

Valaistat : E-Atlas du Valais

R. Bubloz, L. Martinez, N. Vallotton

Semestre d'Automne 2016

Résumé

Le présent rapport accompagne le rendu du cours de *Géovisualisation dynamique et traitement de données*. Il retrace les principales étapes et choix effectués, de la planification conceptuelle aux critiques du résultat en passant par le fonctionnement et les diverses technologies utilisées.

1 Introduction

Dans le cadre du cours de *Géovisualisation dynamique et traitement de données*, il a fallu réaliser un projet de communication d'une problématique spatiale de manière interactive. Le choix s'est porté sur l'Atlas du Valais disponible en ligne (e-atlas).

Cet atlas a été originellement développé en 2011 par une équipe de l'*Institut de Géographie* (ex-IGUL) de l'UNIL en collaboration avec l'EPFL. Le site fonctionne grâce au SIG *Géoclip* qui permet à la fois de visualiser les données sous forme de cartes (choroplèthes et symboles proportionnels) et de les télécharger sous formes de tableau. Il est disponible ici: eAtlas du Valais - Accueil.

Le choix est motivé par le fait qu'à l'échelle du numérique, les technologies évoluent rapidement et les modes passent. L'e-Atlas, qui repose sur du Flash n'a pas été actualisé depuis sa création et de nombreuses techniques de visualisation de données ont été développées depuis. Si de multiples possibilités s'offrent en termes de techniques, il convient d'adapter le choix au résultat visé. C'est pourquoi, afin de pouvoir sélectionner les éléments à incorporer et les technologies à utiliser, le travail a commencé par la définition des objectifs à atteindre.

2 Buts

Pour commencer, il est judicieux de définir un public cible pour éviter une trop grande simplicité ou au contraire, un trop grand nombre de fonctionnalités dans l'application. Si l'on considère la version actuelle de l'E-Atlas, on remarque que l'interface pourrait être plus intuitive, c'est pourquoi il a tout de suite paru important de rendre l'application plus "*user friendly*" et d'améliorer le sens de la navigation et de guider l'utilisateur.

C'est à ce moment que l'on peut redéfinir le public visé par cette application : alors que l'e-Atlas disposait de beaucoup de fonctionnalités pour un public plus enclin à faire l'effort d'accéder à ces fonctions, un recadrage vers un public plus néophyte semblerait plus pertinent.

3 Objectifs initiaux

L'eAtlas actuel a été étudié et les aspects jugés les plus importants ont été conservés. Parmi ceux-ci, on relèvera les cartes en symboles proportionnels colorés effectuées automatiquement au

choix de certains indicateurs, la possibilité d'afficher ou non certaines couches d'habillage (routes, cours d'eau principaux, toponymes) ou encore le fait de pouvoir exporter les données.

Si ces aspects ont été jugés intéressants, d'autres ont, au contraire, été désignés comme améliorables : design général (carton trop gros, organisation des différents boutons), difficulté d'accès aux données (rapports détaillés s'ouvrant sur une autre page) et esthétique.

Une fois ces étapes terminées, plusieurs autres sites qui permettent une représentation cartographique de données ont été visités pour s'inspirer d'éléments jugés pertinents ou alors peu pratiques. Parmi les sites consultés, on citera notamment :

- le portail cartographique de la Confédération Helvétique
- l'atlas statistique de l'Office Fédéral de la Statistique

Une ébauche des fonctionnalités à conserver et plusieurs maquettes ont été réalisées sur la base de cette réflexion. Le but serait de disposer des diverses fonctionnalités offertes par l'e-Atlas (indicateurs, cartes, graphiques) sur une seule page interactive. Les principaux apports seraient principalement graphiques, avec notamment une amélioration de la convivialité de l'interface et une mise en page personnalisable selon des pré-réglages (grande ou petite carte, avec ou sans graphique).

Le sommet de la page serait composé d'un *header* permettant de naviguer à travers les différentes pages du site. On y trouverait des boutons pour revenir à la page d'accueil, accéder aux deux types d'atlas statistiques (socioéconomique ou environnemental), obtenir des informations relatives au projet ou encore des liens externes vers les organismes à l'origine du projet.

Etant donné l'importance de la composante spatiale au sein d'un Atlas, la carte resterait l'élément principal de l'application, au centre de la page. Sur la gauche se trouverait un menu de navigation permettant d'accéder aux indicateurs statistiques. La partie de droite comporterait différentes représentations graphiques apportant des informations complémentaires relativement aux indicateurs sélectionnés. Enfin, le bas de la page serait réservé à des commentaires permettant d'agrémenter le contenu graphique de la page (synthèse de l'information, faits intéressants, compléments historiques ou politiques). Ces éléments ont inspiré la maquette de base (Figure 1).

