



MISE À JOUR DE L'APPLICATION CARTOGRAPHIQUE DE LA CAISSE D'ALLOCATIONS FAMILIALES DE LA HAUTE-GARONNE

ATELIER MASTER SIGMA 2020



RÉALISÉ PAR :

OLIVIA BERNARDOFF
JULIE DAUVERGNE
TOMMY FERNANDEZ
NICOLAS SPECKLIN

POUR :

CAF 31



Table des matières

Introduction	3
I. Méthodologie de travail.....	5
II. Présentation des données et outils du SIG de la CAF Haute-Garonne	7
1. Données disponibles	7
2. Fonctionnement du site cartographique de la CAF	7
III. Choix méthodologiques	9
1. Mise à jour des indicateurs statistiques	9
2. Mise à jour des géométries.....	10
IV. Agrégation des indicateurs dans la base de données avec la méthode SQL	12
1. Aperçu des cas de figure	12
2. Assignation des COG correspondant	14
i. Méthode par comparaison avec le fichier de l'INSEE	14
ii. Méthode par comparaison de centroïdes	16
V. Formulaire de mise à jour des indicateurs statistiques	18
1. Mot de passe.....	18
2. Import du fichier de mouvement de COG	18
3. Formulaire	19
4. Action	19
VI. Mise à jour des géométries des communes via QGIS.....	21
VII. Proposition d'une configuration de serveur adapté à l'hébergement du site	23
1. Configuration du serveur actuel	23
2. Cas d'utilisation.....	24
3. Limites du site	24
4. Evaluation des besoins de migration vers PostgreSQL/PostGIS	28
VIII. Perspectives d'améliorations.....	30
Références bibliographiques	31
ANNEXES	32
Annexe 1 Liste des champs de la table des évènements survenus aux communes (Source : Insee)	32
Annexe 2 : Résultats des tests de performance.....	33

Table des figures

Figure 1 : Diagramme de Gantt du projet 1/2	5
Figure 2 : Diagramme de Gantt du projet 2/2	6
Figure 3 : Architecture du site.....	7
Figure 4 : Carte thématique du nombre de famille par commune en Haute-Garonne en 2018. L'utilisateur peut choisir avec une liste déroulante la famille d'indicateur, le nombre de famille, la zone géographique, le niveau géographique et l'année des valeurs (dans cet ordre). Cliquer sur une commune permet d'ouvrir un popup qui va renseigner la valeur de l'indicateur.	8
Figure 5 : Méthode par comparaison des centroïdes. Les tables avec les indicateurs posséderaient deux colonnes correspondant aux coordonnées des centroïdes (rouge) qui seraient comparées aux polygones des géométries actuelles (vert) afin d'additionner les indicateurs pour lesquels les centroïdes seraient dans la même commune.	10
Figure 6 : Résumé des deux méthodes sélectionnées pour répondre au cahier des charges. La correspondance entre les tables "bca_20xx" et le fichier de mouvement des COG de l'INSEE permet de créer de nouvelles tables des indicateurs agrégés "v_bca_20xx" qui permettent l'affichage des bonnes valeurs. L'outil QGIS de son côté utilise aussi le fichier de mouvement des COG pour la mise à jour des géométries.	11
Figure 7 : Structure de la BD locale PostGIS pour les tables de 2015.....	12
Figure 8 : Page permettant d'importer le fichier de changement des communes dans la base de données.....	18
Figure 9 : Formulaire rempli	19
Figure 10 : Fonctionnement de la mise à jour des indices statistiques. Les rectangles bleus sont les quatre fichiers *.php avec leurs caractéristiques dans les cercles blancs.....	20
Figure 11 : Description du modèle de traitements.....	22
Figure 12 : Diagramme des cas d'utilisation pour un utilisateur.	24
Figure 13 : Schémas des cas retenus pour les tests à différents débits.	25
Figure 14 : Schémas des cas retenus pour le test de la capacité du serveur.....	26

Table des tableaux

Tableau 1 : Aperçu des changements possibles dans les tables des communes	13
Tableau 2 : Aperçu du nombre de communes fusionnées par département par année.	14
Tableau 3 : Exemple de fusion de commune.	14
Tableau 4 : Script SQL permettant l'agrégation des valeurs.....	15
Tableau 5 : Temps de chargement des fichiers ajax en fonction du nombre d'utilisateur.	27
Tableau 6 : Poids des fichiers téléchargés afin de tester la bande passante.....	27
Tableau 7 : Comparaison du temps de chargement entre le serveur de la CAF et le serveur test de l'Université.	28

Introduction

La Caisse d'Allocation Familiale (CAF) de la Haute-Garonne possède un site internet permettant de visualiser l'évolution d'indicateurs sociaux de 2009 à aujourd'hui. Ces données peuvent être représentées à différentes échelles géographiques : par EPCI, par commune ou par quartier en utilisant un fond de carte de l'année actuelle. Un problème évident est alors mis en avant : comment sont représentées les données des années antérieures sur des fonds de cartes actuels ? En effet, d'une année sur l'autre, on constate des fusions de communes, effaçant les données des communes fusionnées et faussant les indicateurs affichés.

La France encourage à la fusion des communes depuis plusieurs années. Par exemple, la loi Marcellin du 16 juillet 1971 avait pour but de faciliter la fusion des communes. Seulement, elle n'a pas eu l'effet attendu. En effet, même si dans un premier temps des communes ont fusionné, elles se sont séparées par la suite. Il faut attendre la loi de 2014 pour observer plus de fusions en France qui comptait à l'époque 36 767 communes dont 86 % avec moins de 2 000 habitants [[Sénat](#)]. Cette loi permet d'améliorer le régime de la commune nouvelle en facilitant la procédure de fusion et apporte des avantages à la commune qui fusionne. Les anciennes communes deviennent des communes déléguées, permettant au maire de l'ancienne commune de garder le statut de maire délégué [[vie-publique](#)]. Depuis 2016, 317 fusions ont eu lieu sur le territoire Français [1].

Encore de nos jours, « les Systèmes d'Information Géographique (SIG) peinent à devenir de véritables outils d'aide à la décision. En effet, pour aider au suivi, à l'analyse et à la compréhension des phénomènes qui se jouent sur les territoires, il est indispensable de pouvoir étudier conjointement la dimension géographique et la dimension temporelle » [5]. Or, notamment à cause de la complexité technique de la modélisation temporelle, les SIG proposent souvent une représentation biaisée ou incomplète.

Dans ce contexte, la CAF de la Haute-Garonne, représentée par Mme Elodie Francès, et le master de géomatique SIGMA de l'Université Jean Jaurès et de l'ENSAT se sont associés pour prendre en compte ces variations territoriales et présenter des cartes statistiques correctes quelle que soit l'année affichée. Cette commande fait suite à 12 ans de collaboration ce qui permet une mise à jour régulière du site cartographique de la CAF 31 (<http://www.sig-cafmipy.fr/>).

Lors de notre travail, nous nous sommes focalisés uniquement sur la gestion des fusions entre communes. En effet, comme le prouve le document des mouvements des Codes Officiels Géographiques (COG) en région Midi-Pyrénées aucune scission n'a été enregistrée à ce jour ([accéder ici au document](#)).

Le cahier des charges nous impose de répondre à deux problèmes techniques :

- Le premier porte sur l'**automatisation de la fusion géométrique des communes** ainsi que la création de leur **centroïde**. Nous gérons uniquement les fusions contrairement au travail réalisé par Christine Plumejeaud [3] .
- Le deuxième traite la **mise à jour des indicateurs statistiques** des années passées sur les géométries de l'année actuelle.

Les données utilisées dans cet atelier sont fournies par la CAF. Les données de mouvements des communes proviennent uniquement de l'INSEE et permettent d'éviter le problème évoqué par Christine Plumejeaud et Jérôme Gensel liés à la non unicité des valeurs pour une même donnée [4] .

Par ailleurs, nous avons cherché à mettre en place des solutions qui permettent de gérer les changements de structure des fichiers d'entrées.

Dans ce rapport, nous allons tout d'abord exposer les données et outils disponibles ainsi que les méthodologies de travail. Nous présenterons ensuite les requêtes SQL utilisées pour traiter la mise à jour des indicateurs statistiques ainsi que le formulaire associé. Nous traiterons par la suite la mise à jour automatisée des géométries des fonds de carte. Enfin nous détaillerons certains tests de performance du site de la Caf effectués sur le serveur test de l'Université, dans le but d'évaluer l'intérêt d'une possible migration du site vers un autre hébergeur et un autre moteur de base de données.

Liste des livrables :

- une application web d'upload des changements de COG et de mise à jour des tables statistiques, construite autour de quatre fichiers PHP :
 - o **mdp.php** : page d'authentification à l'application ;
 - o **upload.php** : page d'importation du fichier COG ;
 - o **formulaire.php** : à la fois page d'exécution de l'upload et formulaire de mise à jour des statistiques ;
 - o **action_formulaire.php**: page d'exécution du formulaire.
- un fichier de type model builder contenant les séries de traitements à réaliser sous Qgis pour mettre à jours les géométries des communes.
- un guide d'utilisation et de maintenance à destination des agents de la CAF 31.

I. Méthodologie de travail

Le travail réalisé s'est déroulé sur 7 semaines et peut être découpé en 3 phases, à savoir :

- une 1ère phase d'appropriation du sujet avec de la recherche bibliographique et des séances de décryptage du script du site de la caf. A ce titre, nous avons réalisé une carte mentale que vous pouvez visualiser [ici](#) ;
- une 2nd phase de développement des outils qui fait suite à une réunion de mise au point avec la CAF. Cette réunion nous a permis de mieux caractériser leur besoin au regard de nos recherches préliminaires et de la commande initiale. Le choix des outils de travail a été fixé lors de la réunion de suivi du 4 février avec notre superviseur M.Jégou ;
- une 3ème phase de finalisation des outils qui fait suite à une réunion de démonstration des prototypes à l'équipe de la CAF. Cette réunion nous a confirmé l'adéquation entre les outils développés et les besoins de la CAF. La dernière phase a aussi consisté à réaliser une batterie de tests de performance du site et d'écrire les différents livrables (guide d'utilisation & de maintenance).

