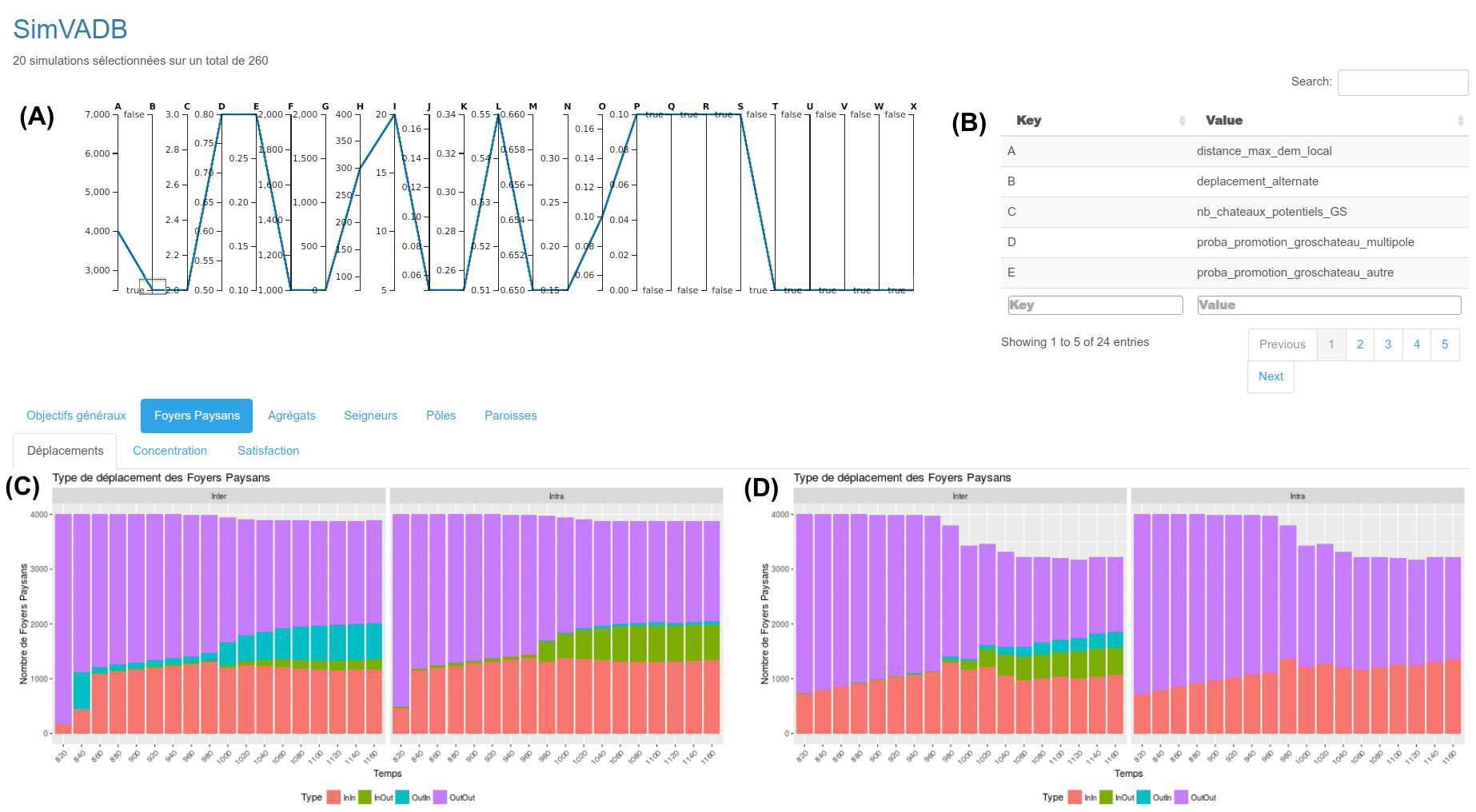
mais du coup on a aucune image de la plateforme telle qu'elle est belle en réalité (d'ailleurs même si elle n'était pas belle ça manquerait). Je pense que tu pourrais en mettre un apperçu (y compris pleine page) au tout début, qui pourrait aller de pair avec le fait de donner l'adresse de l'application.

***lue***

*2019-11-24 20:23:06*

--------------------------------------------

à mettre au début de la phrase.



Chapitre 5

5.4.

SIMEDB

(a) SimVADB

(b) SimEDB

Figure 5.22 – Comparaison visuelle de SimVADB (a) et SimEDB (b).

**Une comparaison verticale.** Le premier changement tient à la disposi-

tion des « contrôleurs », c’est-à-dire aux composants de l’interface sur lesquels l’utilisateur peut jouer pour choisir les expériences qui seront affichées. Dans SimVADB, le seul contrôleur était un graphique en coordonnées parallèles in- teractif, situé dans le haut à gauche de l’interface (figure 5.22a, partie **(A)**). Celui-ci permettait de régler le choix des expériences présentées dans l’indi- cateur de droite (**D**), celui de gauche (**C**) étant constitué d’une moyenne de l’ensemble des expériences.

Dans SimEDB, on a déjà expliqué le choix de permettre une double sélection,

c’est-à-dire de régler les deux sous-ensembles de simulation à comparer. Par conséquent, la barre des contrôleurs (encadré orange dans la figure 5.23) est désormais constitué de deux graphiques en coordonnées parallèles interactifs (en bleu et en rouge), qui agissent sur les indicateurs présentés à leur droite.

76

***lue***

*2019-11-24 20:23:21*

--------------------------------------------

au présent ?