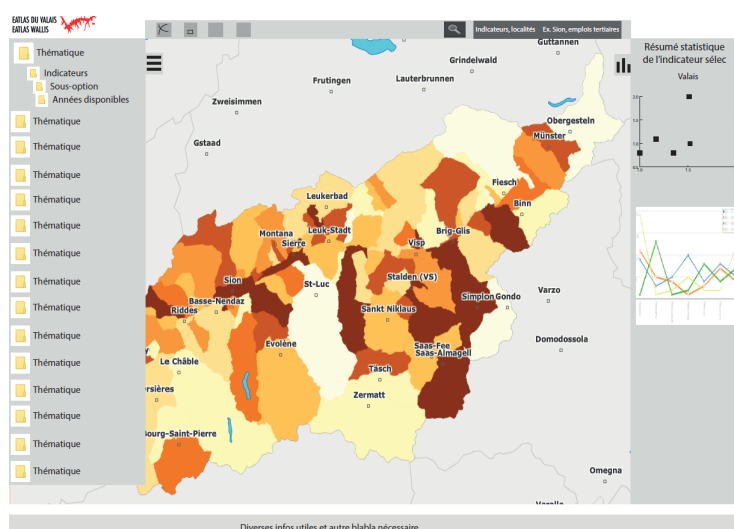


FIGURE 1 – Première maquette

L'une des principales nouveautés serait la possibilité de changer la mise en page en sélectionnant parmi plusieurs templates permettant de visualiser, *p.ex* :

- Une carte principale avec un ou deux graphiques (Figure 2)
- Deux cartes simultanément avec deux indicateurs différents sur des données longitudinales
- Une carte plus petite avec une série de graphiques, permettant de mieux analyser les éventuelles structures émergentes au sein d'une série d'indicateurs

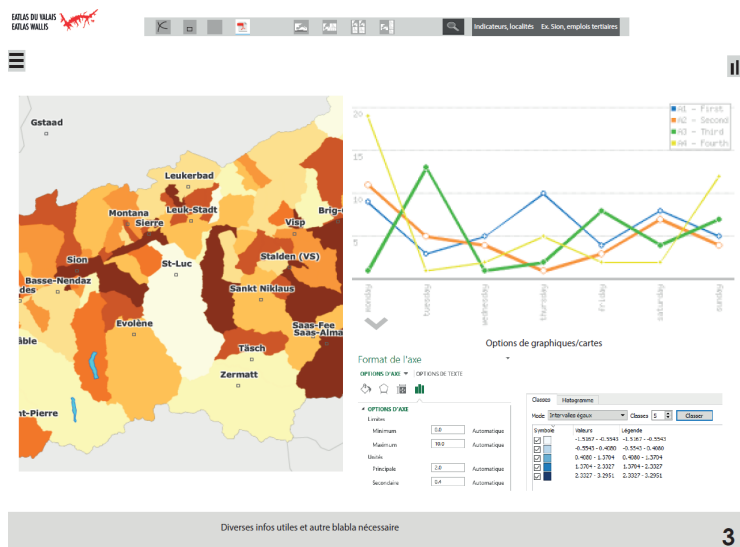


FIGURE 2 – Un des différents templates

Le fait d'offrir autant de possibilités de visualisation pose un certain nombre de complications relatives à la sélection des indicateurs à visualiser, notamment lorsque l'on affiche deux cartes avec des données différentes. Il s'agirait de trouver un compromis entre la flexibilité et la liberté de l'interaction d'une part, et l'anticipation envers les utilisations plus complexes d'autre part.

Au niveau de la carte, elle serait relativement similaire avec celle de l'e-Atlas actuel tout en améliorant son design. Il serait toujours possible de jouer avec l'habillage comme sur la version actuelle (affichage des routes, hydrographie, toponymes).

Pour ce qui est de la navigation au sein des données, elle respecterait l'architecture actuelle de l'e-Atlas (*domaine-thème-indicateur*) mais la visualisation serait différente; les menus déroulants seraient remplacés par une arborescence sous forme de dossiers, permettant une meilleure visualisation du contenu des différents domaines et thèmes simultanément. Dans l'idéal, cette barre de navigation serait rétractable pour gagner de la place sur la page lorsque les indicateurs sont sélectionnés.

Les graphiques constitueraient l'une des principales évolutions de l'e-Atlas : différents types de visualisations (*diagramme en barres, nuage de points, toile d'araignée, séries temporelles*) serviraient à compléter l'information disposée à l'aide de la carte et également à apporter des informations complémentaires afin de lier les indicateurs entre eux. L'objectif principal est d'apporter une information supplémentaire pertinente à ce que propose la carte (corrélation ou structure par typologie *p.ex*).

Maintenant que la direction initiale et les réflexions qui l'ont accompagnée ont été exposées, il est possible de s'intéresser à l'application telle qu'elle a été conçue au final. De noter son fonctionnement, ainsi que, surtout, les principales difficultés rencontrées au cours de son développement

4 Application et développement

L'étape de réflexion et décision a pu paraître longue, mais elle a permis de se faire une bonne idée de la démarche à suivre pour l'implémentation ainsi que pour le choix des technologies.