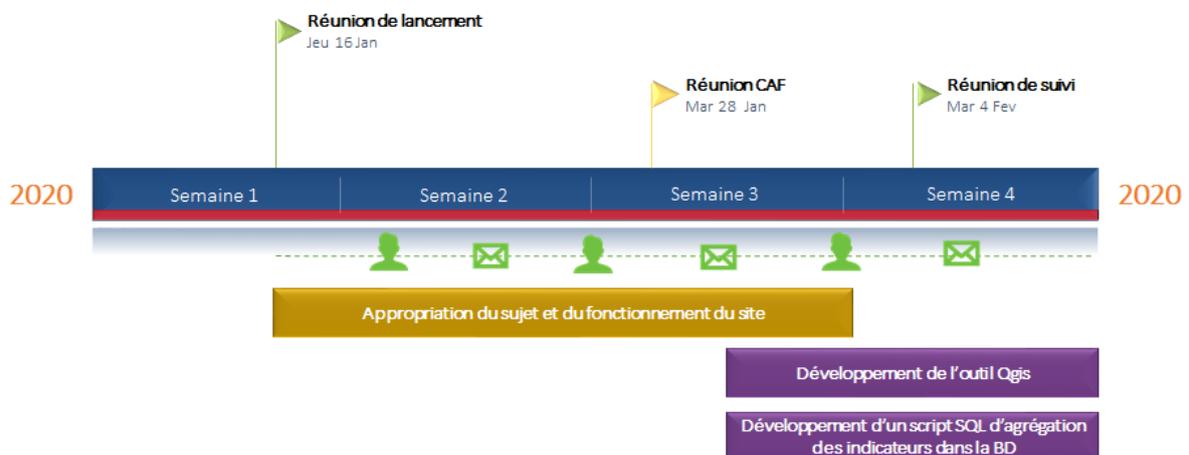


Figure 1 : Diagramme de Gantt du projet 1/2

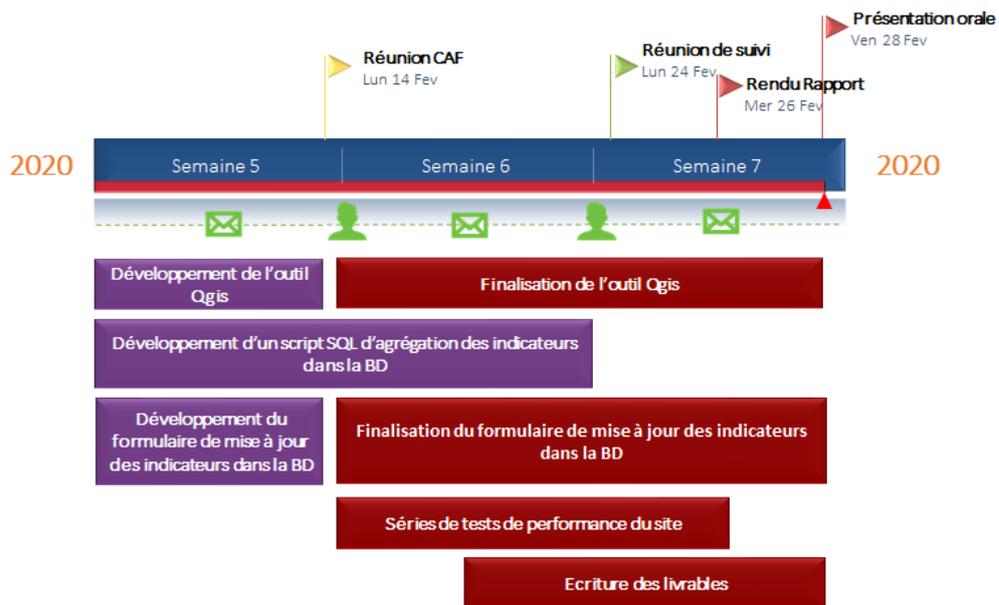


Figure 2 : Diagramme de Gantt du projet 2/2

Afin de pouvoir réaliser nos différents tests, nous avons eu à notre disposition une copie du site de la CAF.

II. Présentation des données et outils du SIG de la CAF Haute-Garonne

1. Données disponibles

La CAF de la Haute-Garonne met un site cartographique à disposition de toutes les communes de Midi-Pyrénées et leurs partenaires afin qu'ils puissent faire un suivi des statistiques des allocataires de la CAF sur leur territoire.

Tous les ans, les agents de la CAF reçoivent les fichiers des statistiques de la CNAF. Ils mettent ensuite à jour tous les ans les géométries des communes au format GeoJSON.

Les fichiers GeoJSON sont stockés sur un serveur cloud tandis que Les statistiques des années 2009 à 2018 se trouvent dans une base de données MySQL (avec l'extension MariaDB). Ces tables sont nommées "bca_20xx".

De plus, l'INSEE met tous les ans à disposition le fichier des mouvements du COG qui sera utilisé tout au long de notre démarche.

2. Fonctionnement du site cartographique de la CAF

Lorsqu'un utilisateur consulte le site cartographique de la CAF, il doit sélectionner une série de champs : famille d'indicateurs, indicateurs, zone géographique, niveaux géographiques, année. En fonction des critères choisis, le serveur enverra une liste de valeurs à afficher par le client sur le contour du niveau géographique sélectionné (exemple dans la Figure 3).

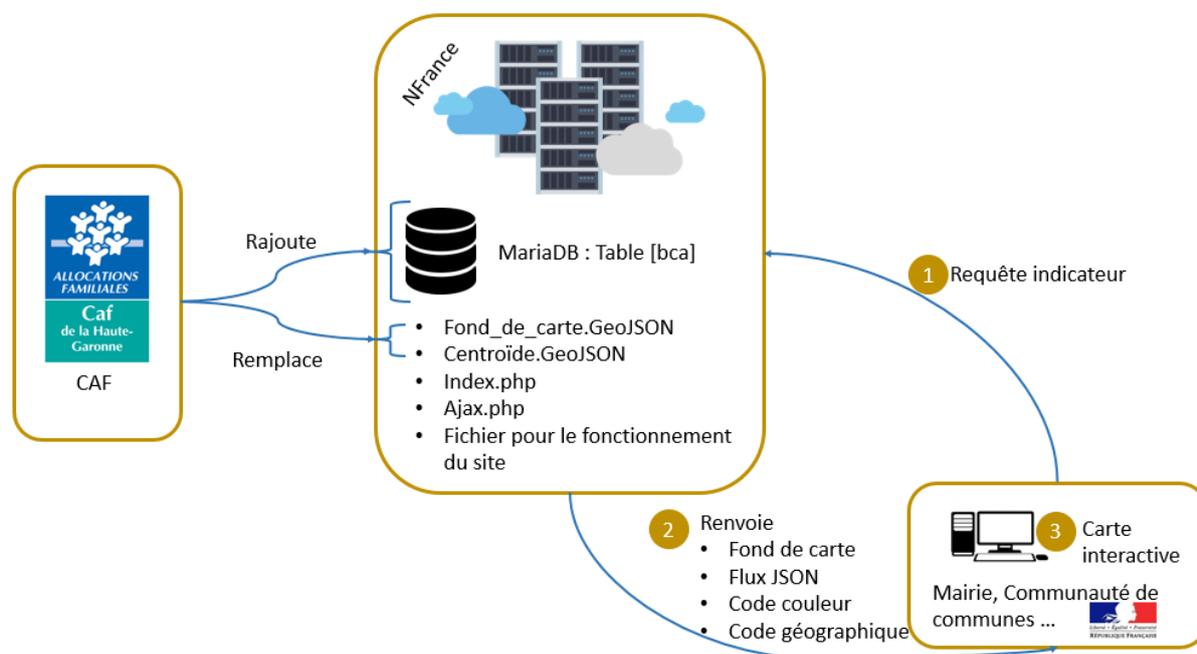


Figure 3 : Architecture du site

Le navigateur va ensuite réaliser une carte thématique avec un symbole adapté à la nature de l'indicateur demandé (cercles proportionnels pour les nombres bruts et choroplèthes pour les taux).

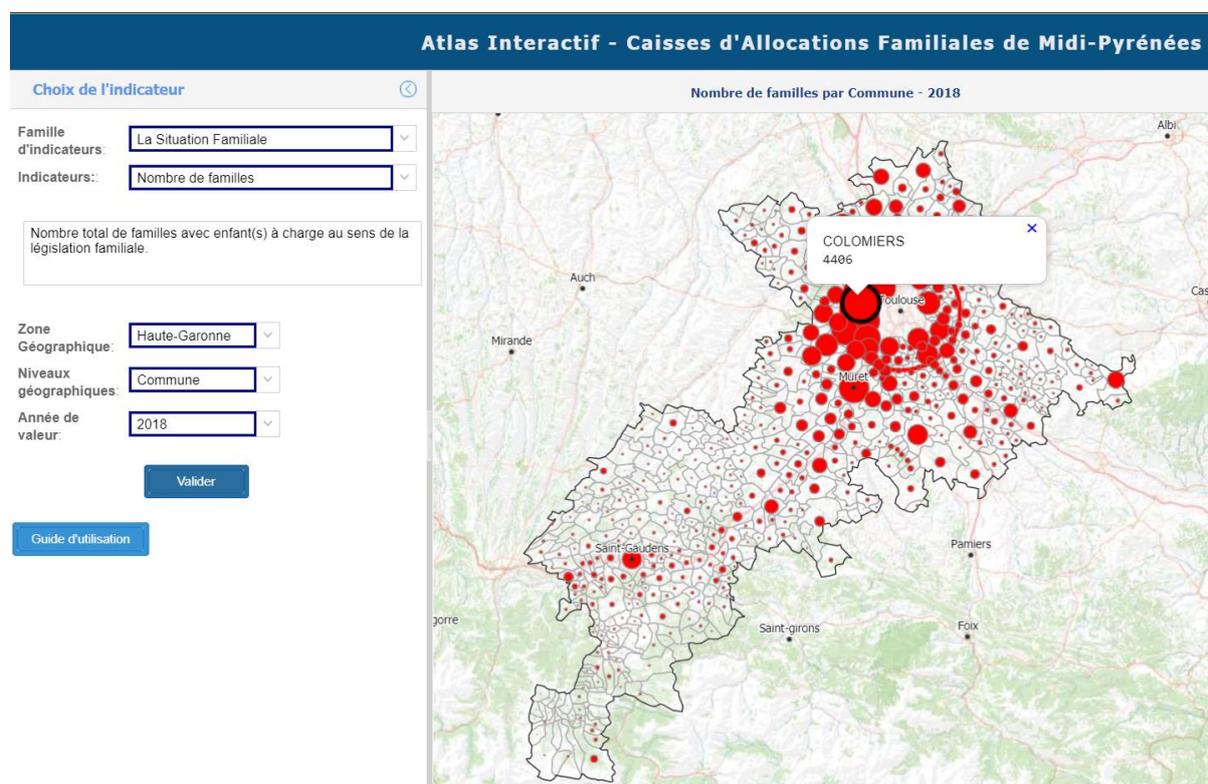


Figure 4 : Carte thématique du nombre de famille par commune en Haute-Garonne en 2018. L'utilisateur peut choisir avec une liste déroulante la famille d'indicateur, le nombre de famille, la zone géographique, le niveau géographique et l'année des valeurs (dans cet ordre). Cliquer sur une commune permet d'ouvrir un popup qui va renseigner la valeur de l'indicateur.