***lue***

*2019-11-24 20:37:19*

--------------------------------------------

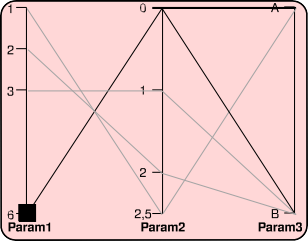
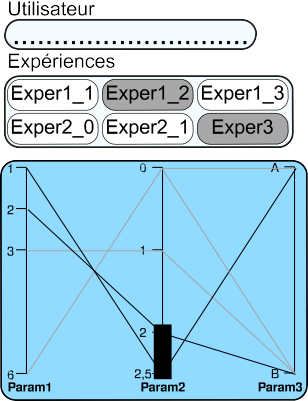
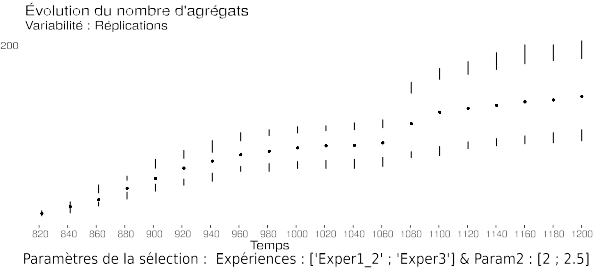
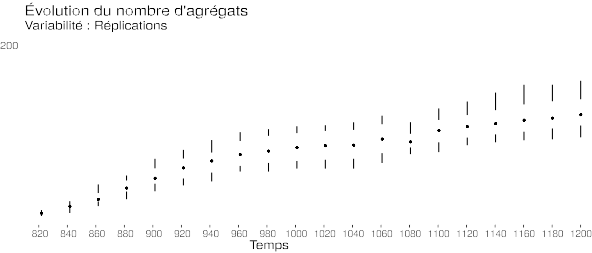
"agir" ?

***lue***

*2019-11-24 20:39:44*

--------------------------------------------

complètement inutile à mon sens de doubler la figure 5.22b en 5.23 : décrit tout sur la figure 5.22 et supprime la 5.23



Chapitre 5

5.4.

SIMEDB

~~Figure 5.23 – La barre de contrôleurs (encadrée en orange) dans l’interface~~

~~de SimEDB.~~

En raison de ces composants supplémentaires dans l’interface, on a choisi d’en changer la disposition. Les contrôleurs sont désormais situés à gauche et les indicateurs à droite, alors qu’ils étaient superposés dans SimVADB.

Ce choix de re-disposition tient à deux arguments. En premier lieu, en termes

d’occupation de l’espace visuel on peut noter que les indicateurs graphiques sont toujours plus larges que haut. Cela s’explique notamment de manière thé- matique, en ce que la variation temporelle, souvent présentée en abscisse, est plus importante que les valeurs attributaires, en ordonnée : les indicateurs de ce type s’analyse plus de manière relative, en observant leur évolution, que de manière absolue. L’occupation de l’espace visuel est donc mieux employée en superposant les indicateurs : cela permet d’augmenter la taille de chacun.

La seconde raison est méthodologique. SimEDB est un outil qui sert à comparer

visuellement des données de sortie de simulation. Dans les indicateur présen- tés, le plus souvent, les intervalles affichés en abcisse sont constants : qu’ils s’agissent du temps, de modalités d’un indicateur (petits seigneurs, grands sei- gneurs…) ou encore de seuils (composition d’un agrégat en nombre de foyers paysans, discrétisée selon des seuils choisis en amont), l’axe des abscisse est identique quelles que soient les expériences comparées. L’axe des ordonnées, au contraire, est plus variable : qu’il représente une valeur absolue, un dé- compte ou une fréquence, les ordres de grandeur de ces valeurs sont très hété- rogènes selon les expériences comparées. Le choix a donc été fait de présenter visuellement les ordres de grandeur comparables, et donc de superposer les sélections différentes (l’axe des abscisses est alors commun) plutôt que de les afficher côte-à-côte (ce qui aurait complexifié la comparaison des ordonnées en les présentant comme directement comparables alors que les ordres de gran- deur varient).

77

***lue***

*2019-11-24 20:41:46*

--------------------------------------------

"les uns au dessus des autres" (superposé je trouve que ça fait vraiment "l'un sur l'autre")

***lue***

*2019-11-24 20:42:40*

--------------------------------------------

"variations des"

***lue***

*2019-11-24 20:43:23*

--------------------------------------------

". Les indicateurs..."

+"relative et longitudinales" ? et "absolue et ponctuelle dans le temps" ?

***lue***

*2019-11-24 20:46:18*

--------------------------------------------

bon pour la suite c'est effectivement plus simple d'utiliser "superposisation" ^^

***lue***

*2019-11-24 20:56:48*

--------------------------------------------

je mettrai plutôt ", ce qui..." ici.

Chapitre 5

5.4. SIMEDB

~~Avec un axe « fixe », il est donc opportun de mener la comparaison visuelle~~

~~sur cet axe, et donc d’aligner les graphiques sur celui-ci.~~ L’organisation des différents indicateurs est donc verticale plutôt qu’horizontale. Afin que la sé- lection des simulations à explorer soit intuitive, les contrôleurs doivent être alignés aux indicateurs, et dès lors, verticalisés eux aussi. Pour bien différen- cier visuellement ce qui relève d’un affichage et ce qui requiert une interaction, les contrôleurs s’inscrivent dans un panneau dédié, grisé (figure 5.23), ce qui constitue presque un standard dans les interfaces modernes d’applications in- teractives.

**Onglets et sous-onglets.** Comme dans SimVADB ~~( figure 5.22a),~~ on a

choisi de conserver une navigation entre indicateurs par un systèmes d’onglets imbriqués : un premier niveau d’onglets permet d’accéder au type d’agents concernés par les indicateurs, et un second niveau permet de sélectionner spé- cifiquement l’indicateur choisi 61.

En terme de disposition, cela force l’utilisateur à interagir avec l’application régulièrement puisque chaque indicateur doit être sur un onglet dédié. La ma- jorité des utilisateurs potentiels de SimEDB consultent toutefois l’application sur des ordinateurs portables, dotés d’écran réduits et d’une résolution faible. L’encombrement visuel est alors atteint rapidement, et mieux vaut présenter un indicateur à la fois plutôt que de présenter l’ensemble des indicateurs sur une unique page : la démarche d’étude visuelle sera plus longue, mais ne sera pas gênée ou faussée par des graphiques de dimension trop réduites qui peuvent induire des erreurs de lecture.

L’organisation des onglets en eux-mêmes pose aussi une question importante :

vaut-il mieux organiser la consultation par type d’agent, ou plutôt, hiérarchi-

quement, selon la catégorie de processus examinée (

terme,

chap3), par exemple en respectant l’ordre de consultation

des indicateurs déterminé ?

Les deux approches présentent des avantages, mais nous avons choisi de rendre

l’utilisation de SimEDB plus intuitive à tous, c’est-à-dire en organisant les indi- cateurs par type d’agents, plutôt qu’efficace, pour les utilisateurs habitués qui auraient bénéficié d’une organisation structurée hiérarchiquement.