Malheureusement, le décalage entre les objectifs (très ambitieux) et nos capacités à vite pesé lourd dans l'élaboration du site. Il a donc fallu suivre une démarche d'implémentation progressive en se focalisant sur de petits éléments simples à réaliser étape par étape. Cette démarche, aussi appelée KISS (Keep It Simple, Stupid) a permis d'éviter la dispersion dans les différents aspects du site (mise en page, gestion des données, fonctionnalités, visualisation) et de mettre au point

chacun de ces éléments dans un ordre de priorité défini par nos objectifs révisés.

Ces derniers ont été adaptés à nos compétences et à l'amélioration de celles-ci au fur et à mesure de l'avancée du projet. On pourrait résumer leur importance par les listes suivantes :

1. Visualiser la carte du Valais
 - Carte choroplèthe
 - Carte en symboles proportionnels
 - Carte en symboles proportionnels colorés
2. Visualiser le graphique lié à la carte
 - Barres
 - Nuage de dispersion (X, Y)
 - Ajout de la taille au nuage de dispersion
 - Ajout de la couleur au nuage de dispersion
3. Travail sur les transitions pour les différents types de cartes et graphiques

Une fois le travail de visualisation opérationnel, la gestion des données a été abordée :

1. Création de la base de données (*BD*)
2. Réflexion sur l'architecture de la BD et les éléments nécessaires
3. Construction des différentes tables nécessaires et inclusions dans la BD
4. Travail pour récupérer les données dans la BD et les injecter sur le site

Puis, lorsque le site était approximativement fonctionnel, les divers aspects esthétiques ont été travaillés :

- Barre de navigation
- Arrangement des éléments
- Ajustement à la taille de la fenêtre du navigateur
- Présentation soignée et conviviale

La majorité de ces éléments ont des répercussions qui sont visibles par l'utilisateur (et, heureusement pour la motivation, le développeur également), mais la plupart des changements que l'on pourrait qualifier de significatifs sont en arrière-plan (*backend*). En effet, le choix d'utiliser une BD plutôt que d'utiliser des fichiers de type GeoJSON dans un dossier situé au même emplacement que les scripts a eu certaines conséquences sur le développement : familiarisation avec les BD, utilisation de *SQL*, formatage des données pour les insérer, ainsi que l'approfondissement de l'application faisant le lien entre BD et la page HTML.

On pourrait également citer les transitions qui ont nécessité de reprendre le code précédemment écrit et de l'adapter afin d'obtenir un changement d'état des différentes entités et non plus de procéder à un effacement et une création pur et simple d'une nouvelle carte et d'un nouveau graphique à chaque changement d'indicateur.

Certains éléments ont fortement agi sur nos objectifs initiaux, car après avoir développé des éléments simples, mais visuellement appréciables (cartes choroplèthes et en symboles proportionnels, graphique en barres), pendant la première phase de progression, l'accent a été mis sur des aspects internes donc invisibles, mais indispensables à l'implémentation judicieuse de l'application. Il s'est avéré difficile de montrer à nos camarades un réel avancement, alors que le visuel évoluait peu. Ces implémentations ont retardé l'avancement du développement selon le planning envisagé, ce qui a entraîné un report dans le temps (limité) de plusieurs étapes sur lesquelles nous reviendrons dans les sections 5 et 6.

A la suite de cette section, seront abordées les technologies utilisées pour la conception et la bonne marche du site, ainsi que pour le travail en groupe. La suite traite de la logique de fonctionnement du site et des éléments les plus notables de l'implémentation.

4.1 Technologies

Les technologies utilisées sont plutôt diverses, même si l'on y retrouve des langages de programmation connus et commun à ce genre de développement.

Pour ce qui est de la page web au sens restreint, le trio *HTML*, *CSS* et *Javascript* a été utilisé. Pour le *CSS*, *Bootstrap* a été utilisé pour faciliter la structure du site avec notamment la possibilité d'arranger la page en colonnes, ainsi que l'existence d'éléments pré-faits, facilement adaptables pour son propre site. Pour *Javascript*, deux librairies ont été mobilisées : *jQuery*, principalement pour la manipulation facile des éléments du DOM et la librairie *D3* (Data Driven Documents). Cette dernière est utile et puissante en matière de visualisation de données à l'aide du format *SVG*. Son champ d'application est large, de nombreuses possibilités existent (tableau, cartes, réseaux, horloges animées, etc.) et de multiples tutoriels sont disponibles en ligne sur les "blocks" de Mike Bostock, créateur de la librairie. À mentionner également, l'extension *D3 legend* qui permet de gérer les légendes de manière simple (même si la présentation pourrait être améliorée).