Trois différents types de symboles sont présents sur le site :

- des choroplèthes : représentés avec un aplat de couleurs pour les indicateurs de type taux ;
- des cercles proportionnels (Figure 4) pour des valeurs quantitatives ;
- des cercles proportionnels remplis avec un aplat de couleur pour afficher les deux types simultanément.

III. Choix méthodologiques

1. Mise à jour des indicateurs statistiques

La tâche principale de notre projet consiste à adapter les statistiques de la CAF aux géométries des communes de l'année en cours. Cette tâche comporte deux problématiques :

- d'un côté celle de la nature des changements de commune et les correspondances de COG ;
- de l'autre celle de l'affichage des données antérieures sur les fonds de carte de la dernière année.

Pour chacune de ces problématiques, nous avons exploré plusieurs pistes de solution. Pour traiter les cas de changement des géométries des communes, les pistes suivantes ont dans un premier temps été envisagées :

- l'exploitation de Rstudio via le package COGugaison (traitement côté serveur) ;
- la méthode de comparaison des centroïdes des communes des années passées sur les polygones des communes de l'année la plus récente (traitement côté serveur) ;
- l'utilisation de la bibliothèque Turf.js qui aurait permis de fusionner les indicateurs et les géométries en une seule fois côté client (traitement côté client) ;
- l'utilisation de requêtes SQL (traitement côté serveur). C'est cette solution qui a été retenue et fait l'objet de la partie IV ;

Le package R COGugaison permet d'accéder aux mouvements des communes de France à différentes dates. Cette solution a été abandonnée car la prise en main pour générer un tableau exploitable est complexe et que le fichier CSV disponible sur le site de l'INSEE contient toutes les informations dont nous avons besoin. Le package R COGusaison ne présentait donc pas d'avantage.

La méthode par comparaison de centroïdes nous aurait permis de créer plus simplement les correspondances entre COG antérieurs et actuel qu'avec la table de l'INSEE (Figure 5). Toutefois cette méthode requiert de garder les centroïdes des communes de chaque année dans la base de données. La méthode avec la table des mouvements de communes, quant à elle, permet d'obtenir le même résultat sans avoir à générer une dizaine de tables supplémentaires.

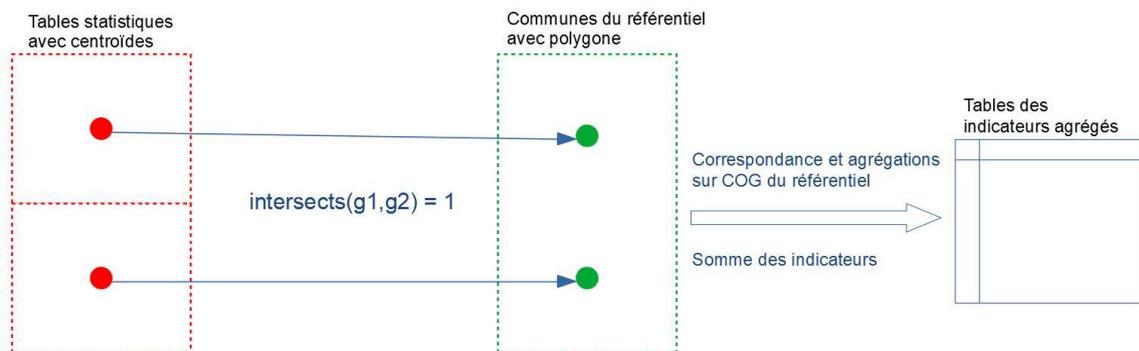


Figure 5 : Méthode par comparaison des centroïdes. Les tables avec les indicateurs posséderaient deux colonnes correspondant aux coordonnées des centroïdes (rouge) qui seraient comparées aux polygones des géométries actuelles (vert) afin d'additionner les indicateurs pour lesquels les centroïdes seraient dans la même commune.

Ces deux méthodes amènent la question de l'affichage des données sur les fonds de carte. Une première alternative envisagée aurait été de conserver les fonds de carte de l'année de référence dans la base MariaDB pour générer les géométries dans le même flux que les indicateurs statistiques à travers une requête SQL transcrite en JSON. Le site est cependant conçu pour précharger les fonds de cartes à partir des fichiers hors de la base de données ce qui épargne justement au serveur BD le calcul des géométries et permet un affichage plus rapide grâce à leur mise en cache dans le navigateur du client.

Enfin, nous avons également considéré la solution de la bibliothèque Turf.js de traitements spatiaux en JavaScript. Cette bibliothèque aurait permis de réaliser l'intégralité des traitements par le client, c'est-à-dire, trouver les correspondances de COG par comparaison géométrique, combiner les polygones des communes fusionnées, et afficher les indicateurs additionnés. Cette solution signifie cependant que ces traitements ne sont jamais faits sur le serveur mais toujours par le client à chaque requête d'un indicateur.

Suite à une réunion avec les commanditaires, il est apparu que la mise à jour des fonds de carte communaux et l'ajout des statistiques de la nouvelle année est effectué qu'une fois par an. Il était donc plus logique de choisir une solution qui modifie le moins possible le fonctionnement actuel du site et qui prépare une fois par an les données correctement agrégées sur les COG correspondants à ceux des fonds de carte les plus récents. Une fois les nouvelles tables statistiques créées à partir des tables présentes et du mouvement des COG (Figure 6), le site va requêter les données selon son fonctionnement habituel. Il n'y a pas d'ajout de temps de calcul côté client ou côté serveur ce qui aurait été le cas avec les traitements dans Turf.js ou avec les géométries intégrées à MariaDB. Nous avons alors fait le choix de créer une nouvelle page internet dont la fonction est de mettre automatiquement les tables à jour et qui intègre les requêtes SQL dans un formulaire web pour simplifier au maximum cette tâche.

2. Mise à jour des géométries

Les commanditaires nous ont également fait part de leur difficulté à mettre les GeoJSON des communes à jour chaque année. Nous avons donc proposé un outil de traitement spécialement créé sous le modeleur graphique de QGIS afin de pouvoir automatiser cette tâche qui est effectuée qu'une seule fois par an (Figure 6). Cette solution nous a paru la plus adaptée compte tenu du fait que les agents de la CAF 31 savent déjà manipuler le logiciel et pourront donc plus aisément prendre en main l'outil et y apporter eux-mêmes des modifications. Une autre option aurait été de concevoir

un script python qui aurait été plus malléable lors de la conception des traitements, toutefois, cela aurait demandé aux agents d’être formés sur ce langage de programmation.

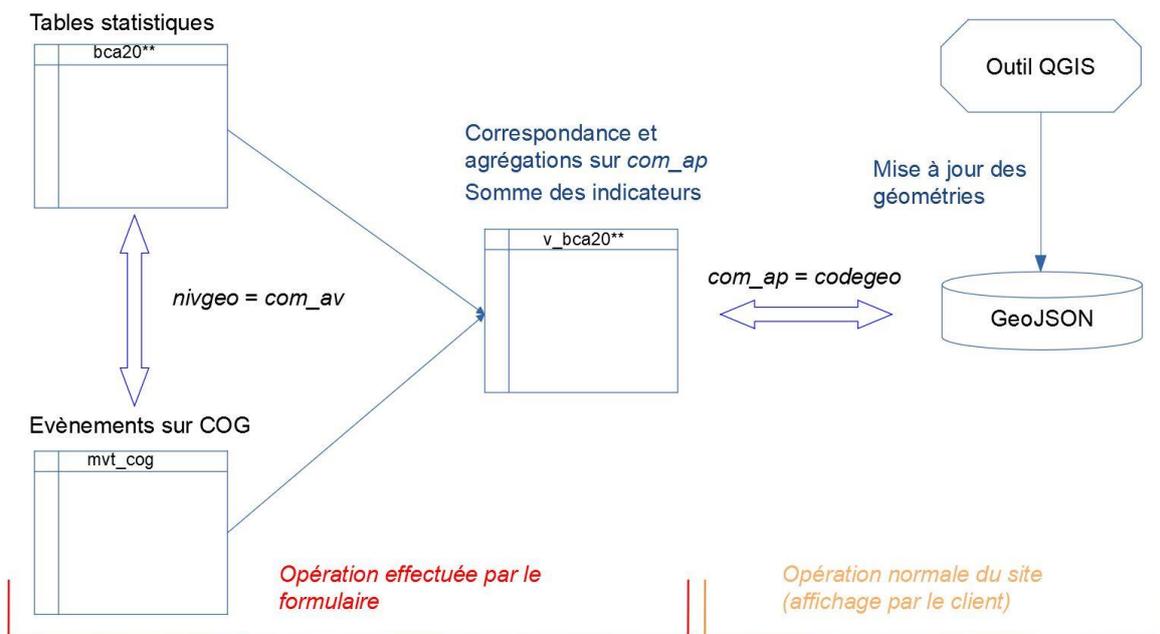


Figure 6 : Résumé des deux méthodes sélectionnées pour répondre au cahier des charges. La correspondance entre les tables “bca_20xx” et le fichier de mouvement des COG de l’INSEE permet de créer de nouvelles tables des indicateurs agrégés “v_bca_20xx” qui permettent l’affichage des bonnes valeurs. L’outil QGIS de son côté utilise aussi le fichier de mouvement des COG pour la mise à jour des géométries.