**5.4.2.3 Choix des modes d’interactions**

Avant même la conception de SimEDB, avec la plate-forme SimVADB, nous

avions décidé de baser la sélection des simulations sur des graphiques en co- ordonnées parallèles interactifs (section 5.2.5 ~~: « Interagir avec les rapports :~~  ~~exploration interactive »)~~. La logique d’ensemble du filtrage de simulations res- tant la même, il n’était pas nécessaire de modifier ce choix pour SimEDB.

L’accumulation d’expériences, reposant sur les variations de paramètres diffé-

61. Notons que cette question revêt une importance réelle en matière d’ergonomie de l’ap- plication, mais que l’aspect technique en est pourtant assez simple. Pour changer le mode d’organisation des onglets et sous-onglets, il suffirait de ré-organiser les appels aux compo- sants dans le code-source de l’application.

78

***lue***

*2019-11-24 20:58:25*

--------------------------------------------

trop répétitif

***lue***

*2019-11-24 20:59:07*

--------------------------------------------

5.22 si tu supprime la 5.23 (oui ! supprime là ! )

***lue***

*2019-11-24 20:59:44*

--------------------------------------------

"dans SimEDB"

***lue***

*2019-11-24 21:01:08*

--------------------------------------------

"est"

***lue***

*2019-11-24 21:03:30*

--------------------------------------------

proposition (phrase trop longue + ordre pas très convaincant) :

"L'encombrement visuel serait rapidement atteint si l'ensemble des indicateurs étaient présentés sur une page unique. Avec notre solution d'onglets emboîtés, la démarche d'étude visuelle est certes plus longue, mais elle n'est pas gênée ou faussée etc."

***lue***

*2019-11-24 21:05:08*

--------------------------------------------

à expliciter avec un exemple

***lue***

*2019-11-24 21:05:43*

--------------------------------------------

phrase alambiquée + paragraphe à mettre directement à la suite du précédent.

***lue***

*2019-11-24 21:08:36*

--------------------------------------------

pas de saut de paragraphe

faire ref au

le

à corriger une fois fixé sur

Chapitre 5

5.4. SIMEDB

rents, ainsi que la démultiplication des paramètres du modèle SimFeodal ayant

accompagné son paramétrage, ont pourtant demandé de reconsidérer l’usage de ces graphiques interactifs. Là où seuls quelques paramètres étaient mobi- lisés auparavant, les graphiques en coordonnées parallèles reposaient sur peu d’axes. Avec l’augmentation du nombre d’axes, le graphique en coordonnées parallèle est rapidement devenu illisibleen raison de la surcharge graphique due au recouvrement des axes.

**Réduire la surcharge visuelle des graphiques en coordonnées paral-**

**lèles.** La première mesure pour y remédier a été de filtrer les paramètres af- fichés : nul besoin d’afficher un axe correspondant à un paramètre qui n’est jamais manipulé dans les expériences. Plutôt que de définir les paramètres

« utiles », et donc d’avoir à les redéfinir dans l’application à chaque ajout d’ex- périence qui reposerait sur la variation d’un paramètre différent, nous avons fait en sorte que cette discrimination ~~d es paramètres « actifs »~~ soit exécutée de manière automatique : quand SimEDB est lancé, une requête est exécutée sur la table des paramètres pour identifier ceux qui présentent plusieurs modali- tés et ceux qui n’en ont qu’une. Seuls sont alors affichés les paramètres de la première catégorie, car eux-seuls présent un intérêt à être discriminés.

Ce faisant, le nombre de paramètre affichés est réduit, et permet d’afficher leurs

intitulés plutôt que de faire appel à une table de correspondance comme dans SimVADB (figure 5.22a, partie **(B)**). L’automatisation de ce traitement permet de plus de ne pas avoir à changer quoi que ce soit à la plate-forme lors d’ajouts ou de suppressions de simulations de la base de données, ce qui concoure à l’objectif d’indépendance aux données de la plate-forme d’exploration.

**Pré-filtrer les simulations.** Au fur et à mesure du paramétrage puis de

la calibration de SimFeodal, les expériences ont tout de même continué à mo- biliser de plus en plus de paramètres différents. Pour réduire la quantité d’in- formation représentée et améliorer en conséquence « l’expérience utilisateur », nous avons ajouté un filtre, moins visuel que les graphiques en coordonnées pa- rallèle, qui permet toutefois de restreindre le nombre de simulations affichées à partir de leur dénomination. Plutôt que de cibler des valeurs spécifiques de paramètres, l’idée est donc de soustraire des choix possibles des expériences entières. Pour SimEDB, on a donc ajouté un pré-filtrage, sous forme de « boîte de sélection » (*select input*, figure 5.24), qui interroge la base de données direc- tement pour connaître les différents intitulés de simulations et agit comme un premier filtre réduisant donc les simulations interrogées dans les graphiques en coordonnées parallèles.

79

***lue***

*2019-11-24 21:10:00*

--------------------------------------------

Avec "là où", on attend une opposition dans la phrase. Proposition ;

"Auparavant, seuls quelques paramètres était mobilisés et les graphiques en coordonnées..."

***lue***

*2019-11-24 21:10:07*

--------------------------------------------

ajouter espace

***lue***

*2019-11-24 21:11:17*

--------------------------------------------

"de paramètres, et donc d'axes dans les graphiques, ces derniers sont rapidement devenus... "

***lue***

*2019-11-24 21:13:03*

--------------------------------------------

"opérée" me semble mieux

***lue***

*2019-11-24 21:14:31*

--------------------------------------------

comparés, étudiés, analysés ? Ta justification ça fait un peu "ils sont discriminés parce qu'ils sont discriminés" Proposition pour que ce soit + explicite :

"utilisés pour comparer différentes simulations"

***lue***

*2019-11-24 21:14:38*

--------------------------------------------

"présentent"

***lue***

*2019-11-24 21:15:04*

--------------------------------------------

"représentés" ? (pour éviter la répétition)

***lue***

*2019-11-24 21:19:27*

--------------------------------------------

Deux idées très différentes, donc je te propose plutôt deux phrases :

"Ce faisant, le nombre de paramètres est réduit est les axes des graphiques en coordonnées parallèles sont plus lisibles. De plus, cette réduction du nombre de paramètre permet d'afficher leurs intitulés en entier, alors qu'il était nécessaire, dans SimVADB, de faire appel à une table de correspondance (appel figure), ce compliquait l'utilisation de la plateforme / ralentissait la lecture des graphiques / blabalba".