La base de données a été créée grâce à un système de gestion de base de données relationnelle libre : *PostgreSQL*, dérivé du langage *SQL*. Il fonctionne avec différentes extensions dont notamment *Postgis* qui permet de gérer des entités géographiques, ainsi que d'exécuter des requêtes spatiales. Pour travailler avec ces outils, il a fallu requérir au langage *SQL* tant pour la constitution de la BD, l'import des tables et les différentes requêtes.

Finalement, pour lier ces éléments, nous avons eu recours à Flask, un framework de développement web en *Python*. Flask a permis de procéder à des requêtes SQL dans la BD et d'attribuer les différents éléments récupérés aux URLs voulus.

Etant donné que l'application a été développée à plusieurs, certaines problématiques sont apparues : synchronisation des versions du code, communication des avancées et du programme envisagé, explication des manipulations réussies, travail à distance, etc. Nous avons donc utilisé un logiciel de gestion de versions décentralisé *Git*, qui permet de remplacer uniquement les fichiers modifiés et donc de pouvoir obtenir les dernières implémentations. Plusieurs sites hébergent des *repositories* et c'est *Bitbucket* qui a été privilégié, car qui permet d'avoir une version accessible à notre groupe uniquement. Toutefois, cela ne permettait pas de gérer les BD et il a fallu faire particulièrement attention de les maintenir à jour séparément grâce à des *dumps*. Pour le travail à distance, nous avons utilisé des outils de communications courants comme *Whatsapp* pour l'envoi de bouts de codes, ainsi que *TeamViewer* pour faire des vidéoconférences avec aperçu de l'écran en temps réel. Une feuille de route de type *log* a également été tenue à jour avec inscription des différentes réussites et objectifs encore en cours.

Pour la résolution des nombreux problèmes rencontrés, la console du navigateur *Chrome*, de nombreuses recherches sur *Google* et quelques questions/réponses sur le site communautaire *StackOverflow* ont été consultés.

4.2 Fonctionnement

Le principe de fonctionnement est plutôt simple, même si la logique derrière a mis du temps à se dessiner. Tout d'abord, au lancement de l'application, un certain nombre de requêtes SQL vont s'exécuter depuis le fichier *app.py* afin d'avoir les différents éléments de la BD (*statlas.sql* prêts et disponibles à des URLs prédéfinies (*routes*). Flask se charge également de charger *l'index.html* comme route par défaut (/).

Le script transforme ensuite les données et les géométries sous formes de tables dans la BD, pour les avoir en format *JSON* et pour ainsi être prêtes à être lues par le *Javascript* (Figure 3). Parmi les différents *JSON* créés, on citera la géométrie (contours et centroïdes), les différents indicateurs, ainsi que le fichier de style, qui contient les informations relative à l'indicateur sélectionné selon quelle représentation pour la carte et le graphique et quels indicateurs ont été préalablement associés pour ces représentations, titres, sources, etc.

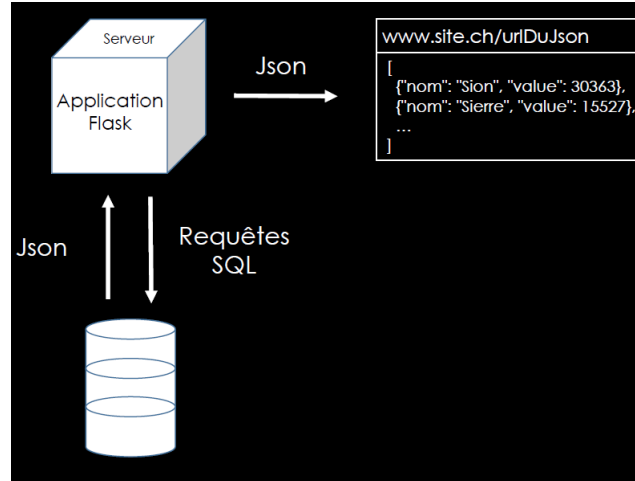


FIGURE 3 – Requête, transformation et affectation

La table de "style" est la partie la plus importante car elle conditionne toutes les visualisations, c'est donc la première partie exécutée dans le code à chaque fois que le domaine et/ou thème est modifié. Une fonction va actualiser les types de représentations ainsi que les indicateurs qui lui sont liés (et leurs URLs, pour plus tard charger les données) (Figure 4).