IV. Agrégation des indicateurs dans la base de données avec la méthode SQL

1. Aperçu des cas de figure

La démarche générale pour afficher correctement des indicateurs statistiques par commune des années passées sur la géométrie communale de l'année la plus récente est de créer une correspondance entre les anciens et les nouveaux COG pour garantir une jointure correcte entre les valeurs attributaires de la base de données et les couches vecteurs de l'application web (GeoJSON). La première étape consiste à créer un aperçu des cas de changements de COG survenant dans le jeu de données de la CAF, car les opérations à effectuer sur les indicateurs dépendent de la nature des changements. Comme mentionné dans le rapport de stage d'E. Catelin (2018) [1], nous avons généré des tables de correspondances afin de détecter des apparitions ou des disparitions de commune. Alors que Catelin comparait chaque jeu de commune les uns aux autres, les tables de données statistiques de la CAF sont toutes comparées tour à tour au même référentiel, à savoir les COG contenus dans le jeu de géométries de l'année la plus récente.

Afin de pouvoir travailler hors ligne et avec des tables géométriques plus facilement, les tables statistiques "bca20xx" ont été téléchargées de MySQL et ré-importées dans une BD PostGIS locale (figure 7). De même, les fichiers *.geojson en ligne sur le site ont été chargés dans QGIS, fusionnés (outil "merge vector layers") et importés dans PostGIS afin d'avoir dans une seule table le COG et la géométrie correspondante pour les huit départements de la région.

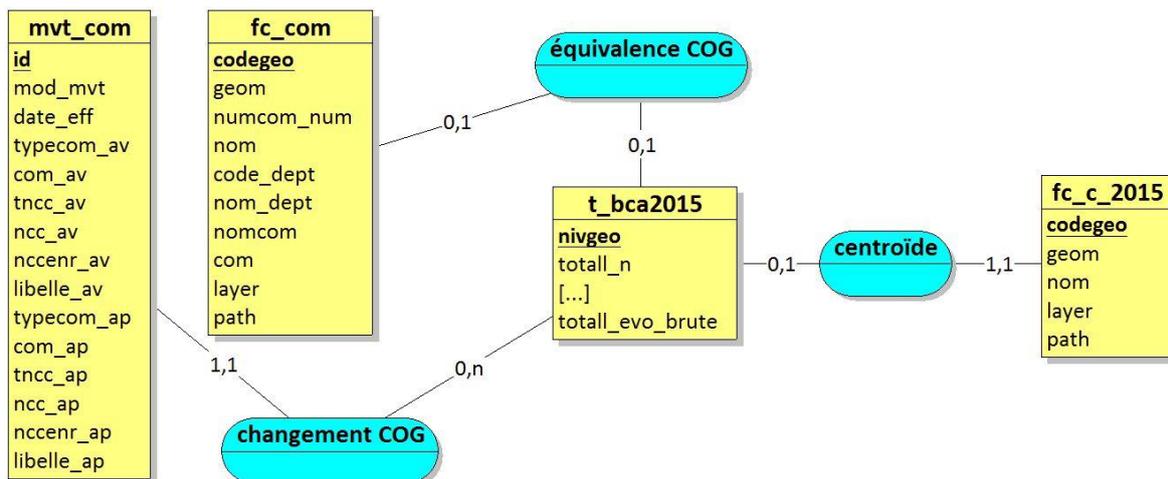


Figure 7 : Structure de la BD locale PostGIS pour les tables de 2015.

La requête suivante a été exécutée pour chaque table statistique :

```
SELECT t_bca2017.nivgeo,
fc_com.codegeo
FROM sigmacaf.fc_com
```

```

FULL OUTER JOIN sigmacaf.t_bca2017
ON fc_com.codegeo = t_bca2017.nivgeo ::VARCHAR
WHERE t_bca2017.nivgeo::VARCHAR LIKE '31%'
AND (t_bca2017.nivgeo IS NULL
OR fc_com.codegeo IS NULL)

```

Où “t_bca2017” est la table de statistiques pour l’année 2017 et “fc_com” la table contenant les géométries (2018). Cette requête permet d’obtenir un aperçu des cas de changements de communes par département (ici la Haute-Garonne) résumé dans le tableau 1. Le résultat est une juxtaposition des COG passés et présents dans les géométries lorsque ceux-ci sont égaux, en affichant également ceux qui n’existent que dans l’une ou l’autre table.

Administrativement parlant, les changements sur les communes peuvent être de nature variée : création, rétablissement, fusion simple, fusion association (INSEE, 2020). Comme indiqué dans le tableau 1, tous ces changements se manifestent géographiquement en deux cas possibles à traiter pour nous :

- l’apparition d’une commune dans l’année de référence qui n’existe pas dans la table statistique d’une année précédente (défusion ou scission) : ce qui signifie qu’une géométrie n’aura pas d’indicateur à afficher sur les années antérieures à la dernière année du jeu de données ;
- la disparition dans l’année de référence d’une commune contenue dans une table statistique (fusion): ce qui signifie que l’indicateur choisi n’aura pas de commune correspondante pour être affiché sur le fond de carte.

Après avoir exécuté ces requêtes sur tout le jeu de données, nous constatons qu’il n’y a jamais eu d’évènement de type défusions. De plus, lors de notre première réunion avec les commanditaires du projet, ceux-ci ont émis le souhait de laisser vide en valeur les communes issues d’une défusion lorsque l’on requête des indicateurs antérieurs au fond de carte. Le site web de la CAF a été programmé pour anticiper ce cas de figure et une commune sans données à afficher sera “grisée” et portera la mention “ns”. Nous avons donc seulement traité le cas des fusions de commune dans le développement de notre application.

Tableau 1 : Aperçu des changements possibles dans les tables des communes

Nivgeo (COG de la table statistique)	Codegeo (COG du référentiel géométrique)
31298	NULL
NULL	31999 (exemple fictif)

Le premier cas est celui d’une fusion de commune où le COG antérieur ne trouve pas d’équivalent dans l’année en cours. Le second cas (jamais observé dans le jeu de donnée de la CAF) est celui d’une défusion de commune, avec une nouvelle géométrie dont le COG n’existe pas dans les années statistiques précédentes. Le tableau 2 est le résultat des évaluations sur tous les tableaux de données, pour les huit départements.

Tableau 2 : Aperçu du nombre de communes fusionnées par département par année.

Référentiel géométrique		BCA 2017		BCA 2016		BCA 2015	
Dépt.	Communes	Communes	Fusions	Communes	Fusions	Communes	Fusion
31	586	587	1	588	2	589	3
46	313	320	7	322	9	326	13
12	285	285	-	285	-	286	1
32	461	462	1	462	1	462	1
9	327	331	4	331	4	332	5
82	195	195	0	195	0	195	0
81	314	319	5	319	5	320	6
65	469	470	1	470	1	472	3

Les tables antérieures à 2015 contiennent les données de la Haute-Garonne seulement. De 2009 à 2014, trois communes (identiques chaque année) ne trouvent pas de correspondance dans le référentiel. La table de 2018 correspond à l'année en cours et n'a aucune différence avec le référentiel. Elle n'apparaît donc pas dans le tableau.

2. Assignment des COG correspondant

La prochaine étape est d'assigner à chaque COG inexistant dans le référentiel celui de la géométrie correspondante afin d'afficher les indicateurs sur la bonne commune. Pour cela, deux méthodes ont été testées : la comparaison avec le fichier des événements sur les communes de l'INSEE et la comparaison des centroïdes.

i. Méthode par comparaison avec le fichier de l'INSEE

Le fichier des événements sur les communes contient le COG et le nom de la commune avant et après "l'événement", dont la typologie est expliquée sur le site de l'INSEE (<https://www.insee.fr/fr/information/3718969>). La liste des champs du fichier est détaillée en Annexe 1. Ce document peut donc être utilisé comme une table de correspondance en effectuant une jointure du "nivgeo" des tables statistiques sur le "com_av" contenant le COG avant événement. Par exemple, en 2018, les communes avec les COG "31277" et "31438" ont fusionné pour former une seule commune avec "31277" comme COG (Tableau 3). Il faut donc retenir pour le flux de données à afficher seulement le COG "31277" et y ajouter les valeurs de "31438" (pour obtenir 215 pour l'indicateur "total_n").

Tableau 3 : Exemple de fusion de commune.

nivgeo	com_ap	date_eff	mod	typecom_ap	total_n	SOMME (total_n)
31277	31277	2018-01-01	32	COM	130	215
31438	31277	2018-01-01	32	COM	85	
31277	31277	2018-01-01	32	COMD	130	Exclu
31438	31438	2018-01-01	32	COMD	85	

Les valeurs ont été agrégées dans des requêtes SQL qui ont été construites en prenant en compte les contraintes suivantes :

- un **SELECT DISTINCT** sur “com_av” et “com_ap” permet d’obtenir les combinaisons uniques de changements de COG (afin de ne pas agréger de doublons dans “total_n” qui équivaldrait alors à 430).
- alors que la combinaison “31277-31277” permet d’avoir la commune absorbante, la commune absorbée apparaît une fois de trop avec la combinaison “31438-31438”. Le critère qui permet de filtrer ce doublon est “typecom_ap”, dont la requête SQL (voir ci-après) n’accepte que la valeur “COM”. La valeur “COMD” signifie que la commune originale continue d’exister en tant que commune déléguée, ce qui ne nous intéresse pas pour la correspondance purement géographique.
- le champ “date_eff” ne contient que des valeurs postérieures au premier janvier de l’année du tableau, afin d’ignorer les changements depuis 1943.
- le “mod” contient les valeurs ‘31’, ‘32’, ‘33’, ‘34’ afin d’inclure uniquement les fusions administratives et les créations de nouvelles communes.

La requête SQL obtenue est exposée dans le tableau ci-après.

Tableau 4 : Script SQL permettant l’agrégation des valeurs.