***lue***

*2019-11-24 21:15:46*

--------------------------------------------

pas de saut de paragraphe

***lue***

*2019-11-24 21:20:09*

--------------------------------------------

"permet en définitive" ? rajouter un petit mot conclusif. Ou bien "Enfin"

***lue***

*2019-11-24 21:23:38*

--------------------------------------------

"En complément du ciblage de valeurs..."

***lue***

*2019-11-24 21:26:51*

--------------------------------------------

dejà dit "on a ajouté un filtre", proposition pour que ce soit moins répétitif : "Nous avons ajouté ce préfiltrage sous forme de..."

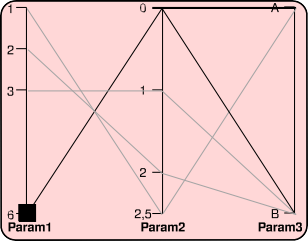
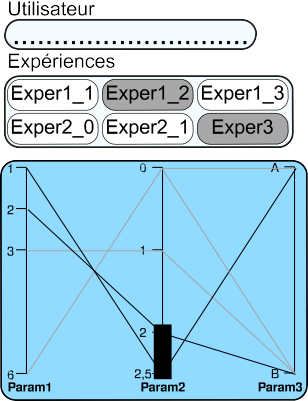
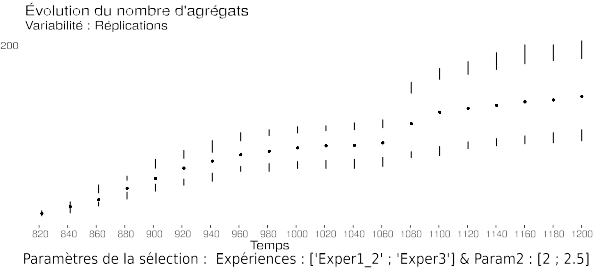
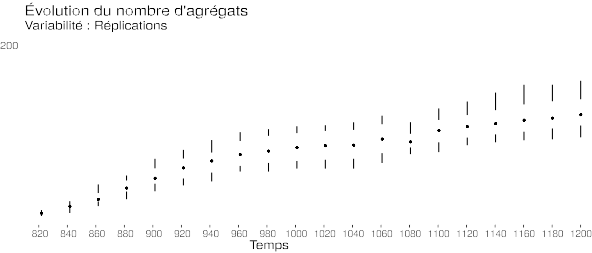
***lue***

*2019-11-24 21:31:55*

--------------------------------------------

répétitioooooon proposition du coup pour toute la phrase :

"Nous avons ajouté ce préfiltrage sous la forme d'une "boite de sélection" (blabla) qui affiche les intitulés des simulations directement à partir de la base de données et qui permet à l'utilisateur de choisir quelles simulations prendre en compte". (ou tu peux finir par "dans les graphiques etc" mais je pense pas que ce soit nécessaire).



Chapitre 5

5.4.

SIMEDB

Figure 5.24 – Le menu de sélection des expériences (encadré en orange) qui

permet un pré-filtrage des expériences à partir de leur nom.

**Optimiser l’occupation de l’espace visuel.** En dépit de ces différentes

techniques visant à minimiser le nombre d’axes affichées dans les graphiques en coordonnées parallèles, la place prise par ces graphiques reste importante, en particulier quand on décide de ne pas diminuer la taille des éléments de lé- gende afin de conserver leur lisibilité. Quand l’application est consultée sur un écran de taille faible, l’appréhension de l’ensemble des informations présentes dans l’interface pose ainsi un véritable problème.

En réfléchissant aux séquences d’usages, par les utilisateurs, de SimEDB, on a

pu comprendre que le mode d’utilisation le plus classique était de ~~b ien~~ consi- dérer le filtrage à effectuer sur les graphiques en coordonnées parallèles, en y consacrant un temps certain, avant de comparer longuement les différents indicateurs de la sélection. Il n’est donc que rarement fait usage de multiples filtres successifs sur un seul indicateur, dans une approche ~~p lus~~ exploratoire

~~donc,~~ mais plutôt d’évaluations complètes de simulations choisies.

Il n’est alors plus indispensable de consacrer une part importante de l’espace aux zones interactives (le panneau de contrôle), ou du moins, pas pendant l’ensemble de la période d’évaluation des simulations.

Un outil de redimensionnement a alors été ajouté à SimEDB, permettant, par

glisser-déposer, de modifier la largeur occupée par le panneau de contrôle en l’adaptant à chaque moment au besoin de visualisation. La figure 5.25 montre ainsi une succession d’états : en début d’exploration, l’utilisateur va augmenter la taille du panneau de contrôle pour augmenter la lisibilité des graphiques en coordonnées parallèles et effectuer une sélection plus simplement. Une fois la sélection effectuée, il pourra ~~a lors re-d~~iminuer la largeur du panneau afin d’augmenter la zone disponible pour les indicateurs de sorties de simulation,

80

***lue***

*2019-11-24 21:32:21*

--------------------------------------------

"petite taille" ?

***lue***

*2019-11-24 21:32:53*

--------------------------------------------

"En travaillant sur" ?

***lue***

*2019-11-24 21:38:56*

--------------------------------------------

"établi" ? "observé" ?

***lue***

*2019-11-24 21:41:39*

--------------------------------------------

"dans un premier temps" + "et de manière approfondie" ?