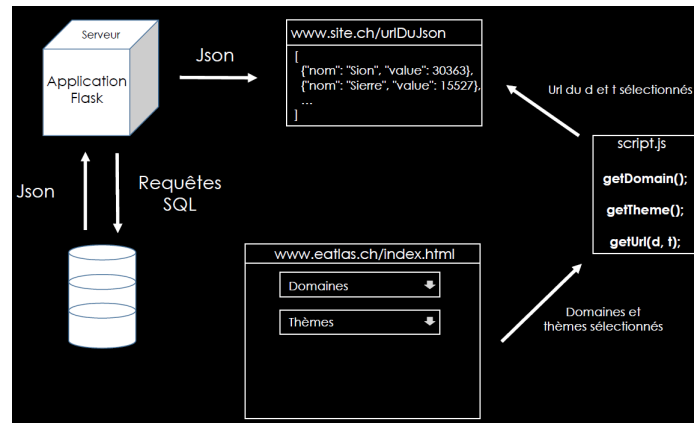


FIGURE 4 – Récupération d et t sélectionnés et accès à l'url contenant ces données

Les données sont ensuite récupérées et utilisées par les fonctions d'actualisation de la carte et du graphique. Ces dernières récupèrent les éléments présent sur la visualisation (communes, symboles proportionnels; barres, points du nuage, etc.) et effectuent une transition pour les amener dans l'état correspondant à la représentation sélectionnée (changement de couleur, taille, emplacement X ou Y, etc.) (Figure 5).

Ainsi, à chaque changement de domaine et/ou thème, il y a une répétition des étapes schématisées sur les Figures 4 et 5.

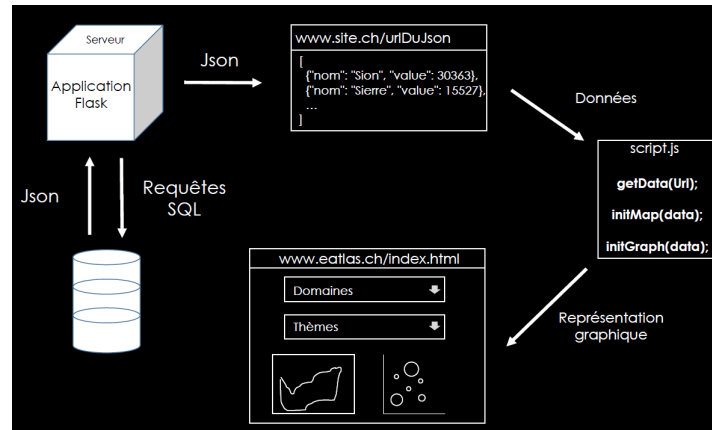


FIGURE 5 – Récupération des données et exécution de la visualisation (avec transitions)

4.3 Visualisation et choix

Comme mentionné auparavant, les imprévus qui se sont dressés en travers du développement ont fait que le travail a pris du retard. Il a donc fallu revoir nos objectifs initiaux, afin de proposer une page fonctionnelle. Repenser nos objectifs a permis de prendre du recul sur le projet et de réorienter notre géovisualisation pour un public cible plus novice dans le domaine de la géographie. Ceci couplé à une meilleure réflexion sur les "principes d'une application de géovisualisation efficace" a permis d'aboutir à une application moins conventionnelle comparée à des applications plus classiques.

Nous arrivons maintenant à la description de la visualisation de l'application telle qu'elle devrait paraître à l'heure actuelle (Figure 6).

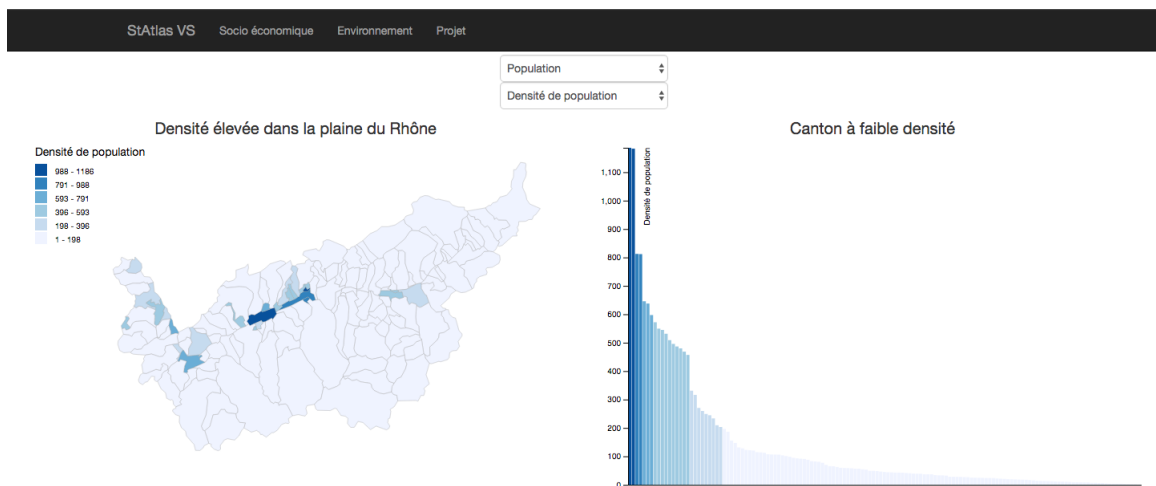


FIGURE 6 – Accueil de l'application

Lorsque la page est chargée avec les différents éléments correspondant, le domaine de la "population" avec comme thème "la densité de population" s'affichent par défaut. Le but ici de proposer un visuel par défaut est de ne pas laisser ce premier choix à un utilisateur impatient.