<pre>CREATE TABLE v_bca2015 AS</pre>	<p>Un nouveau tableau est créé contenant les indicateurs agrégés sur les COG valides pour la géométrie. Les requêtes ajax qui permettent de réaliser les cartes seront redirigées vers ce tableau et non plus les tables « bc20xx ».</p>
<pre>SELECT mvt_com.com_ap AS nivgeo, SUM(bca2015.total_n) AS total_n, ... SUM(total_n_evo_brute) AS total_n_evo_brute</pre>	<p>Le COG “cible” correspondant à celui des géométries est requêté.</p> <p>Les champs des indicateurs sont sélectionnés en additionnant leurs valeurs.</p>
<pre>FROM bca2015 INNER JOIN (SELECT DISTINCT com_av, com_ap FROM mvt_com WHERE typecom_ap = 'COM' AND mvt_com.mod IN ('31', '32', '33', '34') AND date_eff >= '01-01-2015') AS q1 ON bca2015.nivgeo = q1.com_av GROUP BY q1.com_ap</pre>	<p>La jointure et le regroupement sont réalisés en filtrant selon les contraintes vues précédemment.</p>

<pre>UNION SELECT bca2015.nivgeo, totall_n, ... totall_evo_brute</pre>	<p>Les indicateurs des communes n'ayant pas fusionnées sont requêtés simultanément afin d'avoir une table complète.</p>
<pre>FROM bca2015 WHERE bca2015.nivgeo NOT IN (SELECT bca2015.nivgeo FROM bca2015 INNER JOIN (SELECT DISTINCT com_av, com_ap FROM mvt_com WHERE typecom_ap = 'COM' AND mvt_com.mod IN ('31', '32', '33', '34') AND date_eff >= '01-01-2015') AS q1 ON bca2015.nivgeo = q1.com_av);</pre>	<p>Afin d'éviter les doublons, les communes ayant un COG présent dans la requêtes précédente sont exclues ("NOT IN").</p>

Un cas de figure particulier est celui des fusions successives. Si par exemple "A" fusionne pour devenir "B" et "B" fusionne dans l'année du référentiel pour devenir "C", il nous faut également pouvoir assigner "A" à "C" si nous cherchons les statistiques de l'année de "A". Le fichier d'évènement sur les COG contient tous les changements, y compris de "A" vers "C", et l'agrégation des données vue plus haut n'a donc pas besoin de retracer tout l'historique des changements de COG. Nous avons tout de même testé la méthode de comparaison des centroïdes afin de vérifier que tous les cas étaient couverts par la première méthode.

ii. Méthode par comparaison de centroïdes

Le principe est de comparer par requête topologique dans quel polygone communal du référentiel se trouve le centroïde de la commune de l'année pour laquelle les statistiques sont demandées. Après avoir regroupé les fichiers GeoJSON de l'année 2015 dans une seule couche, nous avons calculé les centroïdes à l'intérieur des surfaces avec QGIS puis les avons importés dans la BD locale PostGIS.

La requête SQL suivante renvoie une agrégation sur la base des centroïdes :

```
SELECT q1.com_ap,
SUM(t_bca2015.totall_n ::INTEGER)
FROM sigmacaf.t_bca2015
INNER JOIN (SELECT fc_c_2015.codegeo,
fc_com.codegeo AS com_ap
FROM sigmacaf.fc_c_2015
INNER JOIN sigmacaf.fc_com
ON st_intersects(fc_c_2015.geom, fc_com.geom) = 'True') AS q1
ON t_bca2015.nivgeo = q1.codegeo
```

```
GROUP BY q1.com_ap;
```

Où `fc_c_2015` est la table des centroïdes. La sous-requête imbriquée assigne le COG du référentiel au COG de 2015 via un calcul d'intersection, puis elle est elle-même jointe à la requête des indicateurs (ici "total_n") via le COG de 2015.

En comparant les valeurs de l'indicateur "total_n" agrégées sur chaque COG du référentiel via la méthode des centroïdes et celle du tableau de l'INSEE, il résulte aucune différence de valeurs sur les 2950 objets à afficher. L'utilisation du tableau de l'INSEE est sous sa forme actuelle aussi fiable qu'une comparaison avec les centroïdes et, en tenant compte des avantages cités précédemment, c'est la première méthode qui a été retenue pour le développement de l'application.

La requête suivante renvoie un tableau vide, signifiant qu'il n'y a aucune différence de valeur.

```
SELECT *  
FROM sigmacaf.testcentroïdes15  
INNER JOIN sigmacaf.testdistinct15  
ON testcentroïdes15.nivgeo = testdistinct15.nivgeo  
WHERE testcentroïdes15.total_n <> testdistinct15.total_n;
```

Où "testcentroïdes15" et "testdistinct15" sont les vues des indicateurs agrégés, dans la BD locale PostGIS

V. Formulaire de mise à jour des indicateurs statistiques

La mise à jour des indicateurs statistiques est effectuée avec des requêtes SQL dans la base de données. Afin d'automatiser ces requêtes, nous avons choisi de passer par un formulaire web qui doit faciliter cette mise à jour annuelle par les agents de CAF. En effet, cette méthode nous a semblé être la plus opportune pour pallier les possibles modifications du fichier de mouvement des communes tout en restant simple d'utilisation. De plus, l'import du fichier des mouvements de COG est aussi réalisé en amont avec un formulaire d'upload afin de réaliser toute la chaîne de traitement de mise à jour des indicateurs statistiques.

Ce formulaire est divisé en quatre fichiers (Figure 10) et disponible via le lien suivant : http://193.55.175.126/testcaf/mise_a_jour/mdp.php. Le premier, "mdp.php", est un fichier de protection qui utilise un mot de passe pour accéder au deuxième fichier qui est la page d'import "upload.php". Ce fichier permet d'accéder directement au formulaire "formulaire.php" si l'import s'est bien déroulé. Enfin, le dernier fichier "action_formulaire.php" est déclenché à la validation des informations à saisir dans le formulaire.

1. Mot de passe

Le fichier "mdp.php" permet au personnel de la CAF de s'authentifier pour entrer dans le formulaire. Une entrée de type "password" sert à saisir le mot de passe et un bouton de type "submit" sert à valider ce mot de passe et à accéder au formulaire par la méthode "post".

2. Import du fichier de mouvement de COG

Le bon mot de passe permet d'accéder à la page d'import du fichier des mouvements de communes. Un mauvais mot de passe entraîne l'apparition d'un message d'erreur avec la possibilité de retourner sur la page d'authentification.

Le fichier "upload.php" permet à l'utilisateur d'aller chercher dans son ordinateur grâce à "input" du type "file" le bon fichier. Un bouton "submit" permet d'accéder au fichier "formulaire.php" (Figure 8)



The screenshot shows a web interface for uploading a file. At the top, the title reads "Importer le fichier de mouvement des communes dans la base de données". Below this, there is a text prompt: "Mettre en ligne les changements de COG (*.csv):". To the right of this prompt is a file selection button labeled "Choisir un fichier" and the text "Aucun fichier choisi". At the bottom center of the form is a dark blue button labeled "Valider".

Figure 8 : Page permettant d'importer le fichier de changement des communes dans la base de données.

3. Formulaire

Ce fichier est divisé en deux parties. La première partie sert à importer le fichier de mouvement des COG. Si l'import s'est bien déroulé, le formulaire permettant la mise à jour des indicateurs apparaît. Cinq listes déroulantes allant chercher le nom des colonnes du fichier de mouvement des communes permettent à l'utilisateur de saisir les informations demandées sans erreurs. L'année doit être tapée par l'utilisateur. Enfin, un bouton valider sert à déclencher le fichier "action_formulaire.php" (Figure 9).



Mise à jour des indices statistiques

Choisissez la colonne des codes géographiques antérieurs : com_av

Choisissez la colonne des codes géographiques actuels : com_ap

Choisissez la colonne des types de commune après changement : typecom_ap

Choisissez la colonne des modalités de changement : mod

Choisissez la colonne des dates : date_eff

Précisez la dernière année disponible: 2018

Valider

Figure 9 : Formulaire rempli

4. Action

La création des nouvelles tables v_bca20xx se fait via le fichier "action_formulaire.php". Une première boucle tourne sur les années des tables "bca" pour traiter chaque table et calculer les indicateurs mis à jour avec une deuxième boucle sur les colonnes qui additionne les indices lorsque des fusions de communes ont été identifiées. Une barre de progression s'affiche sur le navigateur pour indiquer à l'utilisateur l'état d'avancement du traitement. Cette barre est codée en JavaScript et la fonction `flush()` de PHP est utilisée pour envoyer les informations au navigateur avant que tous les traitements soient effectués. Si une erreur est présente, le navigateur indique ce qui ne fonctionne pas grâce à une fonction rajoutée au début du script : `error_reporting(E_ALL)`.