***lue***

*2019-11-24 21:42:01*

--------------------------------------------

"puis, dans un second temps seulement,"

***lue***

*2019-11-24 21:47:04*

--------------------------------------------

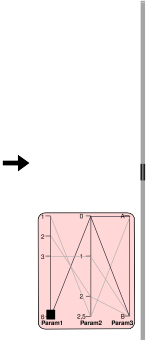
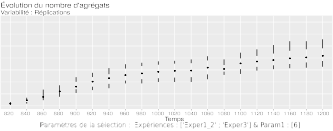
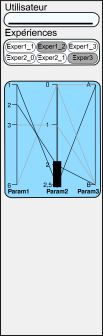
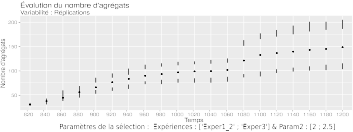
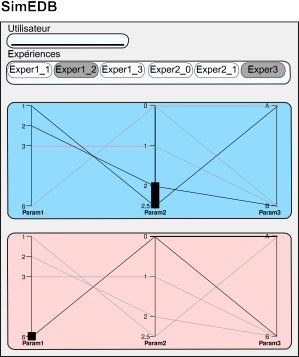
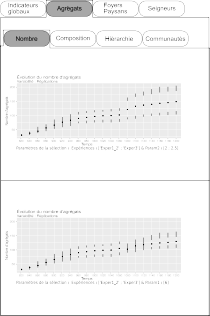
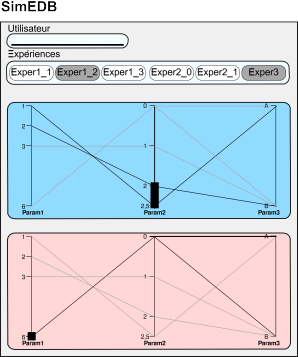
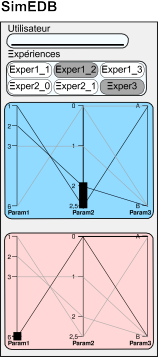
"selon" ? mais dans tous les cas je pense que tu pourrais supprimer cet proposition, qui je trouve rend le propos moins clair (ou alors le mettre entre parenthèses)

***lue***

*2019-11-24 21:47:24*

--------------------------------------------

virgule



Chapitre 5

5.4.

SIMEDB

et donc la taille de ces derniers.

Figure 5.25 – Utilisation interactive

panneau de contrôle.

de SimEDB et redimensionnement du

**Répondre aux demandes des utilisateurs ~~: ajout d’un mécanisme d’ex-~~**

**~~port des indicateurs.~~** L’intérêt d’une interface modulaire et factorisée se ré- vèle véritablement quand les utilisateurs d’un outil demandent des fonctionna- lités supplémentaires, non prévues lors de la conception de l’outil. Dans le cas de SimFeodal, une telle requête est rapidement apparue : les thématiciens, mais aussi les modélisateurs, pour conserver une trace d’une session d’exploration des indicateurs, souhaitaient pouvoir exporter les graphiques correspondant aux indicateurs.

Si au départ, une simple capture d’écran pouvait suffire, ce besoin a été com-

plété par une volonté d’inclure des indicateurs de simulations dans des articles et autres communications, requérant donc des retouches des graphiques. Pour ce faire, on a choisi d’ajouter des fonctionnalités de téléchargement des gra- phiques, selon deux formats – image et vectoriel – afin de satisfaire à ces deux usages. Avec le développement modulaire adopté, il a suffit d’ajouter ces fonc- tions d’export en un unique lieu dans le code-source de SimEDB, et ~~l ’ajout~~  ~~de~~ ces nouvelles fonctionnalités a alors été disponible automatiquement pour chacun des indicateurs graphiques.

Un autre élément répond à une demande forte des utilisateurs : dans une ré-

utilisation hors application des indicateurs, il était difficile de se souvenir des sélections effectuées pour produire un indicateur : en terme de reproductibilité, une fois un graphique exporté sur un ordinateur, il n’y avait plus aucun moyen de connaître les conditions précises de sa création. Nous avons donc ajouté, sur les graphiques eux-mêmes, un résumé des expériences et valeurs de paramètres sélectionnées dans chacun des indicateurs. Cela permet d’en conserver une trace plus durable et augmente leur potentiel réutilisation.

81

***lue***

*2019-11-24 21:49:31*

--------------------------------------------

"des graphiques représentant ces derniers"

***lue***

*2019-11-24 21:52:02*

--------------------------------------------

au pluriel plutôt non ?

***lue***

*2019-11-24 21:52:30*

--------------------------------------------

mettre ç la fin de la phrase plutôt

***lue***

*2019-11-24 21:52:57*

--------------------------------------------

virgule

***lue***

*2019-11-24 21:53:27*

--------------------------------------------

"ont alors été disponibles"

***lue***

*2019-11-24 21:53:54*

--------------------------------------------

Faire une nouvelle phrase

***lue***

*2019-11-24 21:54:15*

--------------------------------------------

"pour"

***lue***

*2019-11-24 21:55:43*

--------------------------------------------

pas que ça au final

Chapitre 5

5.4. SIMEDB

**Noter les simulations.** Un dernier point d’interaction avec l’application

a été prévu, sans pouvoir toutefois être mobilisé jusque là : il s’agissait d’aller vers une semi-automatisation de l’évaluation des simulations, par l’intermé- diaire d’un outil graphique permettant de « noter » les simulations sélection- nées. Pour ce faire, et parce que, on l’a vu, l’évaluation d’un ensemble de simu- lations ne peut se faire de manière unique, on a choisi de donner la possibilité aux utilisateurs experts de noter chacun des indicateurs de sortie, pour cha- cun des ensembles de simulations qu’ils exploreraient. L’évaluation se fait au moyen d’un outil simple, composé de 5 « étoiles », et est enregistré à chaque nouvelle note.

Une piste d’utilisation serait de mobiliser les données ainsi créées, composées

d’une note donnée à un indicateur pour un ensemble d’identifiants uniques de simulations, afin de réaliser des analyses quantitatives des notes attribuées : est-ce que certaines simulations sont systématiquement bien notées avec cha- cun des indicateurs affichés ? Certains indicateurs ne sont-ils jamais observés ou ne donnent-ils jamais lieu à évaluation ?

Cette fonctionnalité, bien qu’implémentée, n’est pas encore utilisée, mais de-

vrait à terme permettre d’aller vers une meilleure connaissance des résultats de simulation, tout autant que vers une mesure de l’efficacité des indicateurs de sortie choisis pour évaluer un ensemble de simulations.

**5.4.2.4 Présentation générale**

SimEDB est une application interactive. Il nous semblait dommage d’en pré-

senter l’utilisation d’ensemble par une succession de captures d’écran commen- tées. Nous avons donc réalisé une vidéo qui en montre l’usage et est disponible

à cette adresse :

82

***lue***

*2019-11-24 21:56:44*

--------------------------------------------

"afin d'en réaliser des analyses quantitatives"

***lue***

*2019-11-24 21:57:12*

--------------------------------------------

trop bien !