Il est possible de distinguer 3 parties composant la page :

- le bandeau supérieur afin de naviguer entre les différents domaines et thèmes proposés, ainsi que quelques liens vers des pages externes.
- la carte positionnée toujours à gauche avec son titre et sa légende.
- le graphique correspondant positionné à droite avec son titre.

La séparation de la page en 3 parties permet de garantir la lisibilité de l'application tout en dirigeant le regard de l'utilisateur par étape : sélection, carte, graphique.

En haut de la page au centre se trouve donc deux barres de menu déroulant (Figure 7) ; la première permet de sélectionner le domaine et la seconde de sélectionner le thème. Il faut noter que lorsque qu'un domaine est choisi, le premier thème de la liste est sélectionné par défaut. Tous les thèmes d'un domaine sont liés à une couleur (*p.ex* bleu pour la population) pour des raisons de compréhension et d'esthétisme. Concrètement, l'unique choix de l'utilisateur réside dans la sélection des domaines et thèmes dans les menus déroulant. L'ambition ici n'est pas freiner les possibilités offertes, mais de proposer une page épurée avec peu fonctionnalités pour éviter à un usager néophyte de se perdre dans la complexité. Avec ces deux menus, l'utilisateur remarque et comprend immédiatement les fonctionnalités mises à sa disposition.

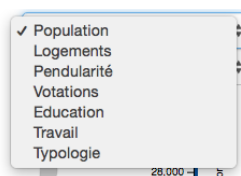


FIGURE 7 – Menu de sélection du domaine

A la sélection d'un thème, l'application va immédiatement récupérer à la bonne adresse les indicateurs nécessaires dans la base de données. Cette dernière est très réactive étant donnée que les indicateurs ont déjà été formatés et assignés par Flask. La page avec les visualisations se chargent donc de manière immédiate avec une transition. Le but était ici d'afficher en temps réel les informations sans que l'utilisateur doive attendre. Il faut noter que la BD n'a été testée que en local, donc nous ne savons donc pas si l'application souffrirait de latence dépendamment de la vitesse de connexion de l'utilisateur ou du serveur d'hébergement.

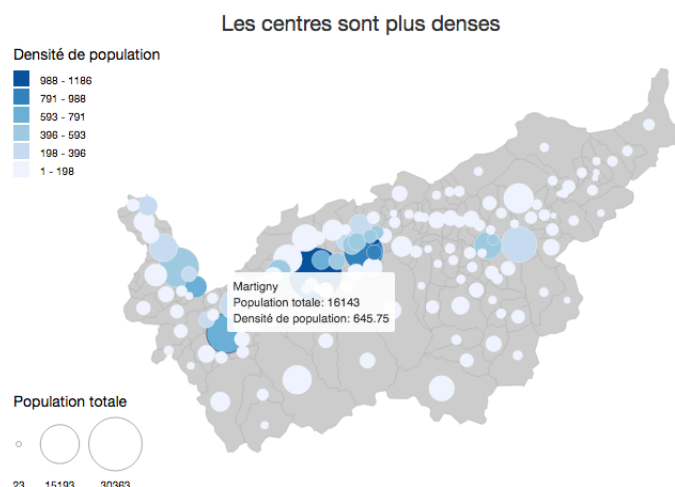


FIGURE 8 – Visualisation d'une carte bivariable avec un tooltip

Si l'on s'intéresse plus particulièrement à la carte, elle est toujours initialisée avec au minimum un fond de carte, un titre et une légende. Les trois types de cartes qu'il est possible de réaliser sont la carte choroplèthe, la carte en symboles proportionnels et la carte bivariable avec des symboles proportionnels colorés.

La principale interaction que l'utilisateur possède avec la carte est sa possibilité de passer la souris sur les différentes entités afin d'obtenir les données correspondant à la visualisation sous forme de *tooltip*. Sur la figure 8), il est possible de remarquer que lorsque l'entité est sélectionnée, le nom de l'entité va apparaître en *hover*, puis selon si la carte est construite avec un ou deux indicateurs, le *tooltip*, qui sert justement à afficher des données relatives à une entité, va afficher uniquement les données relatives liées. Cette technique permet d'éviter la surcharge graphique de la carte pour ne pas biaiser sa lisibilité tout en offrant la possibilité d'obtenir les informations nécessaires à la compréhension du phénomène visualisé.

La visualisation du graphique (figure 9) est possible sous forme :

- d’histogramme trié selon les couleurs de la carte (il est possible d’apprécier la mise en classe automatique de type *Jenks*).
- de nuage de points (simple avec deux indicateurs pour X et Y, ou plus complexes avec une troisième variable pour la taille des points et même une quatrième variable catégorielle pour la couleur des points).