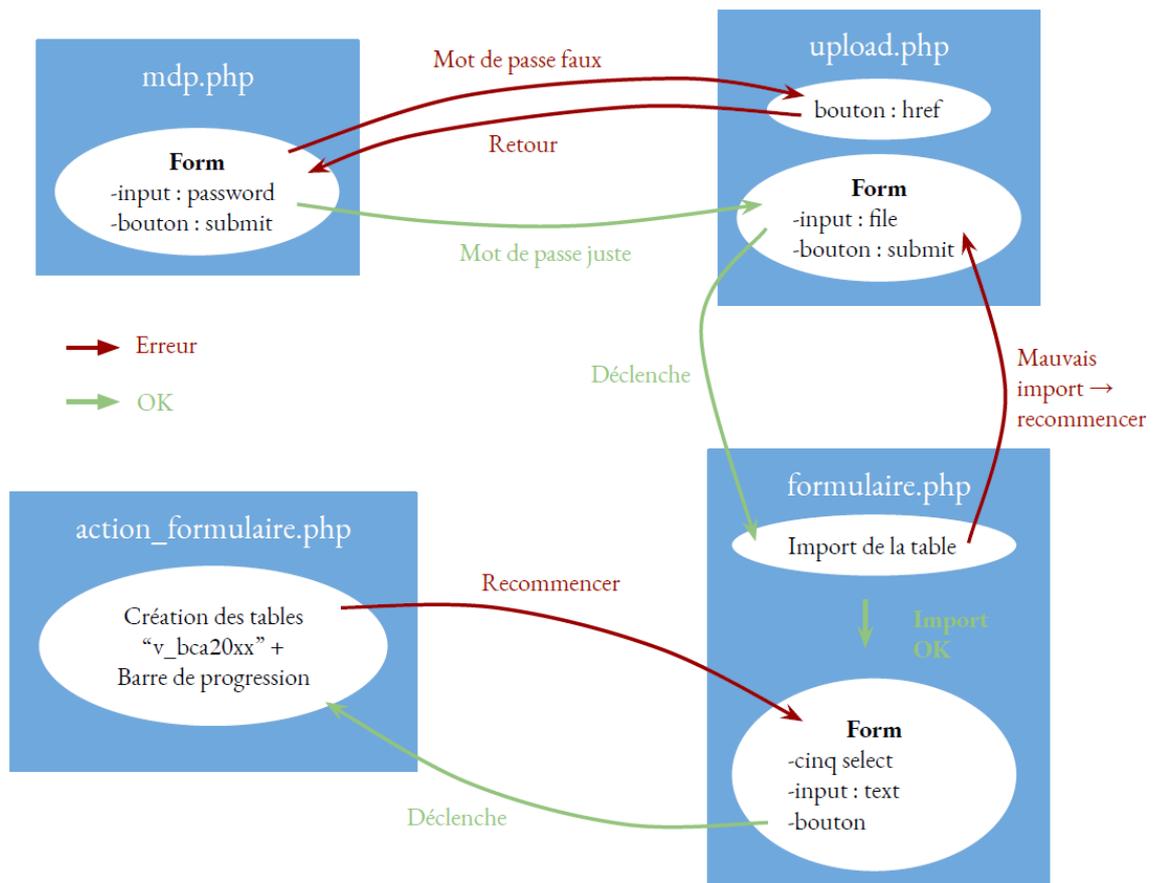


Figure 10 : Fonctionnement de la mise à jour des indices statistiques. Les rectangles bleus sont les quatre fichiers *.php avec leurs caractéristiques dans les cercles blancs.

VI. Mise à jour des géométries des communes via QGIS

Chaque année, lors de la mise à jour de leur SIG, l'équipe de la Caf de Haute-Garonne reçoit un jeu de données contenant les nouvelles géométries des communes par département. Ces dernières sont envoyées par l'antenne nationale de la CAF, au format MapInfo et avec un certain délai qui ne correspond pas forcément au calendrier de travail de la CAF 31.

De plus, l'exploitation du fichier nécessite des traitements répétitifs et non optimisés sous le logiciel Qgis qui peuvent entraîner des erreurs (problèmes de projection, d'encodage,...)

C'est pourquoi, nous avons proposé à l'équipe de la CAF 31, un outil fonctionnant sous Qgis dont les objectifs sont les suivants :

- renforcer leur autonomie, les agents de la CAF pourront gérer eux-mêmes les fusions des communes ;
- faciliter et optimiser la mise à jour du SIG en proposant un outil tout-en-un.

La mise à jour des géométries des communes est réalisée via le modeleur graphique fourni avec le logiciel Qgis. Les étapes de traitement sont présentées sur la figure 11.

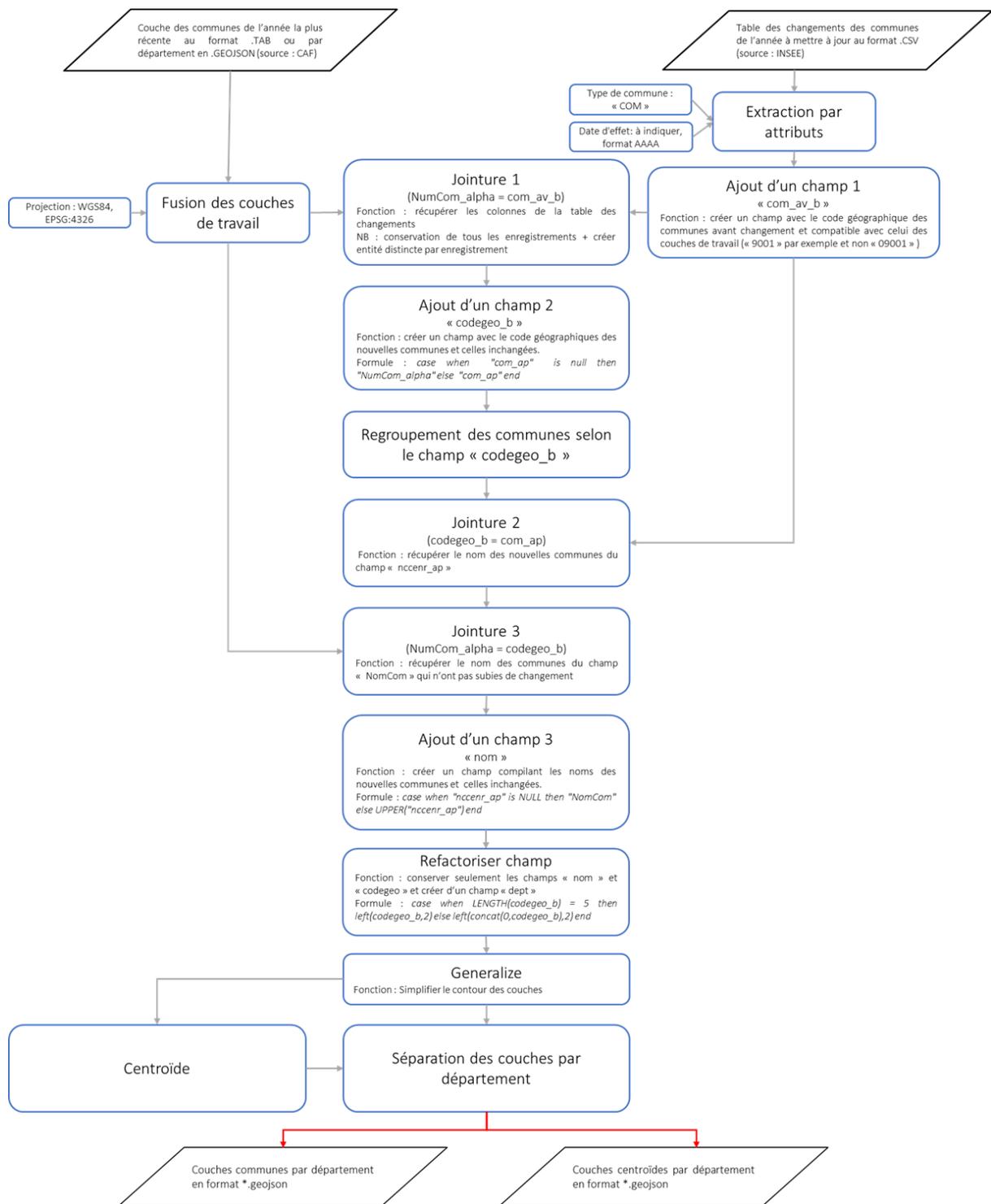


Figure 11 : Description du modèle de traitements

VII. Proposition d'une configuration de serveur adapté à l'hébergement du site

1. Configuration du serveur actuel

Afin de renouveler son système d'hébergement dans les années à venir, il est apparu nécessaire à la CAF de la Haute-Garonne de connaître les besoins techniques pour héberger leur site. Il est actuellement hébergé sur un cloud et possède deux machines virtuelles (VM) fournies par l'entreprise NFrance basé à Toulouse. Une VM présente l'avantage d'être vue par l'ordinateur des utilisateurs comme plusieurs machines et permet un temps de réponse plus rapide car elle exécute plusieurs tâches simultanément. La plus grande partie des traitements du site se fait côté client, c'est pour cela que le script est optimisé avec des requêtes asynchrones (ajax) afin que le temps de réponse/d'affichage soit le plus rapide possible.

Le serveur actuel est un VM Cloud, 2 vCPU/4 Go de ram et 50 Go de stockage. Il possède :

- un système GNU/Linux : distribution NFrance basée sur CentOS 7.x
- un serveur Web : Apache 2.4 avec PHP 7.2
- une base de données : MariaDB 10.2
- un accès Web : PhpMyAdmin
- un protocole mail : IMAP / SMTP
- un antivirus ClamAV
- un antispam : SpamAssassin
- un accès Web : webmail Roundcube
- une connectivité réseau : IPv4 et IPv6
- un transfert de fichiers SFTP, rsync over SSH
- une connexion à distance : ligne de commande SSH

Il est important de noter que les trois principaux facteurs limitants sur un serveur sont la bande passante, la RAM et le CPU.

Il existe trois types d'hébergements : l'hébergement mutualisé, le serveur dédié et le cloud. Nous leur déconseillons le serveur mutualisé à moins qu'il garantisse une bande passante minimum mais c'est rarement le cas. Le minimum de RAM proposé par un hébergeur est de 4 Go.

La CAF nous a informé qu'elle ne pouvait actuellement pas connaître le nombre d'utilisateurs en simultané sur le site. Toutefois, d'après leurs estimations il n'y en aurait pas plus de cinq en même temps. Et aucun client utilisateur ne s'est plaint de problème technique. Cette information est importante car elle nous permettra de dimensionner les besoins du site.

La bande passante correspond à la quantité d'information qui peut être transmise en simultané par seconde, NFrance garantie une bande passante de 100 Mb/s. Cela correspond à la vitesse minimale garantie que l'on peut avoir.

2. Cas d'utilisation

La Figure 12 nous renseigne sur les requêtes que peut envoyer un utilisateur au serveur et la quantité de flux reçu par ce dernier lors de sa navigation sur le site .



Figure 12 : Diagramme des cas d'utilisation pour un utilisateur.

3. Limites du site

Le navigateur Mozilla Firefox permet d'avoir accès au temps de chargement d'une page pour chacun des fichiers envoyés par un serveur. Pour cela il faut "examiner l'élément" et aller sur l'onglet "réseau". Il est possible de visualiser tous les temps de chargement de chaque fichier. Il ne faut pas oublier de vider le cache afin que la réponse vienne bien du serveur lui-même et non d'une précédente requête enregistrée sur l'ordinateur. Mozilla Firefox donne la possibilité à l'utilisateur de désactiver le cache ce qui nous permet de ne pas avoir à le vider à chaque nouvelle requête. Cette interface permet aussi de limiter la bande passante, ce qui va nous permettre de feindre la requête

en bas débit côté client (Figure 13). Il existe d'autres logiciels comme Netlimiter, NetBalancer (payant) ou TMMeter (gratuit) pour limiter la bande passante mais nous n'avons pas besoin de plus que ce qu'offrait Mozilla Firefox. De plus, nous avons comme contrainte d'utiliser des logiciels libres et gratuits.