A faire...

Chapitre 5

5.4. SIMEDB

**Conclusion**

Au terme de la construction de la plate-forme d’exploration, nous disposons

donc d’une application, SimEDB, conçue et développée spécifiquement pour les problématiques propres à l’exploration des données de SimFeodal.

Elle s’inscrit dans les méthodes des Interactions Homme-Machine, ou même

dans ce que certains nomment désormais les « Interactions Homme-Données » (« *Human-Data Interaction* », Elmqvist 2011 ; Mortier et al. 2014) et s’efforce de suivre les préceptes identifiés dans ce champs (Amirpour Amraii 2018,

p. 167-170 par exemple).

Le développement a été fortement guidé par les contraintes et besoins identi-

fiées, aussi bien en terme d’approches méthodologiques que de choix technolo- giques. SimEDB est donc un outil *ad-hoc*, toutefois pensé de manière modulaire. Tous les composants logiciels de SimEDB sont indépendants et communiquent de manière standardisée, ouvrant la voie à leur remplacement ou « interchan- geabilité » : l’architecture logicielle et les choix techologiques le permettent. La plate-forme SimEDB est donc intrinsèquement pensée comme une réponse à des besoins spécifiques, mais cette réponse a été conçue comme générique et en mesure d’être adaptée aisément à d’autres types de données et/ou sorties de modèles de simulation.

Plus généralement, l’ensemble de ce chapitre montre une démarche similaire,

pensée pour répondre à des besoins spécifiques avec des solutions génériques et généralisables. Le passage, depuis une succession de rapports jusqu’à une application d’exploration de ces rapports, ou encore les différents éléments relatifs au choix d’un système de gestion de base de données ou au dessin d’un modèle conceptuel de données s’inscrivent en effet dans cette même démarche qui s’ancre profondément dans une logique de recherche reproductible, aussi bien d’un point de vue technique que de celui du concept et de la méthodologie.

83

***lue***

*2019-11-24 21:59:40*

--------------------------------------------

booooum.

Peut-être pour te mettre plus en valeur :

"mais nous avons conçu cette réponse comme générique..."

***lue***

*2019-11-24 21:59:59*

--------------------------------------------

similaire à quoi ?

***lue***

*2019-11-24 22:01:39*

--------------------------------------------

"montre comment nous avons travaillé de manière à ce que les solutions choisies répondent à des besoins spécifiques tout en étant génériques et généralisables. "

***lue***

*2019-11-24 22:05:57*

--------------------------------------------

phrase troooooop longue mais surtout trop alambiquée Proposition pour tout le paragraphe :

"Plus généralement, l'ensemble de ce chapitre montre comment nous avons travaillé de manière à ce que les solutions choisies répondent à des besoins spécifiques tout en étant génériques et généralisables. Que ce soit pour passer d'une succession de rapports à une plateforme d'exploration, pour choisir un système de gestion de base de donnée ou un type de modèle conceptuel de données, notre démarche s'ancre profondément dans une logique de recherche reproductible, et ce tant du point de vue technique que conceptuel et méthodologique".

A reprendre après redécoupage du chapitre !

Agarwal, Sameer et al. (2013). « BlinkDB : Queries with Bounded Errors and

Bounded Response Times on Very Large Data ». In : *Proceedings of the 8th ACM European Conference on Computer Systems*. EuroSys ’13. 00474. New York, NY, USA : ACM, p. 29-42. doi : 10/bwrd. url : [http://doi.acm.](http://doi.acm/) org/10.1145/2465351.2465355.

Amirpour Amraii, Saman (2018). « Human-Data Interaction in Large and High-Dimensional Data ». PhD Thesis. University of Pittsburgh.

BattY, Michael (2015). « A Perspective on City Dashboards ». In : *Regional Studies, Regional Science* 2.1. 00016, p. 29-32. doi : 10/gfw9mr. url : https:

//doi.org/10.1080/21681376.2014.987540.

BeZanson, Jeff et al. (2014). « Julia : A Fresh Approach to Numerical Compu- ting ». In : url : [http://arxiv.org/abs/1411.1607.](http://arxiv.org/abs/1411.1607)

Bimonte, S., A. Tchounikine et M. Miquel (2005). « Towards a Spatial Multi- dimensional Model ». In : *Proceedings of the 8th ACM International Workshop on Data Warehousing and OLAP*. DOLAP ’05. 00075. New York, NY, USA : ACM, p. 39-46. doi : 10/cfn8w7. url : <http://doi.acm.org/10.1145/> 1097002.1097009.

Bimonte, Sandro (2007). *Intégration de l’information Géographique Dans Les En- trepôts de Données et l’analyse En Ligne : De La Modélisation à La Visualisation*. 00029. Lyon, INSA. url : [http://www.theses.fr/2007ISAL0105.](http://www.theses.fr/2007ISAL0105)

Bokeh, Development Team (2014). *Bokeh : Python Library for Interactive Visua- lization*. url : https://docs.bokeh.org/en/latest/index.html.

Bostock, Mike (2018). *A Better Way to Code*. url : https : / / medium . com /

@mbostock/a-better-way-to-code-2b1d2876a3a0.

Chang, Winston et al. (2015). « Shiny : Web Application Framework for R ».

In : *R package version 0.11* 1.4. 00553, p. 106.

Commenges, Hadrien et al. (2014). *R et espace : Traitement de l’information géographique*. Groupe ElementR. Lyon : Framabook.

Cura, Robin (2015). « Créer des documents reproductibles et des applications

web interactives d’analyse de données avec R : Knitr & Shiny ». http ://umr5600.ish- lyon.cnrs.fr/20150610\_EVS-ISIG\_CafeMethodo. url : https : / / github . com/RCura/CafeMethodo.

84

**Références**

Chapitre 5

RÉFÉRENCES

Cura, Robin (2017a). « Making Large

Spatio-Temporal Data Analysis Easier.

Illustrated Plea for Using (Geo)Visual Analytics. » cites : cura\_making\_2017.

url : [http://www.geog.leeds.ac.uk/ectqg17/home.html.](http://www.geog.leeds.ac.uk/ectqg17/home.html)

— (2017b). « « TimeLineEDB », application web d’exploration interactive de données de géolocalisation ». In : *M@ppemonde* 120.2015/4. url : https:

//halshs.archives-ouvertes.fr/halshs-01935702/document.