Ces deux visualisations sont automatiquement choisies selon le thème sélectionné. Le but est ici de contraindre l’usager à un affichage par défaut pour orienter sa compréhension du phénomène lié aux indicateurs. C’est pour cette raison que la BD est formatée pour n’afficher que certaines visualisations selon le thème choisi. L’implémentation demande donc une expertise préliminaire d’un géographe pour décider ce qu’il est pertinent de visualiser pour chaque thème. La question du texte explicatif s’est posée lors du développement, mais pour garder un design simple et cohérent, il a été préféré de ne soumettre l’utilisateur qu’à des visuels lui permettant de mieux saisir un phénomène.

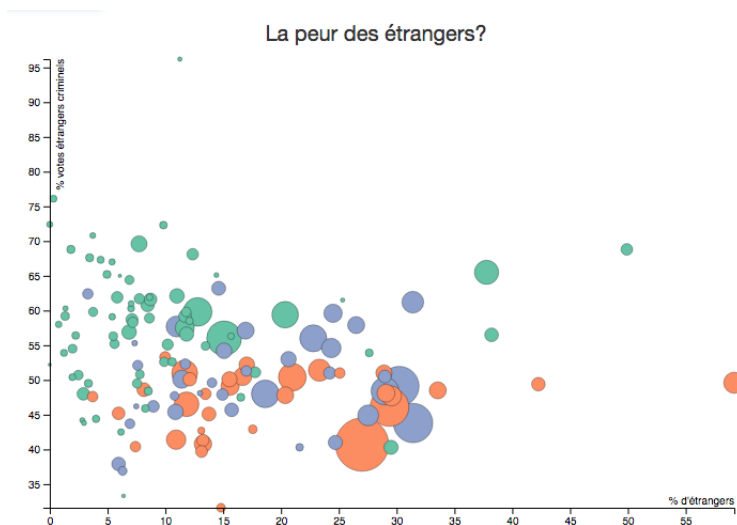


FIGURE 9 – Graphique nuage de points à 4 variables

Entre chaque visualisation de la carte et du graphique, des animations créées avec la librairie *D3* servent à guider l’œil de l’utilisateur. Outre leur aspect esthétique, les animations permettent de capter l’attention tout en permettant de faire un parallèle entre la première et la seconde visualisation. D’un point de vue technique, ces animations ont été optimisées par une cascade de conditions créant chaque type de visualisation en fonction de la précédente. Tous les éléments graphiques sont créés au lancement de la page mais sont invisibles (cercles de la carte de rayon nul ou rectangles du diagramme en barre hauteur nulle, *p.ex*), puis leur paramètres sont modifiés à chaque changement de visualisation. Ceci évite de recréer les éléments à chaque fois et permet par ailleurs de minimiser la duplication de code.

5 Tests et retours

Le site a été testé en phase de développement par plusieurs de nos collègues de géographie, ce qui ne correspond pas tout à fait au public cible actuel qui est un public plus novice, mais permet d’avoir un panel sensible à la représentation graphique. De ces retours est venue la volonté d’intégrer les transitions pour permettre de suivre l’évolution des entités entre deux indicateurs et de rendre le site plus "vivant". D’autres ont également mentionné qu’il pourrait y avoir un bref texte d’analyse de la situation présentée, à destination de personnes ne connaissant pas la région. Après avoir implémenté les transitions, les retours étaient unanimement positifs. Il a donc fallu demander plus directement pour savoir si elles semblaient nécessaires au final, ce qui a été confirmé par la suite.

L’application sous sa version finale a aussi été présentée à quatre personnes hors du domaine de la géographie. Le design minimaliste et les animations ont été les appréciations les plus positives. Les visualisations, que ce soit la carte ou le graphique ont été considérées comme "élégantes". Pour ce qui est de la compréhension, l’absence des unités sur les données et les titres raccourcis (afin d’éviter que le texte ne prenne trop de place) péjorent sur la lisibilité générale de l’information. Un point de repère (entités voisines, toponymes) par certains de ces utilisateurs. Pour l’un d’eux, la carte en symbole proportionnels colorés était peu explicite et difficile à interpréter, à l’inverse du graphique en nuage de points. Ceci démontre l’intérêt qu’il peut y avoir à varier les types de visualisation. Il semblerait aussi que sur des graphiques à plus de deux variables, les informations affichées grâce au *tooltip* ne permettent pas de comprendre aisément quelle donnée est liée avec quelle élément visuel. Enfin, des éléments d’explications sous forme de textes paraissent effectivement primordiaux pour une lecture correcte de la visualisation.

De manière générale, la singularité des fonctionnalités mises à disposition nous rend confortables dans l’idée que l’utilisation de l’application reste intuitive et compréhensible, même si elle ne permet pas toujours d’obtenir une compréhension/analyse optimale.

6 Possibilités d’améliorations

Le temps donné pour effectuer le travail étant limité, il a fallu faire des choix afin de ne pas s’égarer, car un tel projet est constamment perfectible. Il a donc fallu privilégier la fonctionnalité présente des avantages comme des inconvénients.