Nous avons sélectionné trois types de requêtes qui appellent chacune un fichier différent (aplat de couleur, cercle proportionnel ou les deux) :

- ajax_ChoropletheTx3.php
- ajax_symboles_s.php
- ajax_double.php

Nous avons choisi trois niveaux d'échelles : Quartier (Cahors), Commune (Haute-Garonne) et EPCI (Midi-Pyrénées), ils seront utilisés pour toutes les étapes suivantes. Cahors fait partie d'une des plus petites villes, la Haute-Garonne est le plus grand département présent en Midi-Pyrénées et nous avons voulu aussi tester sur toute l'ex-région Midi-Pyrénées. Nous avons choisi l'année la plus récente pour effectuer nos tests (2018).

L'efficacité du site a été testée d'abord côté client, puis côté serveur. Pour le premier test, nous avons limité la bande passante en Regular 2G, en Wifi et comparé sans aucune limitation de la bande passante. Les tests ont été réalisés sur les ordinateurs de l'ENSAT branchés sur le réseau de l'école (Figure 13). Les résultats se trouvent dans les tables en Annexe 2 .

La carte avec les symboles les plus complexes, à savoir celle de la part des couples avec enfant dans la Région Midi-Pyrénées présente un temps de chargement de 30s en 2G. Les temps de chargement varient peu avec une bonne connexion, mais deviennent sensiblement plus long en 2G pour les cartes de département et de la région. D'une manière générale, une bonne connexion est requise pour une utilisation fluide du site.

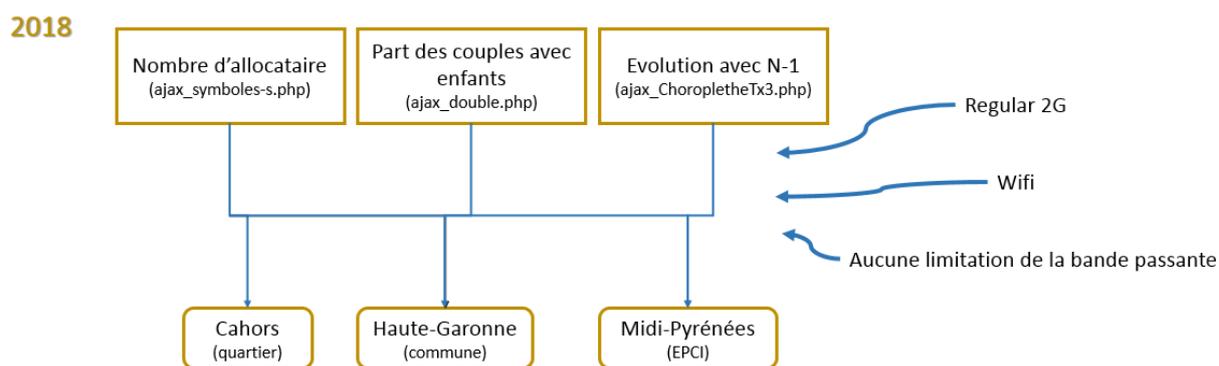


Figure 13 : Schémas des cas retenus pour les tests à différents débits.

Ensuite nous avons testé la capacité du serveur à répondre à de multiples demandes. Pour cela nous avons simulé la même requête faite par plusieurs utilisateurs en simultanément (Figure 14) en ajoutant une boucle d'itération avec compteur dans chacun des fichiers ajax.php. Le serveur devra alors générer le flux de données autant de fois qu'il l'est demandé dans la boucle. Ici nous avons commencé par simuler 10 utilisateurs, 30 utilisateurs, 50 utilisateurs et pour finir 100 utilisateurs. Un fichier *.php ne peut tourner plus de 30 secondes c'est pour cela que dans certains cas aucune réponse ne nous est revenue.

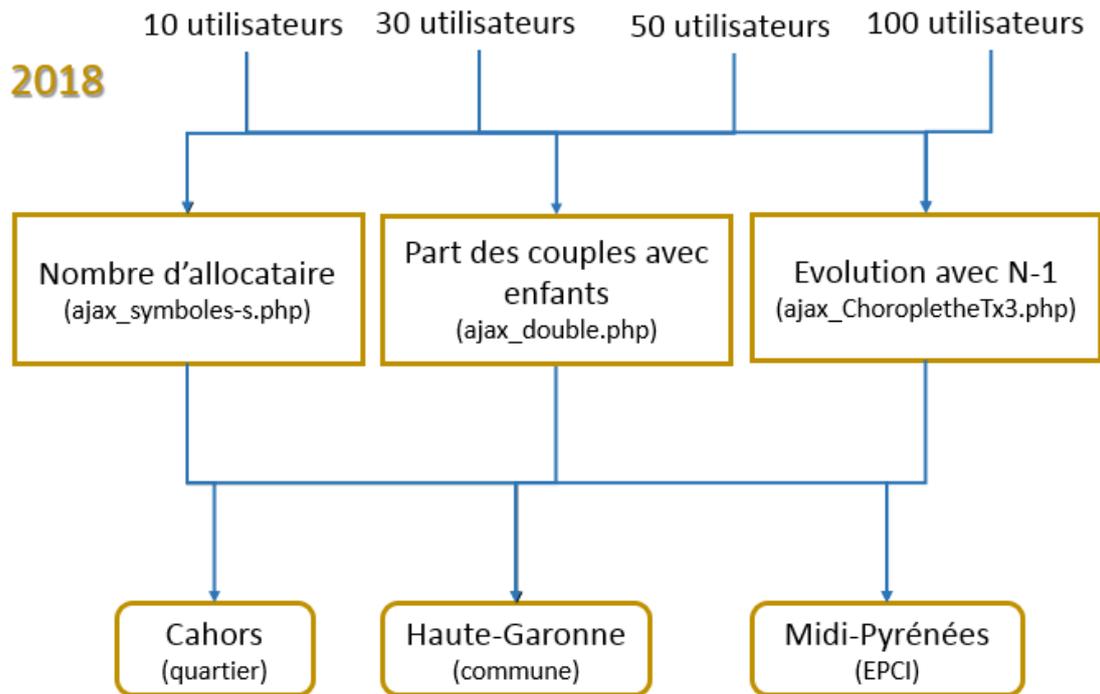


Figure 14 : Schémas des cas retenus pour le test de la capacité du serveur

Nous observons des différences croissantes de temps de chargement (Tableau 5) et nous rencontrons des erreurs à partir de 50 utilisateurs pour les requêtes les plus complexes et 100 utilisateurs pour toutes requêtes sur la Haute-Garonne. Il est possible d'anticiper que l'affichage rencontrera des problèmes entre 50 et 100 utilisateurs pour les cartes des départements. Nous conseillons à la CAF d'augmenter le nombre de RAM uniquement si elle estime avoir dans l'avenir plus de 50 utilisateurs en simultané sur son site.

Nous pouvons penser que le département de la Haute-Garonne demandera un temps de calcul toujours supérieurs aux autres départements même étendu à l'Occitanie car il contient le plus d'objets. Par exemple le département de l'Hérault d'où Montpellier est la préfecture, compte 342 communes tandis que la Haute-Garonne en compte 589. Les autres niveaux géographiques (EPCI et quartiers) ne semblent pas encore atteindre leurs limites avec environ 2s de temps de chargement pour les cartes les plus complexes à 100 utilisateurs.

Tableau 5 : Temps de chargement des fichiers ajax en fonction du nombre d'utilisateur.

Temps fichier (ms)

Cahors				
	10 utilisateurs	30 utilisateurs	50 utilisateurs	100 utilisateurs
Nombre allocataire	110	250	375	1141
Evolution avec N-1	188	187	609	1531
Part des couples avec enfants	125	391	734	2141

Haute-Garonne				
	10 utilisateurs	30 utilisateurs	50 utilisateurs	100 utilisateurs
Nombre allocataire	719	4704	4907	30000(erreur)
Evolution avec N-1	2463	12484	19001	30000(erreur)
Part des couples avec enfants	1422	14657	30000(erreur)	30000(erreur)

Midi-Pyrénées				
	10 utilisateurs	30 utilisateurs	50 utilisateurs	100 utilisateurs
Nombre allocataire	110	250	406	1187
Evolution avec N-1	292	422	671	2000
Part des couples avec enfants	126	485	719	2296

Pour estimer la consommation de la bande passante lors de nos tests nous avons regardé le poids des fichiers téléchargés pour les mêmes requêtes que précédemment (Tableau 6). Ensuite nous les avons multipliés par 50 afin de voir le poids envoyé par le serveur si 50 utilisateurs faisaient la même requête. Nous avons choisi 50 utilisateurs qui correspond au seuil à partir duquel apparaissent les premières limites techniques du serveur. Dans tous les cas nous n'utilisons pas la totalité de la bande passante (garantie à 100 Mb/s), les fichiers les plus lourds pesant 25 Mb.

Tableau 6 : Poids des fichiers téléchargés afin de tester la bande passante.

Poids fichier (Mb) pour 1 utilisateur

	Cahors	Haute-Garonne	Midi-Pyrénées
Nombre allocataire	0,194	0,309	0,407
Evolution avec N-1	0,177	0,296	0,504
Part des couples avec enfants	0,178	0,320	0,512

Poids fichier (Mb) pour 50 utilisateurs

	Cahors	Haute-Garonne	Midi-Pyrénées
Nombre allocataire	9,7	15,45	20,35
Evolution avec N-1	8,85	14,8	25,2
Part des couples avec enfants	8,9	16	25,6

*50

Tous les tests de performance ont été réalisés sur l'environnement de test mis en place sur le serveur SIGMA, cependant tout le protocole est applicable au serveur de la CAF. Le tableau 7 compare les temps de chargement de base (un utilisateur) des fichiers ajax, depuis les serveurs CAF

et SIGMA. Mis à part la troisième carte de la Région Midi-Pyrénées, les temps de réponse du serveur de la CAF sont d'un ordre de grandeur similaire à celui du serveur SIGMA. Nous pouvons donc raisonnablement nous attendre à ce que les conclusions des simulations des requêtes multi-utilisateurs soit applicables au serveur de la CAF.

Tableau 7 : Comparaison du temps de chargement entre le serveur de la CAF et le serveur test de l'Université.