Dos Santos, Selan et Ken Brodlie (2004). « Gaining Understanding of Mul- tivariate and Multidimensional Data through Visualization ». In : *Compu- ters & Graphics* 28.3, p. 311-325. doi : 10 / cttwpw. url : http : / / www . sciencedirect.com/science/article/pii/S0097849304000251.

Elliott, Roxana (2017). *How Page Load Time Affects Bounce Rate and Page Views*. Avec la coll. de Section.io. url : https://[www.section.io/blog/](http://www.section.io/blog/) page-load-time-bounce-rate/.

Elmqvist, Niklas (2011). « Embodied Human-Data Interaction ». In : *ACM CHI 2011 Workshop “Embodied Interaction : Theory and Practice in HCI*, p. 104- 107.

Epstein, Joshua M. et Robert L. Axtell (1996). *Growing Artificial Societies : Social Science from the Bottom Up*. 05176. Washington, D.C : MIT Press. 228 p.

Fekete, Jean-Daniel (2010). « Infrastructure ». In : *Mastering the Information Age - Solving Problems with Visual Analytics*. Sous la dir. d’Eurographics Association. Eurographics Association, p. 87-108. url : https://hal. inria.fr/hal-00696814.

Fekete, Jean-Daniel et al. (2019). « Progressive Data Analysis and Visuali- zation ». In : Dagstuhl Seminar 18411. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. url : https://[www.dagstuhl.de/en/program/calendar/](http://www.dagstuhl.de/en/program/calendar/) semhp/?semnr=18411.

Few, Stephen (2006a). *Information Dashboard Design : The Effective Visual Com- munication of Data*. O’Reilly Media, Inc.

— (2006b). « Multivariate Analysis Using Parallel Coordinates ». In : *Perceptual edge*. 00039, p. 1-9.

Forch, Valentin et al. (2017). « Are 100 Ms Fast Enough ? Characterizing La- tency Perception Thresholds in Mouse-Based Interaction ». In : *Engineering Psychology and Cognitive Ergonomics : Cognition and Design*. Sous la dir. de Don Harris. Lecture Notes in Computer Science. 00001. Springer Interna- tional Publishing, p. 45-56.

Fotheringham, A. Stewart (1999). « Trends in Quantitative Methods III : Stressing the Visual ». In : *Progress in Human Geography* 23.4. 00038, p. 597- 606. doi : 10/c549jd. url : https://doi.org/10.1191/030913299667756016.

Gowda, Shashi (2018). *Escher - Composable Web UIs in Julia.* JuliaGizmos. url :

https://github.com/JuliaGizmos/Escher.jl.

Grignard, Arnaud et Alexis Drogoul (2017). « Agent-Based Visualization : A Real-Time Visualization Tool Applied Both to Data and Simulation Out- puts ». In : *The AAAI-17 Workshop on Human-Machine Collaborative Learning*. Association for the Advancement of Artificial Intelligence 17. 00002.

HealY, Kieran (2018). *Data Visualization : A Practical Introduction*. 00000. S.l. : Princeton University Press. 304 p. url : [http://socviz.co/.](http://socviz.co/)

85

Chapitre 5

RÉFÉRENCES

Heinrich, Julian et Daniel Weiskopf (2013). « State of the Art of Parallel

Coordinates ». In : Eurographics (STARs), p. 95-116. doi : 10/gd87qm. url :

https://diglib.eg.org:443/handle/10.2312/conf.EG2013.stars.095- 116.

Iannone, Richard, Joseph J. Allaire et Barbara Borges (2018). *Flexdash- board : R Markdown Format for Flexible Dashboards*. 00000 R package version

0.5.1.1. url : https://CRAN.R-project.org/package=flexdashboard. Inselberg, Alfred et Bernard Dimsdale (1987). « Parallel Coordinates for

Visualizing Multi-Dimensional Geometry ». In : *Computer Graphics 1987*. 01538. Springer, p. 25-44.

Keim, Daniel et al. (2010). *Mastering the Information Age Solving Problems with Visual Analytics*. 00019. Eurographics Association.

Kitchin, Rob, Tracey P. Lauriault et Gavin McArdle (2015). « Knowing and Governing Cities through Urban Indicators, City Benchmarking and Real-Time Dashboards ». In : *Regional Studies, Regional Science* 2.1. 00169, p. 6-28. doi : 10 / gc92g7. url : https : / / doi . org / 10 . 1080 / 21681376 . 2014.983149.

Laurini, Robert (2018). « From GIS to Smart Cities : Key-Concept Landmarks ». In : *AIAP’18*. International Conference on Artificiel Intelligence and Its Ap- plication. Sous la dir. de Brahim Lejdel, Mohammed-Khireddine Kholladi et Mohammed Charaf Eddine Meftah. El Oued, Algeria.

Liu, Z. et J. Heer (2014). « The Effects of Interactive Latency on Exploratory Visual Analysis ». In : *IEEE Transactions on Visualization and Computer Gra- phics* 20.12. 00125, p. 2122-2131. doi : 10/f3tvrw.

MacEachren, Alan M. et al. (2004). « Geovisualization for Knowledge Construc- tion and Decision Support ». In : *IEEE computer graphics and applications* 24.1. 00256, p. 13-17. pmid : 15384662. url : https://www.ncbi.nlm. nih.gov/pmc/articles/PMC3181162/.

MacKenZie, I. Scott et Colin Ware (1993). « Lag As a Determinant of Human Performance in Interactive Systems ». In : *Proceedings of the INTERACT ’93 and CHI ’93 Conference on Human Factors in Computing Systems*. CHI ’93. 00442. New York, NY, USA : ACM, p. 488-493. doi : 10 / dr7rj6. url : [http://doi.acm.org/10.1145/169059.169431.](http://doi.acm.org/10.1145/169059.169431)

Mortier, Richard et al. (2014). *Human-Data Interaction : The Human Face of the Data-Driven Society*. SSRN Scholarly Paper ID 2508051. Rochester, NY : Social Science Research Network. url : https://papers.ssrn.com/ abstract=2508051.

Nielsen, Jakob (2009). *Powers of 10 : Time Scales in User Experience*. url :

https://[www.nngroup.com/articles/powers-](http://www.nngroup.com/articles/powers-) of- 10- time- scales- in- ux/.