Il est donc possible d’inclure n’importe quel indicateur relatif aux communes valaisannes. Il est nécessaire en plus de l’ajouter à la suite de la table des indicateurs, de définir le "style" de l’indicateur (quelle représentation pour la carte et le graphique, lié à quel(s) indicateur(s), quel titre, quelles sources, etc.) et finalement de l’ajouter à la liste des thèmes existants du domaine correspondant ou alors de créer un nouveau domaine et de l’inclure dans sa liste de thèmes. Cette inclusion n’est pas nécessairement compliquée, elle demande par contre une réflexion en amont relativement aux choix des types de représentation ainsi qu’aux indicateurs liés, ce qui demande une connaissance du terrain et une analyse préliminaire.

Il serait possible de décomposer la table des styles (actuellement 12 colonnes) en plus petites tables que l’on pourrait séparer en fonction de leurs liens avec les principaux éléments graphiques :

- Carte (titre de la carte, style de représentation carte, indicateur relié pour la couleur ou indicateur relié pour la taille)
- Graphique (titre du graphique, style de représentation du graphique, indicateur relié pour la position Y, indicateur relié pour la couleur, indicateur relié pour la taille)
- Tooltip et Légende (libellé, labelTooltip)

En tant que géographes, l’absence d’échelle pour la carte peut déranger. Ce point nous paraît être une amélioration importante à amener par la suite. Ce genre d’éléments graphique n’est malheureusement pas programmable depuis la librairie *D3* à l’heure actuelle.

Toujours au niveau de la carte, il serait intéressant de pouvoir choisir d'afficher ou non d'autres couches telles que les surfaces productives pour montrer où sont les vallées, les principaux cours d'eau ou encore les principales routes et voies de chemin de fer à l'aide de boutons de sélection et que celles-ci se positionnent de manière optimale à la carte.

D'autres ajouts esthétiques, mais visuellement pertinents, consisteraient à utiliser une légende logarithmique pour certains graphiques, ce qui nécessite à nouveau d'avoir une connaissance du résultat visuel optimale et d'insérer de multiples conditions dans le code. Il faudrait également prévoir une échelle divergente dans le cas d'indicateurs à seuil externe qui modifie le sens (ex. votations (50%) ou croissances (0%). Cela contribuerait à une meilleure perception du résultat représenté.

Quant aux graphiques, d'autres types de représentation peuvent être envisagés, tels que les séries temporelles, ce qui nécessiterait l'ajout d'un sélecteur ou d'un curseur pour choisir l'année voulue. Des histogrammes ou encore un diagramme en barres des valeurs extrêmes qui apporterait plus d'intérêt selon l'indicateur sélectionné.

Il serait également possible d'effectuer divers traitements statistiques (ACP ou classification, *p.ex*) en amont et d'en proposer le résultat analysé, ce qui apporterait une plus-value intéressante.

7 Conclusions

Pour ce projet, nous avons appris à suivre et à concevoir un projet de géovisualisation de manière simulée de A à Z, tout en essayant de suivre un planning sans forcément s'y tenir. Il a fallu à plusieurs reprises réorienter et repenser notre projet en fonction de nos compétences. Ce n'est qu'une fois que l'on a "la tête dans le code" qu'il est possible de se rendre compte du niveau de complexité et de l'éventail des possibilités disponibles pour créer un atlas interactif. La découverte de nouvelles techniques de programmation et d'outils disponibles et la gestion de la "frustration" en cas de dysfonctionnement du code font partie des apprentissages significatifs effectués dans le cadre de ce projet. D'autres améliorations ne sont parfois même pas directement visibles, comme l'implémentation de la BD. Cela a néanmoins permis de travailler sur la flexibilité et de créer des scripts génériques qui pourraient être réutilisés avec d'autres indicateurs, voire même sur d'autres entités géographiques. A ces égards, il a fallu travailler étape par étape tout en se satisfaisant des petites réussites.

La plupart des conventions graphiques ont été respectées d'un point de vue sémiologique. On notera tout de même l'absence d'une échelle sur la carte et de sources des données, ainsi qu'un manque d'unités dans les légendes sur la version rendue. Construire une application web n'implique évidemment pas de mettre les règles de base de la cartographie sur un second plan, mais qu'il est impensable de foncer tête baissée sans lever la tête. Prendre un crayon et se poser la question du but recherché reste une étape nécessaire, quit à devoir faire des modifications conséquentes par la suite.

Ce projet a été développé à trois, ce qui peut être un avantage comme un inconvénient ; cela a permis à certains de se spécialiser dans certains domaines, mais aussi de confronter différents avis, parfois même divergents, sur le code comme le visuel. C'est cette complémentarité qui nous a permis de mener à bien ce projet malgré les objectifs fixés dans la phase préliminaire tout en sachant que la programmation fonctionne de manière itérative et qu'une telle application peut toujours être améliorée à l'infini.