Temps fichier (ms)

Cahors		
	Site CAF	Site test
Nombre allocataire	36	31
Evolution avec N-1	50	16
Part des couples avec enfants	44	32
Midi-Pyrénées		
	Site CAF	Site test
Nombre allocataire	58	47
Evolution avec N-1	55	93
Part des couples avec enfants	121	47
Haute-Garonne		
	Site CAF	Site test
Nombre allocataire	35	31
Evolution avec N-1	39	64
Part des couples avec enfants	41	63

Pour conclure, les différents tests effectués semblent montrer que le serveur actuel correspond aux attentes de la CAF. Nous pensons qu'il restera suffisant performant même en élargissant le SIG à toute l'Occitanie. Il faudra penser à changer de serveur ou au moins augmenter la RAM et la bande passante si le nombre d'utilisateur en simultané dépasse 50.

4. Evaluation des besoins de migration vers PostgreSQL/PostGIS

Pour rappel le système de base de données actuel de la CAF est MySQL avec l'extension géographique MariaDB. La CAF envisage pour 2020 ou 2021 un passage à PostgreSQL/PostGIS. L'avantage que présente une BD avec extension spatiale est de pouvoir garder pour chaque entité des tables une géométrie, et donc de pouvoir chercher par une même requête à la fois les données attributaires et les fonds de carte. Cependant cette méthode alourdit considérablement les flux de données et les temps de chargement de la carte en plaçant le plus gros volume des traitements sur le serveur. Dans la configuration actuelle du site, les géométries stockées dans les fichiers GeoJSON sur le serveur sont pré-chargées par le client et affichées comme fonds de carte vide. Le client fait ensuite la jointure entre les indicateurs et le fond de carte pour créer la carte thématique. Le serveur a ainsi une charge de calcul modéré qui lui permet de mieux répondre à plusieurs requêtes simultanées. Il faut noter que si le site devait s'articuler intégralement autour d'un moteur BD, il faudra repenser la façon dont il est construit (la jointure d'indicateurs et de géométries ne serait plus requise). En recherchant des comparaisons de MySQL et PostgreSQL sur Internet, il ressort principalement les différences suivantes entre les deux moteurs de BD :

- PostgreSQL est réputé plus rapide lorsque le volume de données demandé est important ou si les requêtes sont complexes et contiennent des opérations sur les données ;
- MySQL est réputé plus performant pour exécuter des requêtes de données simples, sans opérations et en lecture seule.

Dans son fonctionnement actuel, le site fait appel à trois types de requêtes :

- Les requêtes de mise à jour des tables développées par notre application : ces requêtes sont complexes et nécessitent des opérations d'agrégation sur un volume important de données, mais elles sont exécutées qu'une fois l'an.
- Les requêtes des indicateurs: leur complexité varie en fonction du type de symbole, de la requête simple à la requête imbriquée avec jointures et filtres. MySQL semble cependant suffisamment performant: les flux les plus complexes ne mettent pas plus de 100 ms à être générés (pour un seul utilisateur, voir table en annexe).
- Le chargement des géométries : le choix de stocker les géométries dans des fichiers plutôt que dans des BD permet un chargement rapide des fonds de carte et permet au client de les mettre en cache pour un accès futur plus rapide. Alternativement, il serait possible de requêter simplement les géométries à partir de tables MariaDB s'il devenait indispensable d'avoir les géométries dans une BD. Ces requêtes sont des appels de données simples, sans opérations, ce qui ne nécessiterait pas une migration vers PostGIS.

Le véritable avantage de PostgreSQL et de son extension spatiale PostGIS réside dans les traitements qui peuvent être exécutés lors d'une requête : par l'exemple des calculs de longueurs, surfaces, périmètres, des transformations de référentiel géographique, des opérations de géotraitement (union, intersection, centroïdes, etc...). Tant que de telles opérations n'ont pas besoin d'être exécutées à chaque requête de données, une migration vers PostgreSQL n'est selon nous pas justifiée car les requêtes actuelles ne font pas appel à des fonctions d'analyse complexes. Notons que les fonctionnalités spatiales qui sont utilisées dans notre application QGIS, à savoir la fusion de géométries, la création de centroïdes et l'écriture en GeoJSON, sont disponibles dans MariaDB si jamais le site et son application devaient être intégralement refait autour d'un moteur de BD.

Ainsi, au vu des besoins actuels du site, MySQL/MariaDB offrent des fonctionnalités ainsi que des performances suffisantes. Tant que les requêtes de données ne se complexifient pas par l'ajout de traitement géospatiaux lourds, PostGIS n'est pas utilisé au meilleur de son potentiel.

VIII. Perspectives d'améliorations

L'application web que nous avons développée a été conçue avec une certaine adaptabilité, par exemple afin de pouvoir anticiper un changement de structure dans le fichier des changements de COG (choix des noms de colonnes depuis le fichier mis en ligne). Néanmoins, des pistes d'améliorations peuvent être proposées :

- un code PHP qui permettrait de vérifier que le type de données des colonnes choisies dans les listes déroulantes du formulaire corresponde au type de valeur dans les filtres SQL ;
- au moment du upload du fichier de mouvement de COG, des cases à cocher pour sélectionner les colonnes à importer dans la BD ;
- un code pour détecter et assigner automatiquement le bon format de données aux colonnes à importer ;
- améliorer la gestion des erreurs pour la partie formulaire afin que l'utilisateur soit exactement au courant de ce qui se déroule ;
- dans sa forme actuelle, l'application web est programmée pour rapporter les erreurs PHP et pour notifier un message d'erreur lorsqu'une requête SQL n'a pu être exécuté dans la BD. Cependant, toutes les requêtes SQL sont conçues pour écraser les tables précédentes. Si une requête syntaxiquement juste mais avec un contenu erroné devait être lancée, les tables originales seraient perdues. On pourrait ajouter des conditions aux pré-requêtes de "DROP TABLE" pour vérifier la qualité de la table à créer. En attendant, les tables pré-existantes sont sauvegardées dans des copies avant d'être supprimées ;
- mettre à jour MariaDB afin de bénéficier de requêtes avancées (exemple "OPTIMIZE TABLE").

Références bibliographiques

[1] Rapport de stage, Eliette Catelin, Septembre 2018

[2] Damien Paque, « Gestion de l'historicité et méthodes de mise à jour dans les SIG », *Cybergeo : European Journal of Geography* [En ligne], Cartographie, Imagerie, SIG, document 278, mis en ligne le 23 juin 2004, consulté le 24 février 2020. URL : <http://journals.openedition.org/cybergeo/2500> ; DOI : <https://doi.org/10.4000/cybergeo.2500>

[3] Christine Plumejeaud-Perreau. Modèles et méthodes pour l'information spatio-temporelle évolutive. *Cartes & géomatique*, Comité français de cartographie, 2013, pp. 33-38. <hal-00925751>

[4] Christine Plumejeaud, Jérôme Gensel. Complexité liée à la variabilité sémantique des statistiques socio-économiques. 2010. <hal-00953209>

[5] Patricia Bordin. L'intégration du temps dans les SIG / *in XYZ, n° 122 (mars - mai 2010)*

ANNEXES

Annexe 1 Liste des champs de la table des événements survenus aux communes
(Source : Insee)

– Dessin du fichier « Liste des événements survenus aux communes, arrondissements municipaux, communes associées et communes déléguées depuis 1943 » et liste des variables

Nom de la variable	Longueur de la variable	Désignation et modalités de la variable
MOD	2	Type d'événement de communes
DATE_EFF	10	Date d'effet (AAA-MM-JJ)
TYPECOM_AV	4	Type de la commune avant événement
COM_AV	5	Code de la commune avant événement
TNCC_AV	1	Type de nom en clair
NCC_AV	200	Nom en clair (majuscules)
NCCENR_AV	200	Nom en clair (typographie riche)
LIBELLE_AV	200	Nom en clair (typographie riche) avec article
TYPECOM_AP	4	Type de commune après l'événement
COM_AP	5	Code de la commune après l'événement
TNCC_AP	1	Type de nom en clair
NCC_AP	200	Nom en clair (majuscules)
NCCENR_AP	200	Nom en clair (typographie riche)
LIBELLE_AP	200	Nom en clair (typographie riche) avec article

– Modalités pour le type de commune

COM	Commune
COMA	Commune associée
COMD	Commune déléguée
ARM	Arrondissement municipal

– Modalités pour le type d'événement

Type de modification	Code
Changement de nom	10
Création	20
Rétablissement	21
Suppression	30
Fusion simple	31
Création de commune nouvelle	32
Fusion association	33
Transformation de fusion association en fusion simple	34
Changement de code dû à un changement de département	41
Changement de code dû à un transfert de chef-lieu	50
Transformation de commune associée en commune déléguée	70

Annexe 2 : Résultats des tests de performance

Temps de chargement en seconde de la carte thématique via différentes bandes passantes, pour un utilisateur, par type de symbole et par étendue géographique.

Cahors			
	Regular 2G	Wifi	Aucune limitation
Nombre allocataire	14,34	9,67	9,09
Evolution avec N-1	13,3	8,51	13,11
Part des couples avec enfants	15,12	9,88	11,81

Haute-Garonne			
	Regular 2G	Wifi	Aucune limitation
Nombre allocataire	20,5	10,51	10,29
Evolution avec N-1	26,95	17,53	18,77
Part des couples avec enfants	24,56	12,28	13,63

Midi-Pyrénées			
	Regular 2G	Wifi	Aucune limitation
Nombre allocataire	28,09	8,26	12,5
Evolution avec N-1	27,97	11,8	10,8
Part des couples avec enfants	29,94	10,74	9,75

Temps de chargement en millisecondes du fichier ajax via différentes bandes passantes, pour un utilisateur, par type de symbole et par étendue géographique.

Cahors			
	Regular 2G	Wifi	Aucune limitation
Nombre allocataire	318	12	31
Evolution avec N-1	311	9	16
Part des couples avec enfants	318	3	32

Haute-Garonne			
	Regular 2G	Wifi	Aucune limitation
Nombre allocataire	344	47	47
Evolution avec N-1	363	50	93
Part des couples avec enfants	971	62	47

Midi-Pyrénées			
	Regular 2G	Wifi	Aucune limitation
Nombre allocataire	315	3	31
Evolution avec N-1	315	19	64
Part des couples avec enfants	318	3	63