Pafka, Szilard (2017). *Benchm-Databases : A Minimal Benchmark of Various Tools (Statistical Software, Databases Etc.) for Working with Tabular Data of Moderately Large Sizes (Interactive Data Analysis)*. url : https://github. com/szilard/benchm-databases.

Pandre, Andrew (2011). *Charts and Their Dimensionality*. url : https : / / apandre.wordpress.com/dataviews/dimensionality/.

86

Chapitre 5

RÉFÉRENCES

Patel, Neil (2011). *Speed Is A Killer*. 00000. url : https://neilpatel.com/

blog/speed-is-a-killer/.

PlotlY (2017). *Introducing Dash*. url : [https://medium.com/@plotlygraphs/](https://medium.com/%40plotlygraphs/) introducing-dash-5ecf7191b503.

Raasveldt, Mark et Hannes Mühleisen (2018). « MonetDBLite : An Embed- ded Analytical Database ». In : *Proceedings of the 2018 International Confe- rence on Management of Data*. SIGMOD ’18. New York, NY, USA : ACM, p. 1837-1838. doi : 10/gfw9mm. url : [http://arxiv.org/abs/1805.08520.](http://arxiv.org/abs/1805.08520)

ReY-CoYrehourcq, Sébastien (2015). « Une plateforme intégrée pour la construc- tion et l’évaluation de modèles de simulation en géographie ». Thèse de doctorat en Géographie. Paris : Université Paris I - Panthéon-Sorbonne.

ReYnolds, Craig W. (1987). « Flocks, Herds and Schools : A Distributed Beha- vioral Model ». In : *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH ’87. 10133. New York, NY, USA : ACM, p. 25-34. doi : 10/chhdjr. url : [http://doi.acm.org/10.](http://doi.acm.org/10) 1145/37401.37406.

Rivard, Kurt et Doug Cogswell (2004). « Are You Drowning in BI Reports ? Using Analytical Dashboards to Cut through the Clutter ». In : *DM Review, http ://goo. gl/hle9Wc*. 00000.

Root, Christopher et Todd Mostak (2016). « MapD : A GPU-Powered Big Data Analytics and Visualization Platform ». In : *ACM SIGGRAPH 2016 Talks*. SIGGRAPH ’16. New York, NY, USA : ACM, 73 :1-73 :2. doi : 10/gd7hg8.

url : [http://doi.acm.org/10.1145/2897839.2927468.](http://doi.acm.org/10.1145/2897839.2927468)

Roth, Robert E. (2013). « Interactive Maps : What We Know and What We Need to Know ». In : *Journal of Spatial Information Science* 6. 00099. doi : 10/gfw9mq. url : http:/ /www .josis. org/ index. php/josis /article / view/105.

— (2015). « Interactivity and Cartography : A Contemporary Perspective on User Interface and User Experience Design from Geospatial Professionals ». In : *Cartographica : The International Journal for Geographic Information and Geovisualization*. 00016. doi : 10/gfw9mp. url : https://www.utpjournals. press/doi/abs/10.3138/cart.50.2.2427.

Roumpani, F., O. O’Brien et A. Hudson-Smith (2013). « Creating, Visuali- zing and Modelling the Realtime City ». In : *Proceedings of Hybrid City II ‘Subtle rEvolutions’ Conference*. 00002.

Schelling, Thomas C (1971). « Dynamic Models of Segregation ». In : *Journal of mathematical sociology* 1.2. 04241, p. 143-186.

Shneiderman, Ben (1996). « The Eyes Have It : A Task by Data Type Taxo- nomy for Information Visualizations ». In : *Proceedings 1996 IEEE Symposium on Visual Languages*. IEEE, p. 336-343. doi : 10/fwdq26.

*Snowflake Schema* (2018). In : *Wikipedia*. Page Version ID : 856363739. url :

https : / / en . wikipedia . org / w / index . php ? title = Snowflake \_ schema & oldid=856363739.

*Star Schema* (2018). In : *Wikipedia*. Page Version ID : 853084099. url : https:

//en.wikipedia.org/w/index.php?title=Star\_schema&oldid=853084099.

Tufte, Edward R. (2001). *The Visual Display of Quantitative Information*. 2nd edition. 11580. Cheshire, Conn : Graphics Press USA. 190 p.

87

Chapitre 5

RÉFÉRENCES

Vermeij, Maarten et al. (2008). « MonetDB, a Novel Spatial Columnstore Dbms ».

In : *Academic Proceedings of the 2008 Free and Open Source for Geospatial (FOSS4G) Conference, OSGeo*. 00015, p. 193-199.

Ware, Colin (2012). *Information Visualization : Perception for Design*. 05270.

Elsevier.

Wickham, Hadley (2016). *Ggplot2 : Elegant Graphics for Data Analysis*. 00082.

Springer.

— (2017). « Tidyverse : Easily Install and Load ’tidyverse’ Packages ». In : *R package version* 1.1. 00113.

Wickham, Hadley et al. (2015). « Dplyr : A Grammar of Data Manipulation ».

In : *R package version 0.4* 3. 00547.

Wilkinson, Leland (2006). *The Grammar of Graphics*. 01037. Springer Science

& Business Media.

Zaamoune, Mehdi et al. (2013). « A New Relational Spatial OLAP Approach for Multi-Resolution and Spatio-Multidimensional Analysis of Incomplete Field Data ». In : *ICEIS 2013 INSTICC International Conference on Enterprise Information Systems*. 00010, p.

Zeng, Kai, Sameer Agarwal et Ion Stoica (2016). « iOLAP : Managing Un- certainty for Efficient Incremental OLAP ». In : *Proceedings of the 2016 Inter- national Conference on Management of Data*. SIGMOD ’16. 00012. New York, NY, USA : ACM, p. 1347-1361. doi : 10/gfw9mn. url : [http://doi.acm.](http://doi.acm/) org/10.1145/2882903.2915240.

Çöltekin, Arzu, Halldór JanetZko et Sara Fabrikant (2018). « Geovisuali- zation ». In : *Geographic Information Science & Technology Body of Knowledge* 2018.Q2. doi : 10.22224/gistbok/2018.2.6. url : https://gistbok. ucgis.org/bok-topics/geovisualization.

88